Visual Encoding Quality and Scalability in Information Visualization

by

Rafael Veras Guimarães

A thesis submitted to the
School of Graduate and Postdoctoral Studies in partial
fulfillment of the requirements for the degree of

Doctor of Philosophy in Computer Science

Faculty of Science

University of Ontario Institute of Technology

Oshawa, Ontario, Canada

February 2019

© Rafael Veras Guimarães, 2019

Rafael Veras Guimarães: *Visual Encoding Quality and Scalability in Information Visualization*
Doctor of Philosophy

SUPERVISOR:
Christopher Collins

COMMITTEE:
Ken Pu
Mark Green
Miguel Vargas Martin
Remco Chang (external)

LOCATION:
Oshawa, Ontario, Canada

COMPLETED:
February, 2019

©Rafael Veras Guimarães, 2019.

THESIS EXAMINATION INFORMATION

Submitted by: **Rafael Veras Guimaraes**

Doctor of Philosophy in **Computer Science**

Thesis title: VISUAL ENCODING QUALITY AND SCALABILITY IN INFORMATION VISUALIZATION

An oral defense of this thesis took place on **February 22, 2019** in front of the following examining committee:

Examining Committee:

Chair of Examining Committee	SHAHRAM HEYDARI
Research Supervisor	CHRISTOPHER COLLINS
Research Co-supervisor	
Examining Committee Member	KEN PU
Examining Committee Member	MARK GREEN
University Examiner	MIGUEL VARGAS MARTIN
External Examiner	REMCO CHANG, TUFTS UNIVERSITY

The above committee determined that the thesis is acceptable in form and content and that a satisfactory knowledge of the field covered by the thesis was demonstrated by the candidate during an oral examination. A signed copy of the Certificate of Approval is available from the School of Graduate and Postdoctoral Studies.

AUTHOR'S DECLARATION

I hereby declare that this thesis consists of original work of which I have authored. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize the University of Ontario Institute of Technology to lend this thesis to other institutions or individuals for the purpose of scholarly research. I further authorize University of Ontario Institute of Technology to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research. I understand that my thesis will be made electronically available to the public.

The research work in this thesis that was performed in compliance with the regulations of UOIT's Research Ethics Board/Animal Care Committee under **REB Certificate Numbers #14590 and #15025.**

Rafael Veras Guimaraes

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LISTINGS

ABSTRACT

Information visualization seeks to amplify cognition through interactive visual representations of data. It comprises human processes, such as perception and cognition, and computer processes, such as visual encoding. Visual encoding consists in mapping data variables to visual variables, and its quality is critical to the effectiveness of information visualizations. The scalability of a visual encoding is the extent to which its quality is preserved as the parameters of the data grow. Scalable encodings offer good support for basic analytical tasks at scale by carrying design decisions that consider the limits of human perception and cognition. In this thesis, I present three case studies that explore different aspects of visual encoding quality and scalability: information loss, perceptual scalability, and discriminability.

In the first study, I leverage information theory to model encoding quality in terms of information content and complexity. I examine how information loss and clutter affect the scalability of hierarchical visualizations and contribute an information-theoretic algorithm for adjusting these factors in visualizations of large datasets.

The second study centers on the question of whether a data property (outlierness) can be lost in the visual encoding process due to saliency interference with other visual variables. I designed a controlled experiment to measure the effectiveness of motion outlier detection in complex multivariate scatterplots. The results suggest a saliency deficit effect whereby global saliency undermines support to tasks that rely on local saliency.

Finally, I investigate how discriminability, a classic visualization criterion, can explain recent empirical results on encoding effectiveness and provide the foundation for automated evaluation of visual encodings. I propose an approach for discriminability evaluation based on a perceptually motivated image similarity measure.

Keywords: HCI; information visualization; perception; visual data analysis; statistics

ACKNOWLEDGEMENTS

I would like to thank my mother, Socorro, and my father, Adelino, for investing a lot in my education and for fuelling my desire to know more with their recognition and incentive. During my childhood, my mother spent much of her time helping me with my assignments after classes. I credit to her my passion for learning.

I'm very grateful to Dr. Chris Collins, my supervisor. Chris taught me lessons that I will carry with me for years to come. Before I came to Canada, I already had an attention to detail, but Chris pushed me to take it to the next level. He taught me how to communicate my research in an accessible way, to write clearly, and inspired me with his ambition. He gave me access to many opportunities and offered generous funding for almost a decade. He set the standards high and it took years for me to comfortably meet the expectations. His influence in my future work will be undeniable.

Special thanks go to my friends Mariana, Jay, David, Daniel, Guilherme, and all members of the vialab at UOIT, for the camaraderie during all these years of life in the lab. I would like to thank my friend Bianchi for the genuine friendship and for the great examples of leadership during the time he was my professor. Bianchi is a master in assembling teams, fostering their unity, and growing talent. When I was part of his team I learned too many lessons, all by example.

Finally, and most importantly, I would like to acknowledge the role my spouse, Brittany, had in this achievement. She delayed some of her plans to be by my side and shared the financial burden of graduate studies. With me, she celebrated paper submissions and acceptances, and cursed the rejections. In all aspects, specially the emotional, I lived with her a life well above graduate student standards, and because of her I will remember my PhD as much more than time in school.

Oshawa, ON, Canada, February 2019

STATEMENT OF CONTRIBUTIONS

I hereby certify that I am the sole author of this thesis and that I am the sole source of the creative works and inventive knowledge described in this thesis. Part of the work described in Chapters 3 and 4 have been published as:

Veras, Rafael and Christopher Collins

2017 "Optimizing Hierarchical Visualizations with the Minimum Description Length Principle", *IEEE Transactions on Visualization and Computer Graphics*, 23, 1, pp. 631-640.

2019 "Saliency Deficit and Motion Outlier Detection in Animated Scatterplots", in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, ACM, Glasgow, UK.

1 INTRODUCTION

During the past twenty years, information visualization went from a niche human-computer interaction topic to a critical component in the emerging data science curriculum. A market now exists for applications dedicated to visualizing data, and visualization became mainstream in news media. Research communities are discussing its role in science education, and in the digital humanities. With the promise of amplifying cognition through the visual representation of abstract data, visualization proposes methods for mapping data to images and techniques for interacting with these representations. Visualization research has experienced a boom in the last decade, with contributions that advanced our understanding of graphical perception, interaction, literacy, scalability, and many other topics.

Despite the rapid growth, visualization needs to overcome some issues to become a mature field. At the core of this discussion are the lack of agreement upon a theoretical foundation for visualization, a poor understanding of the factors that drive visualization effectiveness, and a lack of standard and convenient methods for evaluating new designs.

Theoretical frameworks grounded in information theory and mathematics have recently been proposed (Chen and Golan, 2016; Kindlmann and Scheidegger, 2014), and we are starting to see applications and models that build upon them (Correll et al., 2018; Faust et al., 2017). In the midst of the reproducibility crisis in psychology, there has been an effort to revise basic visualization assumptions; some classic experiments were redone (Kim and Heer, 2018), while new questions are being examined empirically (Zgraggen et al., 2018). The area that has seen the least progress is evaluation; designers and engineers still need to pick among methods that are either too costly or inappropriate in order to evaluate their tools.

Furthermore, there are signs of misalignment between the assumptions and goals of visualization research and visualization in practice. An example is the role that visualization plays in machine learning. Machine learning is a lively and expanding field that is drastically changing business models and people's lives. With machine learning being the *de facto* source of data-driven insights, one would expect visualization to be a critical tool. It appears that this is not the case. A recent study has shown that visualization is not used until the communication part of the workflow,

when data scientists need to disseminate results (Batch and Elmqvist, 2018). Visualization is not considered during the exploratory phase for taking too much time, because raw numbers or tables are thought to convey more information, or for being just unnecessary. When visualization is used, it is not interactive.

One of the main tools machine learning practitioners use for visualization is an interactive notebook called Jupyter which, by one account, is only less popular than Python and R as a general tool (Kaggle, 2017). In Jupyter, interaction has a very different connotation than that found in visualization research. Users type code and get "instant" feedback on the computation results, sometimes in the form of data plots. This interactivity is very different than the manipulation of GUI controls or the direct manipulation of plot elements that are common in visualization research. In a sense, it is much more primitive, less fluid. But if the notebook is interactive in essence, what prevents one from adding interactive controls to the plots for more fluid interaction? The answer may be the data scale. Most of the time when dealing with real-world datasets the size of the data prevents fluid interaction. For instance, t-SNE, a visualization method for high-dimensional vectors can take anywhere from seconds to hours to produce an image, depending on the dataset size (van der Maaten and Hinton, 2008). But if it took only 10 seconds, it would still be prohibitively slow for fluid interaction.

How can visualization be scalable when interaction, the main solution we have to explore large data spaces, is not viable? We hope that the few visualizations we make have good quality. Scalability means to retain quality as the size and complexity of the data increases. Here I refer specifically to visual encodings (also known as visual mappings), which are the methods used to map data variables to visual channels; for instance, in scatterplots horizontal and vertical position are used to encode a pair of numerical data variables. Some dimensions of quality of visual encodings are discriminability, expressiveness, and information content. These criteria are proxies for the ability of a visualization to support important tasks, such as cluster detection, outlier detection, and the estimation of summary statistics (e.g., mean).

Interaction is a solution to the problem of scale as it offers mechanisms for navigating information spaces. If a single view of the data is not sufficiently informative, users can obtain additional views by panning, zooming, and filtering. The scale achieved through interaction is bounded by the efficiency of the interaction approach, the quality of the visual encoding, and human energy and time. Information foraging theory helps us understand the interplay between encoding and interaction: humans use available information to estimate resource costs and opportunity costs; these costs are weighed to devise the best information seeking strategy

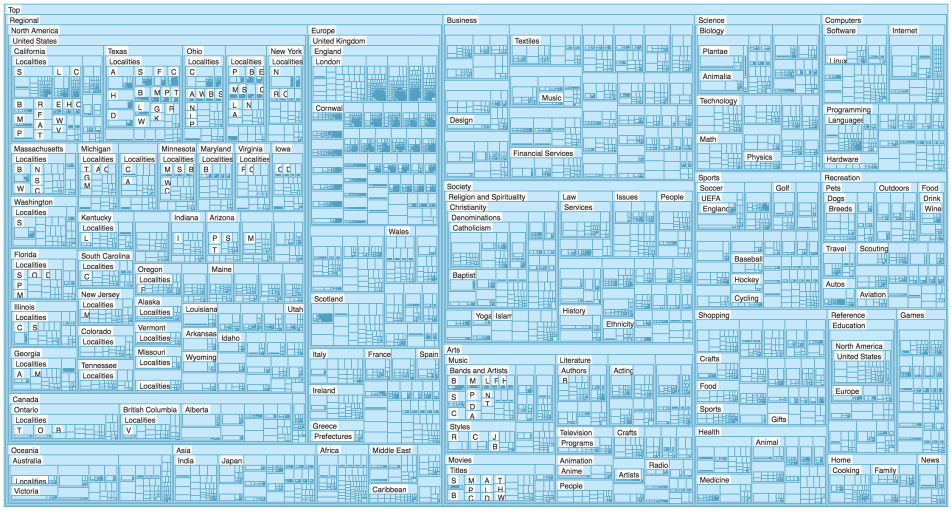


Figure 1.1: A treemap representation of a large hierarchy. Due to lack of space, most categories are represented with tiny dots, or simply not drawn, producing illegible dark blobs. A summarization strategy needs to account for the available display space and the importance of each data point.

(Pirolli and Card, 1999). As such, the quality of the visual encoding influences the results of the interactive experience. The pursuit of better visual encoding quality is, thus, not limited to situations where interaction is not available. Interaction *relies* on visual encoding to provide cues for information foraging.

In the present thesis, I investigate issues of scalability and quality of stand-alone visualizations from three different angles. First, I take an information-theoretic approach to balance information loss and clutter in aggregated views of large hierarchical datasets. Second, I explore the gap between displayed information and perceived information with a controlled experiment that evaluates the extent to which motion outlier detection is supported in multivariate animated scatterplots. Finally, I propose an automated approach for testing the discriminability of visualization encodings. In the next sections, I briefly introduce each of these contributions.

Information Loss

Data plots are in the middle of an information reduction pipeline that begins with data collection and ends with reasoning (Chen and Golan, 2016). The data is sampled from the real world, stored in some representation, mapped into visual representations, then reduced to a visual impression. The space where the data lives in is progressively constrained along this

pipeline. Depending on the data scale and the visualization type, the amount of information loss in the plotting stage varies drastically; from the loss of price precision in stock price charts to large numerical arrays in t-SNE. Information loss causes ambiguity and reduces expressiveness, but it also lets people extract information more easily.

Uncertainty and errors that arise from data collection are widely acknowledged and many visualization techniques exist for representing this kind of error (e.g., error bars). But the uncertainty that arises from information loss at the visual mapping stage and the one resulting from not accounting for ambiguity are less understood. For instance, hierarchical edge bundling, a technique for grouping edges, makes graphs more readable but reduces their discriminability: changes in the graph connectivity may "hide" within the bundles. As a result, different datasets may yield the same image. This trade-off between simplicity and precision permeates the design of many visualization techniques, especially hierarchical visualizations, which are suited to simplification.

In Chapter 3, I investigate the balance between clutter and information loss in hierarchical displays (Figure 1.1). I present a technique that embeds the goal of reducing information loss and the constraint of clutter reduction. Grounded in the information-theoretic principle of minimum description length, the approach consists in treating visualizations as models of the data and using a criterion for selection that is similar to the ones used to select among statistical models. The result is a technique that can prune a hierarchy for display in a screen of a given size. My results show that the technique affords near-constant information density across screen sizes while keeping a low level of clutter.

Saliency Deficit

If the visualization pipeline reduces information, we may think that one way to work around information loss is to simply represent more dimensions. This would be correct if human perception, a component of the pipeline, was unlimited. While the relative sophistication of the human visual system is used to justify visualization as a tool, the reality is that many capacity limits impose obstacles to visualization. For instance, we have great difficulty in detecting unique colors in displays with more than five colors (Haroz and Whitney, 2012).

The gap between displayed information and perceived information happens to be the least understood area of the visualization process. Our lack of understanding gave rise to many visualization designs that now are starting to fall out of flavor due to empirical findings that demonstrate their ineffectiveness (Harrison et al., 2014).

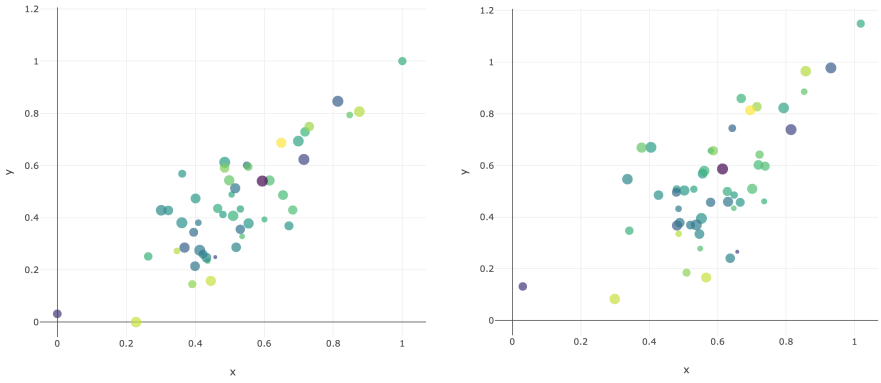


Figure 1.2: Animation is used in scatterplots to allow better perception of changes between data states. Speed of motion encodes the difference in the values represented by the coordinates x and y. The complexity of these charts increases with the number of data points, and the number of data variables represented. Here, color and size encode distinct variables, making this a 4D plot. Are we able to identify a motion outlier independently of other visual features, as we would easily using a univariate visualization (e.g., boxplot)?

One of the mechanisms that modulate performance in visualization tasks is saliency detection, a critical component of attentional control. It allows us to save resources by focusing on select regions of a scene. This natural importance-assignment mechanism can be leveraged to facilitate information processing or it can make it *more difficult* if its allocation is misaligned with the user's task. Computational models of saliency exist, but they only help us predict eye movements, and we don't know to what extent these sensorial processes are offset by goal-oriented strategies. That is, it may not matter that salience does not help as long as the user has a good strategy to accomplish the goal. For this reason, much research is necessary to learn how saliency interacts with information visualization tasks.

In Chapter 4, I contribute the results of an experiment that asked people to detect motion outliers in animated scatterplots with multiple visual dimensions 1.2. With the task focusing on a single dimension (motion) the study seeks to find whether people can perform efficiently in the presence of many other dimensions that can act as distractors. This question is fundamental to encoding scalability. To what extent can we add information to a plot without making it extra difficult to accomplish basic tasks?

Scalable Evaluation

To explain the issue of scalable evaluation in visualization, let's draw a parallel with evaluation in the field of machine learning. Evaluation

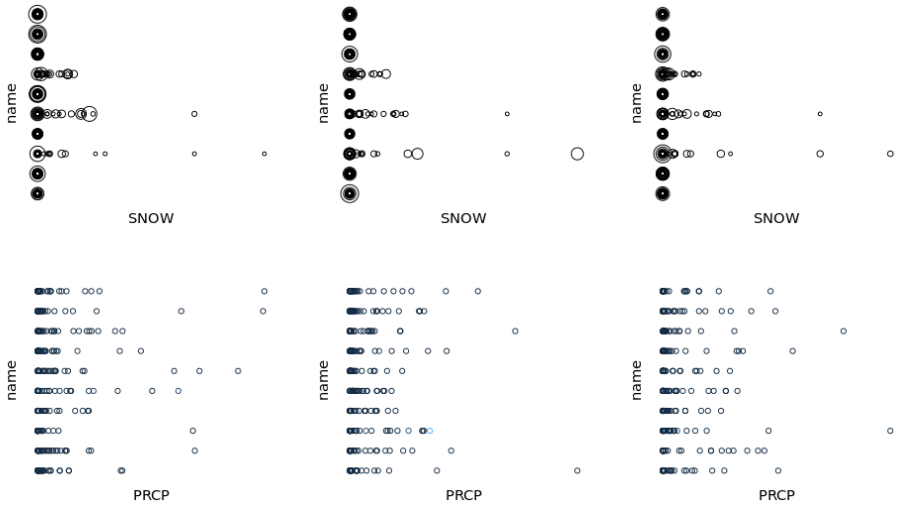


Figure 1.3: A set of three trivariate datasets. In the top row, size encodes the variable PRCP (precipitation) and horizontal position encodes the variable SNOW. In the bottom row horizontal position encodes PRCP and color encodes SNOW. Which encoding allows better perception of differences in the values of PRCP? Would the answer be the same if the values of SNOW were not skewed?

methods in machine learning are split into two classes: offline and online evaluation. These classes are concerned with very different metrics. On one hand, online evaluation methods measure indicators that businesses care about, such as customer retention and customer engagement. Online methods are costly and slow because they require deployment of models to production. On the other hand, offline methods are based on the abstraction of the problem and the separation from the model's real-world use; an example is the cross-validation framework, used with metrics such as precision, recall, and accuracy. The problem of improving student retention is reduced to predicting student performance in classes, which is in turn reduced to a label matching problem. Machine learning would not have grown as it grew if it wasn't for offline evaluation, as it is far more scalable than online evaluation.

Visualization evaluation is dominated by online methods. The equivalent of measuring business metrics in visualization research is measuring user performance or collecting user opinions and insights in experiments where users are asked to use a tool to analyze a real dataset. While online methods have their place in the research evaluation life cycle, overreliance on them creates research that can not be compared, because its evaluation is contaminated with factors that cannot be reproduced. This problem affects especially the evaluation of new encodings and layouts.

A visualization field with scalable evaluation methods would have frameworks and metrics that isolate the visual mapping from the tool, from user populations, and from decision making issues. Such measures would assess specific visualization properties such as discriminability, ambiguity, clutter level, saliency, information loss, and uncertainty. This would allow the creation of visual encoding benchmarks. Evaluation approaches that are affected by decision making issues, such as cognitive biases, literacy, or domain knowledge, would still be available to research with broader scope.

The obstacles to realizing this ideal are many. On top of the list is the difficulty in modeling human visual judgments. Second is establishing which and how measurable properties impact visualization effectiveness. Third is defining a general framework that can be used with a wide variety of visualizations. Chapter 5 presents work that make advances in these three fronts. I test an image similarity measure against empirical plot similarity data and propose the use of discriminability tests based on it (Figure 1.3). The results show that the discriminability scores can help explain recent empirical results regarding the effectiveness of visualization encodings.

Summary

In summary, this thesis presents the following contributions:

- A technique for summarizing large hierarchies for visualization purposes (Chapter 3).
- The findings of a controlled experiment designed to assess the effectiveness of motion outlier detection in multivariate animated scatterplots (Chapter 4).
- A method for scoring the discriminability of visualization encodings (Chapter 5).

In Chapter 2 I discuss the problem of scale in visualization and how it relates to visual encoding quality, and in Chapter 6 I conclude this thesis by discussing how the contributions relate to each other, and by outlining the possibilities these contributions open for future research.

2 SCALABILITY AND QUALITY

We can analyze data without plotting it. We can compute statistics or simply read the data values. That is the way my grandfather analyzed daily the sales of his bakery, which he recorded manually and kept in a notebook. He used a calculator to compute aggregates and detected trends without any visual aid. He stayed in business for decades, even after computers and Microsoft Excel became popular in small businesses. The question of when we need visualization is equivalent to the question of when summary statistics or tables are not enough. Like handwritten summary statistics and tables, a visualization is a model of the data, and can be assessed in terms of complexity and accuracy. Given the choice, we weigh this trade-off between representations to select them, in addition to many other considerations, such as the adequacy to the medium, and whether the data is to be communicated. Visualization is most often needed when we do not trust the statistics and when tables are not adequate given the scale of the data. When the nature of the task and the nature of the data allow us to trust a single statistical estimate, then that estimate is a more efficient way to carry the information. We will choose the most trustful way to represent the data provided it is sufficiently concise. Tables are more faithful than visualizations and statistics, but do not scale well.

We can observe this tension between representations (numbers, tables, and images) in journalism. For instance, much country-wide data is reported as a single statistical estimate, like a percentage, while others, such as the voting intentions for presidential candidates during election time, are more likely to be represented in a map. Sometimes the high compression rate of a statistical estimate does not yield enough trust—as when what is at stake is the name of the next president, but is appropriate when the decisions are less consequential. In any case, the data comes from large surveys and goes through a data processing pipeline where it gets reduced. The degree to which the reduction is acceptable helps to determine which representation is most appropriate.

The classic case for visualization is by Francis Anscombe, who demonstrated the effect of outliers on statistical properties through four distinct datasets that yield identical or very similar values for mean, variance, correlation, and regression line coefficients (Figure 2.1) (Anscombe, 1973). By plotting them, we can see that the datasets are wildly different, and thus,

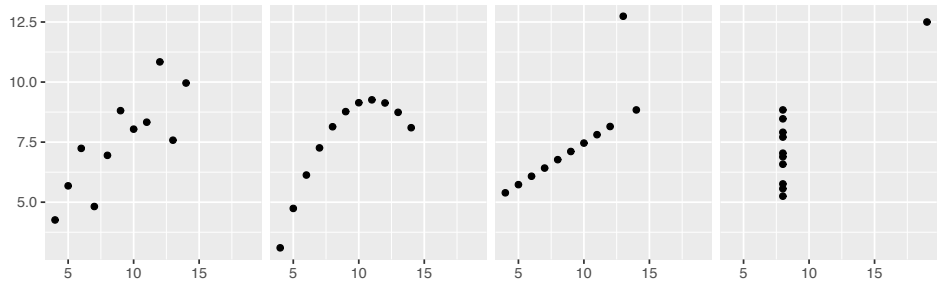


Figure 2.1: Anscombe's Quartet. Statistics computed on these datasets are near identical.

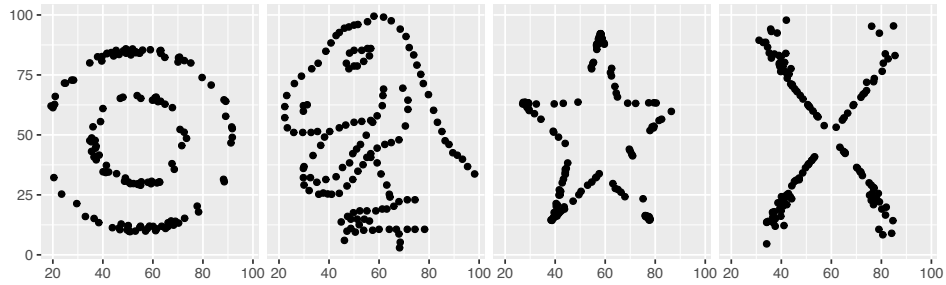


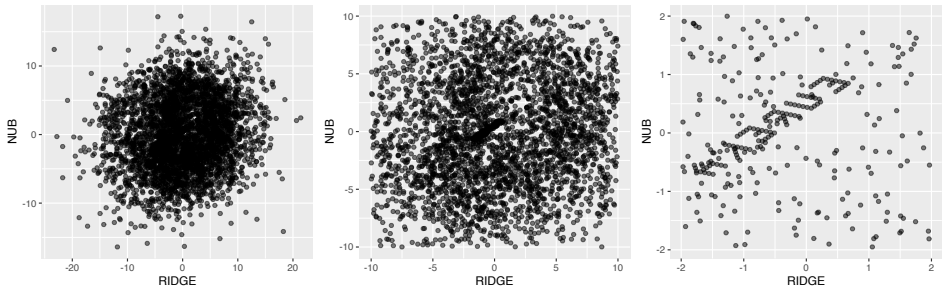
Figure 2.2: Four of Matejka and Fitzmaurice's twelve datasets, also known as "Datasaurus dozen". Similar to Anscombe's, these datasets produce equal statistics.

the statistic properties are misleading. Later, Matejka and Fitzmaurice (2017) proposed a method to generate datasets akin to Anscombe's, but with arbitrary shapes (Figure 2.2).

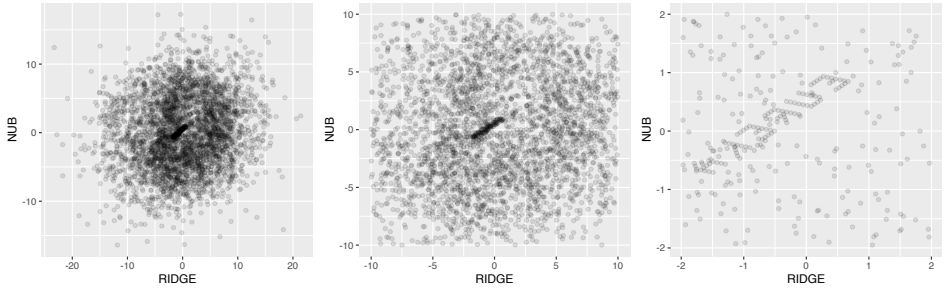
Anscombe's quartet shows how statistics can be unreliable, and it makes evident that we have the ability to estimate visually many properties of a dataset. Anscombe summarizes the virtues of visualization and compares it to statistics as follows:

> Graphs can have various purposes, such as: (i) to help us perceive and appreciate some broad features of the data, (ii) to let us look behind those broad features and see what else is there. Most kinds of statistical calculation rest on assumptions about the behavior of the data. Those assumptions may be false, and then the calculations may be misleading. We ought always to try to check whether the assumptions are reasonably correct; and if they are wrong we ought to be able to perceive in what ways they are wrong.

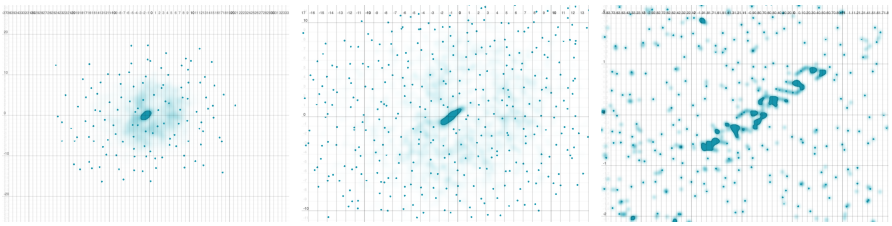
The process Anscombe is referring to, which supports rapid extraction of visual statistics about distributed visual information, is now known as



(a) Scatterplot with alpha = 0.5.



(b) Scatterplot with alpha = 0.1.



(c) Splatterplot.

Figure 2.3: The "Pollen" synthetic dataset at different zoom levels. Overplotted scatterplots hide features in the data overview. Splatterplots (Mayorga and Gleicher, 2013) help to identify dense regions; however, parameter tuning is needed (bandwidth, threshold, and clutter radius).

ensemble coding (Szafir et al., 2016). In information visualization, there are four types of ensemble coding tasks: a) identification (absolute and relative values, outliers); b) summary (mean, variance, distribution statistics, cardinality); c) segmentation; and d) pattern recognition (trend, shape, similarity). Combined, these tasks allow us to construct a more accurate mental model of the data than if we relied solely on a few statistics.

To better understand the role of visualization in the present day it helps to examine the workflows where visualization is a component. Chen and Golan (2016) described in detail six workflows, which they grouped into four blocks according to the role the visualization plays. Disseminative visualizations, often used in news media, are used to communicate information to others. Observational visualizations are used in routine operations, where analysts use it to speedily observe captured data. Analytical visualization is used as part of investigations whose goal is to examine complex relationships between variables, and covers scientific workflows, for instance. And Model-developmental visualization is used to improve existing models, methods, and systems. These workflow groups differ with respect to the kinds of processes they comprise and their order. For instance, in disseminative visualization one understands the data and crafts a message before constructing a visualization that carries that message. Such a human process is not present in observational visualization before image rendering: the data and the tasks are regular enough that no human input is needed in the visual mapping phase.

2.1 POOR STATISTICS AND POOR VISUALIZATIONS

We have seen that visualization allows us to represent more faithfully a dataset than it is possible with descriptive statistics. By representing each point individually we can avoid the problems that are inherent to aggregation. However, this is only true within the capacity limits of a visualization design. Each visualization has a capacity that depends on the display capacity (number of available pixels), use of colours and other visual variables, and layout. The visualization capacity is the entropy of a visualization, determined by the number of datasets that can be represented unambiguously (Chen and Jänicke, 2010). Beyond the capacity of a plot, we start to observe the exact same problem that statistics suffer from in the Anscombe's quartet example. In Figures 2.4 and 2.5, I show examples of poor visualizations that resemble Anscombe's quartet in that the reader cannot trust the message conveyed by the dataset representation. The reason for these failures can vary: sometimes it is the size of the dataset; in other cases the visual marks used in the chart are not robust to noise. I identify three classes of scalability problems.

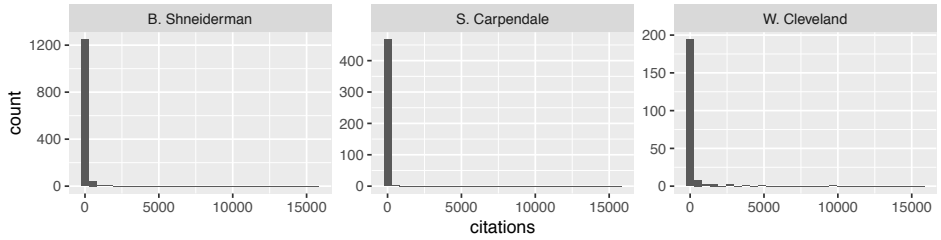


Figure 2.4: A linear histogram fails to depict low-level differences in skewed distributions. Important outliers are represented with little "ink". Data: Citation counts of three information visualization authors (downloaded from Google Scholar on August 23, 2018).

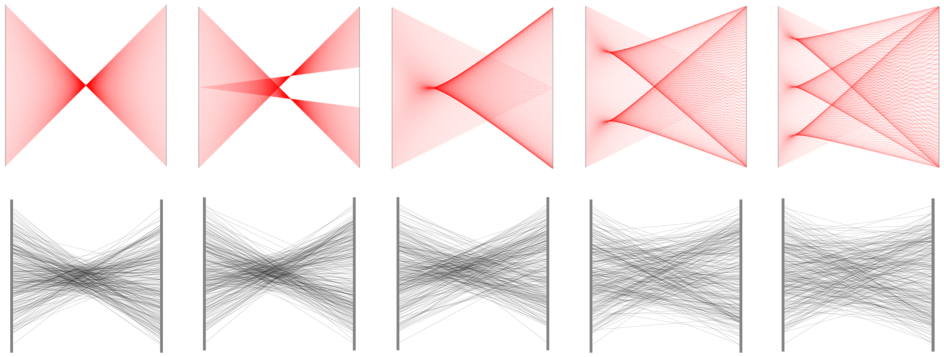


Figure 2.5: Parallel coordinates fail to convey patterns in the presence of noise. Top: Five data relationships produced with mathematical functions. Bottom: The same datasets after the addition of noise at a noise-to-signal ratio of 13%. Example extracted from Johansson et al. (2008).

OVERPLOTTING The most common scalability problem, it occurs when the visualization is rendered ineffective due to the cardinality of the dataset (the number of rows). In an over plotted visualization the structure of a dataset cannot be accurately represented, causing ambiguity. The visual manifestation of this problem can be too much overlap in scatterplots, a sheer amount of edge crossings in node-link plots, or too many tiny elements in a space-filling plot (e.g., a treemap plot).

MULTIDIMENSIONAL Occurs when the number of relevant dimensions (columns) is higher than the number of visual variables a visualization can employ or display effectively. The analyst often needs to "stitch" multiple charts, one for each relevant subset of relevant variables. In Chapter 4, I will present a complex instance of this problem where the task of detecting motion outliers becomes more difficult when multiple visual channels are used in a plot.

DISTRIBUTIONAL Occurs when the dataset lies within a subset of data distributions that are not handled well by the visualization. The common cause of this issue is the presence of outliers, which can render linear scales useless (unless the purpose of the visualization is to detect outliers). A more intricate example is presented by Johansson et al. (2008), who demonstrated that the introduction of Gaussian noise (13% measured as the ratio between the standard deviation of the noise and the range of the variables) in a set of 5 different synthetic, bivariate datasets reduced discriminability of their parallel coordinates representations to 70%, as determined empirically. The datasets featured the following mathematical relationships: negative linear, negative linear with discontinuity, and sinusoidal relationships with one, two, and three periods. In this case, the addition of noise did not disrupt the structure of the data, but the visual structure of the image was affected. Some visual representations are more vulnerable to these kinds of perturbations than others.

2.2 INTERACTION

Information visualization deals with the problem of scale by leveraging interaction. Among the interactive tasks supported in visualization tools are filtering, highlighting, brushing, pan & zoom, and details on demand (Wilkinson, 2006). An interactive control or an interactive gesture in the case of natural user interfaces allows users to express queries that change the view, and should provide near-immediate response. Slow responses (500ms or more) cause users to reduce their activity and cover less data in their analysis, ultimately leading to fewer insights (Liu and Heer, 2014).

Among popular interaction paradigms are the early and prevalent direct manipulation paradigm (Shneiderman, 1983), which was originally realized with tools that feature interactive sliders, and modern paradigms, such as embedded interaction (Saket et al., 2018), free-form sketching (Lee et al., 2013), and visualization by demonstration (Saket et al., 2017). Although all interaction techniques in some way will improve the scalability of a data analysis, there are some tools that notably empower the user to explore vast data spaces efficiently. In the domain of 3D simulation, Bruckner and Möller (2010) proposed scene clustering and "searching by example" as ways to visualize and search for patterns in large parameter spaces. Search by example enables searching for patterns in large collections of scatterplots (Wilkinson and Wills, 2008), and querying specific temporal patterns in large time-series data (Holz and Feiner, 2009).

Interactive visualization has been recently linked with problems related to bias and discovery of spurious patterns. First, by interacting with the data selectively, people may draw conclusions that are affected by known cognitive biases, such as oversensitivity to consistency and the vividness criterion (Wall et al., 2017). This can be remedied by interacting with subsets of that data in a way that covers the entire data space more uniformly. Second, by repeatedly subsetting and querying the data, analysts are more likely to encounter random patterns and mistake them for valid, generalizable ones (Zgraggen et al., 2018). This is formally known as the multiple comparisons problem in statistics.

Interaction is limited as a solution to the problem of scale. Interactive visualization tools have a sliding window nature: one sets the window to the data at a certain position to examine reliably a relatively small subset of the data, and then slides it over the data space in order to achieve good coverage. This is done with operations that constrain the scope of the analysis, such as filtering and zooming, and is informed by an "overview" of the data. While the size of the data space is ever increasing, it is unlikely we will move to larger windows, so the task of analyzing data tends to become more time-consuming and exhaustive. While better interactive techniques for covering large data spaces are needed, better static images and feature extraction are also needed for guiding attention to relevant subsets or to outright eliminate the need to slice and dice the data.

2.3 FEATURE EXTRACTION

Another way of dealing with scale, feature extraction seeks to increase the information content of overviews. It can be based on the information theoretic notion that regularities in the data can be compressed while causing little loss of information (Chen and Jänicke, 2010), or on the Bayesian surprise notion (still linked to information theory) that data that contradicts prior beliefs has more importance (Correll and Heer, 2017). The biggest issue with these models is parameterization: it is often hard to choose parameters that produce useful views, in part because we lack reliable tools to gather users' soft knowledge and goals, which are often necessary to decide what is relevant.

For instance, splatterplots are scatterplot extensions where clutter is controlled with subsampling and dense regions are emphasized with smooth shapes (Mayorga and Gleicher, 2013). This strategy is meant to counteract the "equalization" effect caused by overplotting in scatterplots, whereby relatively sparse regions look just as dense as truly dense regions. Van Goethem et al. (2017) solves essentially the same problem

in time-series by aggregating lines that follow the same trend, similar to edge bundling but depending on parameters that relate to the formal definition of trend in time-series.

In the network visualization domain, Graph Thumbnails are icon-sized visualizations of large graphs that allow large-scale comparisons of graphs using small multiples (Yoghourdjian et al., 2018). The representations focus on the coarse structural characteristics of the graphs. Different than the examples above, which are modifications of existing visual representations, this is a novel visualization method designed from scratch to overcome perceptual scalability issues.

Dimensionality reduction methods have become increasingly popular together with machine learning techniques that represent data points as large vectors. These methods vary with respect to what kind of structure they preserve. Principal component analysis finds a linear reduction of the vectors to maximize the variance; it is generally good at preserving global structures. t-SNE finds a non-linear embedding that tends to capture local structure well (van der Maaten and Hinton, 2008), while UMAP features a parameter that changes the importance of local and global structure preservation (McInnes and Healy, 2018).

2.4 PERCEPTUAL VS COMPUTATIONAL SCALABILITY

When the word *scalability* appears in an information visualization paper, it usually refers to algorithm performance, measured with latency (Liu and Heer, 2014), or more loosely with the number of data points rendered and manageable with interaction (Fekete and Plaisant, 2002; Shneiderman, 2008).

Fekete and Plaisant (2002) presented a tool capable of displaying one million data points in scatterplots and treemaps by leveraging GPU processing; later, Shneiderman (2008) made a call for research efforts to increase the capability of visualization tools to billions of records by exploiting pixel-based representations, density representations, and data aggregation. Recently, a series of techniques based on pre-processed data structures and aggregated plots—Nanocubes (Lins et al., 2013), Hashedcubes (Pahins et al., 2017), and Gaussian cubes (Wang et al., 2017)—allowed visualization and low-latency interaction with hundreds of millions of spatio-temporal records, such as tweets, flights, social media checkins, and taxi rides. Similarly, by pre-computing data subsets and using binned plots, imMens lets users scale their visual analyses from 1 million to 1 billion records with nearly no difference in interaction latency (Liu et al., 2013).

On the perceptual scalability side, techniques were developed to improve some plots or were developed from scratch to be more scalable than the alternative methods. Unlike the works I mentioned under (Section 2.3), these do not involve any statistical or domain-specific judgements of data importance, they are purely representational solutions. For instance, despite the questionable aesthetics, Cushion Treemaps (van Wijk and van de Wetering, 1999) tries to improve the perception of structure in crowded treemaps, and benefited from large adoption by developers of file-system visualization tools (Disk Inventory X, SequoiaView, WinDirStat, GrandPerspective, OmniDiskSweeper, etc.), which can well be considered a statement of its effectiveness.

A large class of recent systems attempts to solve what I referred to as multidimensional scalability problem by embedding subplots that depict additional dimensions (Alsallakh et al., 2012; Krzywinski et al., 2009; Loorak et al., 2017). However, while solving that problem these interfaces may be hurting perceptual scalability since, by design, visual marks are added to the visualization causing increase in clutter. While it is currently difficult to assert this with confidence, as the community lacks a standard method to test perceptual scalability (see next section), the evidently crowded displays produced with these techniques suggest that perceptual scalability is at risk (Figure 2.6).

2.5 QUALITY MEASURES

In a recent survey, Behrisch et al. (2018) divide quality measures for information visualization in three levels: low-level (perception), mid-level (perception/task), and high-level (meta-perception/user). Most existing measures fall into the mid-level category. Mid-level measures rely on a specification of the task; that is, their goal is not to score the absolute quality of an encoding, instead, they seek to score the quality of a *single view* given the data, a task, and sometimes a visual pattern. For instance, if the task is *finding clusters*, a metric may rank views based on the presence and discriminability of groups (Albuquerque et al., 2010; Tatu et al., 2011). These measures are then often used to *search* for views that best feature the pattern.

The vast majority of these measures are designed specifically for one visualization type, and only a small number of approaches have been evaluated against empirical effectiveness data (Behrisch et al., 2018). For instance, based on studies that show that aspect ratio of scatterplots can influence the accuracy of correlation perception, Fink et al. (2013) found that a pair of measurements extracted from the Delaunay triangulation of the scatterplots correlates reasonably with user preferences. A significant

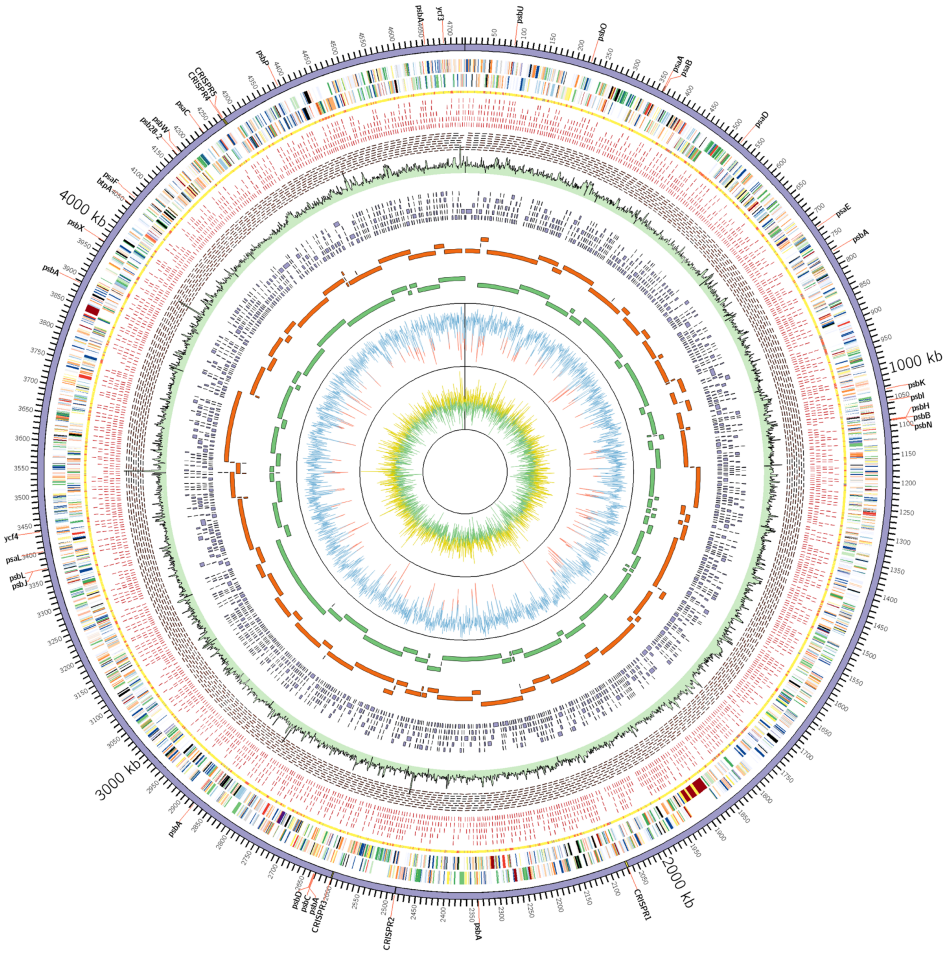


Figure 2.6: Circos visualization of genome sequencing. The effectiveness of hyperdense visualizations like this is questionable. Example extracted from Saw et al. (2013), licensed under Creative Commons.

challenge for task-specific quality measures is that users tend to follow an unstructured exploration path where *multiple* tasks are performed in parallel; thus, a single task-specific measure is not sufficient.

Low-level measures do not assume a specific task, and focus exclusively on the visual mapping. These measures exist mostly in the theoretical realm, with the notable exception of clutter measures and information theoretic measures, both of which have been translated into computational measures. Clutter measures exist in general form, stemming from vision science research (Rosenholtz et al., 2010), and encoding-specific form for scatterplots (Bertini and Santucci, 2004) and parallel-coordinate plots (Ellis and Dix, 2006). In the same vein, graph readability criteria are used in the calculation of layouts and can involve measuring edge crossings, and overlaps between nodes and groups (Dunne et al., 2015; Purchase, 2002). While Behrisch et al. classifies clutter reduction approaches as mid-level, I choose to label them as low-level for they do not carry considerations of task and are thus more general. On the information theory side, Chen and Jänicke (2010) derived visualization capacity measures from classic information theory concepts (e.g., entropy, mutual information); later, Chen and Golan (2016), applying notions of data compression to the pipeline model of visualization, proposed general cost and benefit measures.

Finally, high-level measures attempt to quantify properties that are traditionally perceived as subjective, such as memorability, aesthetics, and engagement. So far, this topic has only been explored with user studies and experiments, and no measures for visualization have been proposed. An interesting aspect of measures at all levels is that most are not presented as quality measures; instead, they are embedded into automated techniques for optimizing visualizations. As a result, they are rarely used to *evaluate* new visualization encodings.

2.6 SUMMARY

This chapter introduced the problem of scalability in information visualization. The main points are:

- As the parameters of the data grow, visualization begins to suffer from the same faithfulness problems that affect summary statistics, and that justify visualization as a complement to statistics in data analysis.
- Scalability problems in visualization stem not only from dataset size, but also from dataset distribution, and multidimensionality.

- Interaction can dramatically improve the scalability of a visual data analysis, but is ultimately limited by human energy and time.
- Feature extraction is used to increase the scalability of static views, with the drawback of needing careful parameterization.
- Quality measures for visual encodings are mostly specific to a visualization type and make assumptions about the analytical task. There are few empirically validated general measures that can be used for evaluation.

3 CLUTTER AND INFORMATION

For many years, the information visualization community followed Ben Shneiderman's celebrated visual information-seeking "mantra" for design: "overview first, zoom and filter, details on demand" (Shneiderman, 1996). However, as datasets have grown (and small displays have become more prevalent), "overview first" is increasingly challenging to achieve in an effective way. Overviews of very large datasets are often too high-level or cluttered to reveal anything interesting. The task of iterative exploration and sifting through the data is left to the analyst in the traditional model. This chapter introduces a method for optimizing large hierarchical visualizations to fit in constrained screen spaces, effectively creating starting point overviews that are designed to balance the goal of maximum information content with the challenge of reducing clutter and enhancing readability. The work is inspired by Keim's visual analytics process, which states: "analyze first, show the important" (Keim et al., 2006). The critical "analyze first" step is addressed to shape the initial view of the data, so as to reveal important data entities while minimizing clutter, harnessing computing power to create data-driven starting points for analysis. The display-optimized tree cut model I present is parameterized to allow for interactive drill down, as well as presentation of optimized overviews of data.

In addition to the challenge of providing optimized overviews for very large datasets, in many situations, visualizations need to be adaptable to a variety of screen sizes. For example, consider an interactive visualization embedded as part of an online news story — one version may be appropriate for a smart phone display, while another will be appropriate for a large monitor. The situation is not as simple as changing the zoom factor, or the flow of the webpage, but rather the *level of abstraction* must adjust to make the visualization readable and aesthetically pleasing across devices.

Many factors influence the ability of visualization systems to effectively display large amounts of data; in particular, the available display size, which is determined by the physical constraints of the screen, and the perceptual scalability of the visualization, which depends on the choice of visual representation and layout (Yost and North, 2006). Most information visualizations become over-cluttered when the dataset is large. Clutter reduction is an active area of research in information visualiza-

tion, as elaborated by Ellis and Dix in their taxonomy of clutter reduction methods (Ellis and Dix, 2007). Clutter is shown to have a negative impact on visual search (Haroz and Whitney, 2012; Rosenholtz et al., 2010; Wolfe, 1998a) and short term memory (Miller, 1956). In a study of orientation judgment, Baldassi et al. (2006) found that clutter causes an increase in orientation judgment errors, and increase in perceived signal strength and decision confidence on erroneous trials. Rosenholtz et al. (2007) include the notion of performance in the very definition of clutter: "a state in which excess items, or their representation or organization, lead to degradation of performance at some task". Besides, in some resource-constrained client environments (e.g., web browser), the number of graphic primitives necessary to represent large data affects rendering and, consequently, interactive tasks, such as selection and filtering.

In visualizations of hierarchical data, one can take advantage of the hierarchical structure to abstract data at varying levels, in order to reduce the level of clutter when the available space prevents depiction of the full data. Visualizations that implement such strategy are called multiscale visualizations (Elmqvist and Fekete, 2010) and deciding the appropriate level of abstraction for them is not trivial. Overly-detailed views have high clutter, whereas overly-abstract views can hide important patterns. The right level of abstraction depends on the dataset and the available display space; for example, large desktop displays afford more detail, while mobile phones have not only less space, but also coarser interaction resolution due to the "fat finger" problem. In this chapter, I refer to this problem as the *level of abstraction* problem.

The display-optimized MDL tree cut technique that I will present in this chapter can be applied to any hierarchical dataset where there are quantitative data values associated with the leaves of the tree. In the next sections I will introduce the mathematical foundation behind the general display-optimized tree cut, and demonstrate the approach applied to two popular hierarchical visualization types — treemap and sunburst. I will also report on multiple validation approaches: a crowdsourced study whose results indicate that the tree cut approach provides for faster target finding compared to traditional approaches, and a quantitative comparison of clutter and information content across traditional techniques and the display-optimized MDL treemaps.

3.1 RELATED WORK

In this section, I survey two areas: techniques for controlling clutter in visualizations using aggregation and the use of tree cuts (also known as antichains) to navigate large graph hierarchies.

Clutter Control

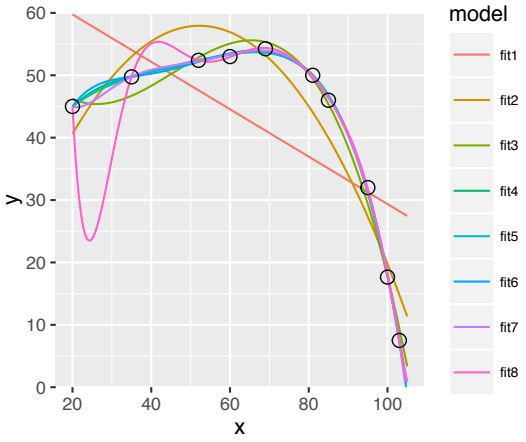
Based on the cartographic principle of constant information density (Töpfer and Pillewizer, 1966), VIDA is a system that automatically creates visualizations in which density remains constant across zoom levels (z dimension) and within each view (x and y dimensions) (Woodruff et al., 1998). The display is divided into regions, where the visual representation is modified (e.g., dots instead of glyphs) to meet a target density value specified by the user. Density measures are number of objects and number of vertices per unit of display area.

ViSizer is a framework for resizing visualizations (Wu et al., 2013). It employs a sophisticated image warping technique that scales important regions uniformly and deforms less important regions. The significance measure is composed of the feature congestion clutter measure (Rosenholtz et al., 2010) and a degree of interest (DOI) function. ViSizer focuses on non-space filling visualizations such as word clouds and scatterplots.

Chuah (1998) employs a simple strategy for automatic aggregation in histograms, and ordered radial and treemap visualizations: aggregate neighboring objects whenever there is occlusion or they are too small to be perceived. This approach works better where data items have an intuitive order (e.g., time series, histograms, or file directories ordered by name). Cui et al. (2006) tackled the optimal level of abstraction problem, but focusing only on accuracy; that is, how well the abstracted data represents the original dataset. They proposed two measures of quality: the histogram difference measure and the nearest neighbor measure, which were integrated into XmdvTool. As the measures do not account for the visual quality of the resulting visualization, the user determines the best view interactively, by tweaking the level of detail and comparing the quality measure values. Likewise, based on aggregation quality measures, Andrienko and Andrienko (2010) allow users to specify the desired level of abstraction in visualizations of movement data (flow maps).

Koutra et al. (2015) proposed a parameter-free method based on the minimum description length to select the best (most succinct) summary for large graphs among a set of alternatives: cliques, stars, chains, and bipartite cores.

Perhaps the closest to this work, Lamarche-Perrin et al. (2014, 2012) introduce a method for selecting abstract representations of hierarchical datasets. In their work, a two-part information criteria consisting of entropy and Kullback-Leibler divergence is used to select the tree cut featuring the best balance between conciseness and accuracy. Their procedure requires tuning a free weighting parameter that specifies the relative importance of one criterion over the other. It does not account for the available display space, so any adjustments to accommodate small



	$\sum L(\hat{a}_k)$	$\sum L(r_i\|\hat{\theta})$	**total**
1	128	640	768
2	192	640	832
3	256	512	768
4	**320**	**128**	**448**
5	384	192	576
6	448	192	640
7	512	128	640
8	576	0	576

Figure 3.1: On the left, a series of polynomials ranging from order 1 to 8 fitted to a 10-point data set. On the right, the cost of encoding (in bits) the two parts of each polynomial model.

or big screens need to be done manually by tuning the aforementioned weighting parameter.

Tree Cuts or Antichains

Tree cuts, also known as antichains, have been widely used in the exploration of large graphs and hierarchies. SentireCrowds (Brew et al., 2011) and ThemeCrowds (Archambault and Greene, 2011) employ a maximal antichain selection method to abstract a hierarchy of topics visualized as a treemap. That method is based on matching node scores resulting from user queries. GrouseFlocks (Archambault et al., 2008) reduces the complexity of interacting with large graphs by letting users manipulate cuts of superimposed aggregate hierarchies. Users can adjust the cut level of abstraction by performing topology-preserving operations involving merging and deletion of aggregates. In order to ensure the abstracted hierarchy view remains under the display capacity, ASK-GraphView (Abello et al., 2006) parametrizes clustering with maximum antichain size. In ASK-GraphView and GrouseFlocks the hierarchies are not part of the data, but created by an algorithm. This allows great flexibility to modify the hierarchy structure around display constraints. In this work, I focus on "rigid" hierarchies, where classes carry domain specific relevance and, thus, cannot be merged or deleted without cost to interpretation.

3.2 THEORETICAL FOUNDATIONS

Suppose a set of measurements $D = (x_1, y_1), .., (x_n, y_n)$ was collected as part of an experiment and we were asked to send this data over a network where the transmission cost is high. Among the countless possible ways of encoding the data, it is in our best interest choosing a scheme that allows for the shortest message. In this scenario, the code length for sending the *raw* data, assuming that encoding a number has a fixed cost of b bits is:

$$L(D) = \sum_{i=1}^{n} \Big\{ L(x_i) + L(y_i) \Big\} = 2nb. \tag{3.1}$$

If the relation between x and y can be described by a polynomial model (or any other model), it might be possible to reduce significantly the code length. As an example, let's examine the polynomial case. A polynomial regression model has the following form:

$$\hat{y} = \sum_{k=0}^{p} \hat{a}_k x^k + \epsilon. \tag{3.2}$$

So the code length of the data as seen through a fitted polynomial model $\hat{\theta}$ is:

$$L(\hat{\theta}, D) = \sum_{i=1}^{n} L(x_i) + \sum_{k=0}^{p} L(\hat{a}_k) + \sum_{i=1}^{n} L(r_i \mid \hat{\theta}), \tag{3.3}$$

where $\hat{a}_k$ is the k-th parameter of the polynomial and r_i is the i-th residual. Namely, the equation above is a sum of the cost of encoding the model and the cost of encoding the data conditioned on the model (residuals). Note that the cost of sending the vector $\vec{x}$ is constant across all models. As a polynomial might not fit the data perfectly, it is necessary to send the model residuals, so that the receiver is able to reconstruct D accurately. However, depending on our tolerance to errors, we might be willing to ignore residuals smaller than a fixed threshold. The better the fit, the more economical is the description of the residuals. Overall, it is only worth representing our data with a polynomial model if we can find a model whose code length overhead is smaller than the code length of vector $\vec{y}$:

$$\sum L(y_i) > \sum L(\hat{a}_k) + \sum L(r_i|\hat{\theta}). \tag{3.4}$$

To illustrate this notion, consider the ten data points depicted in Figure 3.1, left. I fitted to this data a family of polynomials of increasing order

and compared the cost of representing the data with each of them in a setting where any number is represented with 64 bits, and residuals smaller than 0.5 are ignored. Figure 3.1, right, shows the cost of each fitted polynomial from order 1 to 8. It is clear that the more parameters a model has, the better is its fit. However, the model that provides the shortest description is that featuring the best balance between goodness of fit and complexity. In our example, this model is the 4th order polynomial, which also satisfies (3.4), as the cost of encoding y in the naive scheme is 640 bits.

In this example, I used information theoretic reasoning to determine the model that most concisely captures the regularities in the data. The criterion I employed is a simplification of the Minimum Description Length (MDL) Principle, which I describe formally in the following subsection. MDL is a powerful approach to model selection that has been used to solve a large variety of problems, including polynomial regression, Gaussian density mixtures and Fourier series regression (Lee, 1999), and applied problems such as image segmentation (Lee, 2001), learning word association norms (Li and Abe, 1998) and learning decision trees (Quinlan and Rivest, 1989).

Minimum Description Length

Proposed by Rissanen, MDL is an information criterion used for model selection in statistics (Rissanen, 1983). The principle is based on the following notion: given a set of observed data and a family of fitted models, the best model should provide the shortest encoding of the data. The description length of a model is calculated as a sum of two parts: the length of the binary codes that describe a) the model parameters, and b) the data residuals (Lee, 2001). More formally, the MDL criterion can be written as:

$$L(\hat{\theta}, \vec{x}) = L(\hat{\theta}) + L(\vec{x} \mid \hat{\theta}), \tag{3.5}$$

where $\hat{\theta}$ is a parameter vector, $\vec{x}$ is the data, and $L(\hat{\theta})$ and $L(\vec{x} \mid \hat{\theta})$ are the parameter description length (a) and the data description length (b), respectively.

Unlike in the polynomial example, where we used computer-oriented calculations for the code length, MDL is concerned with *optimal* code length. That is, with MDL, we do not care about how a model is encoded in practice as much as we care about how concisely it can be encoded in theory. Let A be an alphabet and α be any of the symbols in A. If the

probability $p(\alpha)$ of occurrence of $\alpha \in A$ is known, then in the optimal encoding scheme for A the length of α is:

$$L_{OPT}(\alpha) = -\log_2 p(\alpha). \tag{3.6}$$

This result is important because often the likelihood function of the model $\hat{\theta}$ is known, so the data description length (number of bits to encode the residuals) follows from (3.6):

$$L(\vec{x} \mid \hat{\theta}) = -\log_2 p(\vec{x} \mid \hat{\theta}). \tag{3.7}$$

For instance, in our polynomial example we could leverage the fact that, as per the regression model assumption, the residuals are approximately normally distributed, and use the log of the Gaussian likelihood, given by $(n/2)log_2(RSS/n)$, as $L(\vec{x} \mid \hat{\theta})$, where RSS is the residual sum of squares.

Frequently, the probability distribution of the model parameters (usually, a vector of integer or real numbers) is not given; in this case, Rissanen (1983) proposes a *universal prior* probability distribution or, equivalently, a coding system, for integers. Rissanen demonstrated that the optimal code length for such integers with unknown probability function can be achieved with his coding system and approximated to $log_2 n$. Therefore, we can estimate the description length of arbitrarily complex models, as long as their parameters can be described as arrays of integers or real numbers.

Let's assume $\hat{\theta}$ is a vector of real numbers, which can be encoded by representing the integer and fractional parts separately. The fractional part needs to be truncated to a pre-defined binary precision ρ, since the binary representation of many numbers can be infinite. Thus, the number of bits to encode $\hat{\theta}$ is:

$$L(\hat{\theta}) = \sum_{i=1}^{k} \log_2 \lfloor \hat{\theta}_i \rfloor + k\rho, \tag{3.8}$$

where k is the number of parameters in the model.

Note that the choice of the precision ρ is of major importance. Choosing fewer bits to encode the fractional parts yields a small $L(\hat{\theta})$, but at the expense of $L(\vec{x} \mid \hat{\theta})$, as the residuals will be larger. A finer precision reduces the residuals, as the encoded values will be closer to the true estimates, but increases the cost of encoding the parameters. In order to minimize the description length, we need to optimize the precision. Rissanen (1989) shows that if the model parameters are estimated from n data points us-

ing Maximum Likelihood Estimation (MLE) and n is large, the optimal precision ρ is approximately $(\log_2 n)/2$. Thus, (3.8) can be rewritten as:

$$L(\hat{\theta}) = \sum_{i=1}^{k} \log_2 \lfloor \hat{\theta}_i \rfloor + \frac{k}{2} \log_2 n. \tag{3.9}$$

With the expressions for data and parameter description length, (3.5) can be written in more detail as:

$$L(\hat{\theta}, \vec{x}) = \sum_{i=1}^{k} \log_2 \lfloor \hat{\theta}_i \rfloor + \frac{k}{2} \log_2 n - \log_2 p(\vec{x} \mid \hat{\theta}). \tag{3.10}$$

Equation 3.10 embodies the fundamental trade-off between conciseness and accuracy that defines the MDL principle. Models with more parameters will achieve better accuracy (high likelihood) at the expense of simplicity. In fact, if we set $L(\hat{\theta})$ constant, MDL falls back to MLE, selecting the model that offers the best fit to the data. In that sense, $L(\hat{\theta})$ can be thought of as a safeguard against over-fitting. Likewise, over-concise models have low accuracy, being just as undesirable. Minimization of the description length tends to select the model featuring the best balance between these criteria. In the information theoretic interpretation, the selected model corresponds to the best compression of the data.

MDL Tree Cut Model

Having laid out the general formulation of the MDL principle, in this section I explain the tree cut model, which is an important building block for our abstraction approach. The tree cut model is a generalization method based on MDL, originally developed for the linguistic problem of automatic acquisition of case frame patterns from large corpora (Li and Abe, 1998).

Consider a tree structure representing the hierarchical relation between abstract classes, e.g., IS-A, part of, instance of. The degree of abstraction grows towards the root. Assume that only the leaves are observable (countable), and the internal nodes accumulate the counts of their children. L is the set of all leaves. A dataset S is a multiset of observations, each representing one occurrence of a leaf $l \in L$, with $l \in S$ denoting the inclusion of l in S as a multiset. We denote the dataset size by $|S|$, the total number of observations.

A tree cut is any set of tree nodes that exhaustively covers the leaf nodes. Graphically, it can be represented by a path crossing the tree lengthwise, as in Figure 3.2. Nodes along the cut represent the subtrees dominated by them and are assigned each a probability value. Depending on how regular is our data S, a concise way to transmit it over an arbitrary

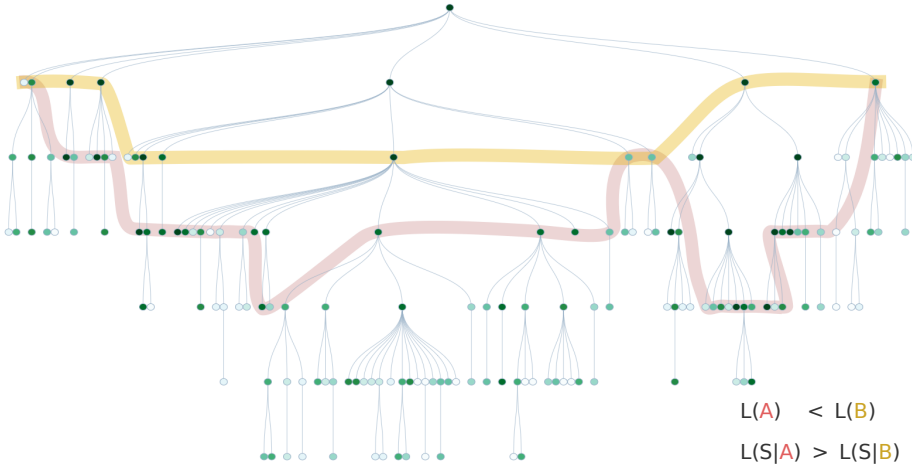


Figure 3.2: An illustration of two tree cuts: A (yellow) and B (pink). More abstract cuts (A) have lower parameter description length ($L(A)$), but higher data description length ($L(S|A)$).

channel to a receiver who has knowledge of the tree is to send a tree cut. The receiver then estimates the value of each leaf based on the value of the node representing it in the cut. In other words, a cut is a model of the data, carrying estimates of the observed values. The residuals are sent separately, in the MDL fashion, as discussed in Section 3.2.

A tree cut model M is defined as the tuple $(\Gamma, \hat{\theta})$, where $\Gamma = [C_1, C_2, ..., C_k]$ and $\hat{\theta} = [\hat{P}(C_1), \hat{P}(C_2), ..., \hat{P}(C_k)]$: the vector of nodes (classes) and their parameters (estimated probabilities), respectively. The probability $\hat{P}(C)$ of a class is estimated by MLE, as follows:

$$\hat{P}(C) = \frac{f(C)}{|S|}, \tag{3.11}$$

where $f(C)$ is the accumulated count of the class C. The estimated probability of each of the leaves under a class is obtained by normalization of the class probability over the number of leaves C under the class:

$$\hat{P}(l) = \frac{\hat{P}(C)}{|C|}. \tag{3.12}$$

Note that behind this formula is the assumption of uniform probability. This means the probabilities (or frequencies) of the leaves under a cut are smoothed.

As discussed in the previous section, the data description length is the log of the likelihood of the data:

$$L(S \mid \Gamma, \hat{\theta}) = -\sum_{l \in S} \log_2 \hat{P}(l). \tag{3.13}$$

The minimum data description length is held by the deepest tree cut model, comprised of all leaves, which features no better abstraction than the raw data. The cost of encoding the parameters $\hat{\theta}$ of the model, an array of real numbers, is given by (3.9). Note that, Li & Abe omit the first term in (3.9), namely, the cost of encoding the integer part of the parameters, because the model parameters are probabilities; hence, the cost of encoding the integer parts is always 0. In summary, Li and Abe's tree cut model minimizes the following information criterion:

$$L(\hat{\theta}, S) = \frac{k}{2} \log_2 |S| - \sum_{l \in S} \log_2 \hat{P}(l). \tag{3.14}$$

To be more precise, in addition to the probabilities $\hat{\theta}$, a receiver would also need to know the classes Γ to decode the data correctly. Since the number of possible cuts in a tree is finite, in theory we could use an index to inform Γ, as part of the coding scheme. As such indexes would be equally probable a priori, their code length would be constant for all models and so, can be safely ignored. For the purpose of model selection, all we need to account for is the cost of encoding $\hat{\theta}$ and $(S \mid \hat{\theta})$.

Li and Abe (1998) provided an efficient, greedy algorithm that is guaranteed to find the tree cut whose description length is minimal (Listing 3.1). The algorithm is based on the following insight: for each tree cut segment, the description length calculation depends only on the subtrees covered by it. Therefore, the best tree cut is either the root tree cut or the concatenation of the best tree cuts for each of the child subtrees. This algorithm is guaranteed to find the minimum description length regardless of changes in the description length calculation so long as these changes do not alter the independence between subtree cuts. In the rest of this chapter, I present different ways to calculate parameter and data description lengths, but the same algorithm is used for minimization.

3.3 MDL DRILL-DOWN

In this section, I experiment with using the tree cuts selected by Li and Abe's approach to inform which nodes should be abstracted in views of a hierarchical dataset. Since such cuts are generated with no consideration of the available display size, I adapt the method by introducing a weighting parameter that determines the relative importance of fitness to the data over clutter. An increase in weight results in a deeper tree cut. In my proof-of-concept, the user can manipulate this parameter interactively to increase the level of detail of the view.

Listing 3.1: Find-MDL. For each child subtree recursively finds the best treecut. The child treecuts are appended and the resulting description length is compared to that of the root treecut, which consists of a single node, the root. Whichever holds the lowest description length is returned.

```
1
2  def find_MDL(t):
3      '''
4      Recursively finds the best treecut.
5      Args:
6          t - a tree
7      Return:
8          the MDL treecut for t
9      '''
10
11      if is_leaf_node(t):
12          return [t]
13      else:
14          c = []
15
16      for child in t.children:
17          treecut_child = find_MDL(child)
18          c.append(treecut_child)
19
20      # L calculates the description length
21      if L([t.root]) < L(c):
22          return [t.root]
23      else:
24          return c
```

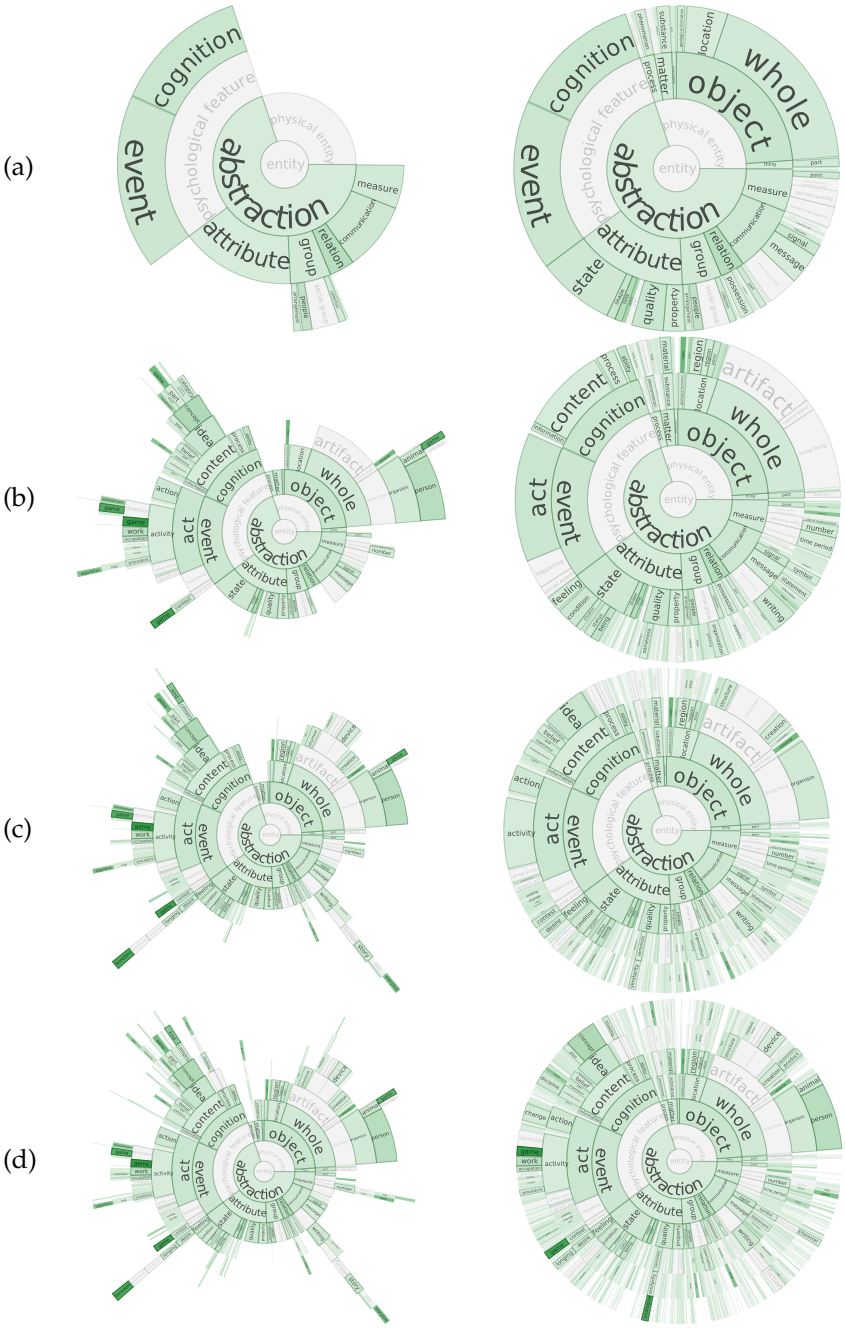


Figure 3.3: Display-optimized MDL tree cuts (left) reveal important nodes while reducing clutter when compared to simple depth thresholding (right). These are Docuburst views of the book *Gamer Theory*, revealing occurrences of concepts in the book using color. Concepts are organized into a semantic hierarchy. From top to bottom increasing numbers of nodes are revealed through adjusting the tree cut parameter (left) or increasing the depth threshold (right). We can see that in (b, left) several important dark-green nodes are revealed while unimportant nodes remain hidden. Figure (b, right) has twice as many nodes, but important nodes remain hidden. These nodes are not revealed by the simple depth threshold until (d, right), where there are a significant number of unimportant nodes also visible.

I chose to implement the technique on Docuburst, an open-source document visualization tool (Collins et al., 2009a). Docuburst displays a sunburst representation of the WordNet ontology where the size of nodes and categories (angular extent) is weighted by their occurrence in the input document, allowing users to inspect which words and categories of words are more prevalent in a document. The color of a node is based on the non-cumulative count of uses of the corresponding word in the document. In their future work section, Collins et al. discuss two problems that could potentially be solved with uneven MDL tree cuts. First, abstracting subtrees that have low relative importance. Second, the top levels of WordNet are too abstract, as far as carrying little information about the document's content.

Figure 3.3 features views of the book *Gamer Theory*, by McKenzie Wark. The most representative categories of the document are the darkest (most frequent); for instance: game, entertainment, algorithm, storyline, boredom, etc. In a full tree view, 6,302 nodes would be rendered, which is likely enough to cause latency in a browser-based visualization. Also, displaying this many nodes results in small, illegible labels and the need to interactively zoom and pan.

The tree cut resulting from minimizing Li and Abe's information criterion is shown in Figure 3.3(a). Nodes under the tree cut are hidden, whereas nodes on or above the tree cut are visible. Unless the available display size is limited, that view can be considered too abstract.

Following Wagner (2000), we introduce a free weighting parameter W to equation (3.14) as a means to control the importance of the data description length over the parameter description length and, as a result, the tree cut depth:

$$L(\hat{\theta}, S) = \frac{k}{2} \log_2 |S| - W \sum_{l \in S} \log_2 \hat{P}(l) \quad (W > 0). \tag{3.15}$$

The semantics of increasing W is equivalent to that of drilling down; the more weight applied to the data description length (residuals), the more parameters (nodes) will be included in the model (tree cut) to minimize the overall description length. Thus, weighted MDL tree cuts can be useful to reveal details at a rate that is more compatible with the distribution of values in the hierarchy. In order to illustrate this concept, we mapped W to the drill-down action in Docuburst; that is, when users roll the mouse wheel, W is incremented/decremented by a predefined delta. Figure 3.3(b-d), *left*, shows the result of three subsequent increments in W, starting from 3.3(a), *left*. In contrast, Figure 3.3(b-d), *right*, shows the result of three drill-down steps where a conventional depth threshold is incremented. It is clear that, in only a few steps, weighted MDL views

allow access to most of the representative nodes in the document with much less clutter than using the depth threshold or the full overview. In terms of number of nodes rendered (a-d), the weighted MDL views cost 32, 387, 730, and 887 nodes; while the depth threshold views cost 183, 808, 2202, 4199 nodes.

An important concern is choosing ΔW so that every increment results in a view that has significantly more information than the previous. In my tests, ΔW was defined empirically, and a value of 250 yielded good results for visualizing a variety of documents. Since the amount of information and the number of tree nodes increase monotonically with W, ΔW could be determined dynamically with the definition of a minimum number of tree nodes to enter the view. Then a standard optimization algorithm, such as Nelder-Mead (Olsson and Nelson, 1975), could be used to find the smallest increment to W that satisfies this minimum. Alternatively, ΔW could be based on a model of user's interest.

Weighted MDL views can be useful as a way to explore visualizations interactively, but the problem of optimizing the level of detail as a function of the available display space *before any user input* remained unsolved. Specifically, we needed a method capable of generating a *first* view of the dataset that is as informative as possible within the bounds of readability. The next section presents a satisfactory method.

3.4 DISPLAY-TAILORED TREE CUT MODELS

This section begins with the consideration that hierarchical visualization concerns, in general, the representation of tree cut models, in the sense defined in Section 3.2. If we treat visualization techniques (e.g., treemap, sunburst) as coding schemes and the views produced with them as encoded tree cut models, we can select optimal views using MDL criteria. In particular, we are interested in expressing parameter and data description lengths in a way that relates to clutter and fitness in visualizations. We will focus on space-filling hierarchical visualization techniques, as the connection to MDL is more obvious.

In a space-filling visual representation of a hierarchical dataset, the pixel grid is divided into areas proportional to the data values. Areas are recursively grouped in the visual space according to the hierarchy topology, so that siblings are always adjacent. In addition, color and labels can be used to convey the hierarchical structure.

A non-aggregated hierarchical visualization V_{max} is an encoding of the *deepest* tree cut model of a dataset. For example, in a treemap without decorations (e.g., padding), V_{max} fills the entire display space with rectangles representing the tree leaves. Given the dataset S and the set Λ of

visualizations of S using a specific layout, each of which corresponds to a tree cut of S, V_{max} is the visualization that maximizes $L(V)$:

$$V_{max} = \arg\max_{V \in \Lambda}(L(V)), \tag{3.16}$$

Note that, for sake of simplicity, we make no distinction in the notation between a visualization V and the tree cut encoded by it.

In the information theoretic interpretation, if visualizations allowed for lossless coding, V_{max} would always minimize $L(S \mid V)$ and provide the best fit to the data, corresponding to the model selected by MDL when we set $L(V)$ constant or, equivalently, to the model selected by MLE. However, a space-filling visualization is a *partial* and *lossy* coding system: partial because there exist some source symbols that cannot be encoded (e.g., data points that map to subpixel areas); lossy because it is possible that a pair of symbols share a code word (e.g., data points that map to overlapping areas due to rounding).

Depending on the available display space, when the dataset is relatively small, V_{max} generally provides the best fit to the data, but when the number of leaves is large, decoding of information is impacted, due to the aforementioned limitations caused by display pixel resolution. This is a key departure from Li and Abe's method, where an increase in the length of the model always yields an increase in fitness. In other words, there is a limit on the model fitness to data achievable by a space-filling visualization. This constrain results from limited pixel availability and from limitations in visual acuity. The fact that V_{max} does not necessarily hold the minimum data description length can be denoted as follows:

$$L(S \mid V_{max}) \geq \min_{V \in \Lambda} L(S \mid V), \tag{3.17}$$

This inequality can be read as: the data description length of the visualization of the deepest tree cut (V_{max}) is not necessarily minimal. As a result, before even considering the parameter description length, we can observe that it pays off selecting treemaps more abstract than V_{max} when datasets are large relative to the available screen size.

Treemap

Before I introduce the calculations for the treemap, recall that the fitness to data is a function of an estimated probability and a true value (i.e., the data). The fitness to data is degraded the more the estimation deviates from the true value, in terms of likelihood. Assume that in visualization models the estimated probability is a function of a quantity estimated visually; for instance, the area or the position of a polygon. Thus, by

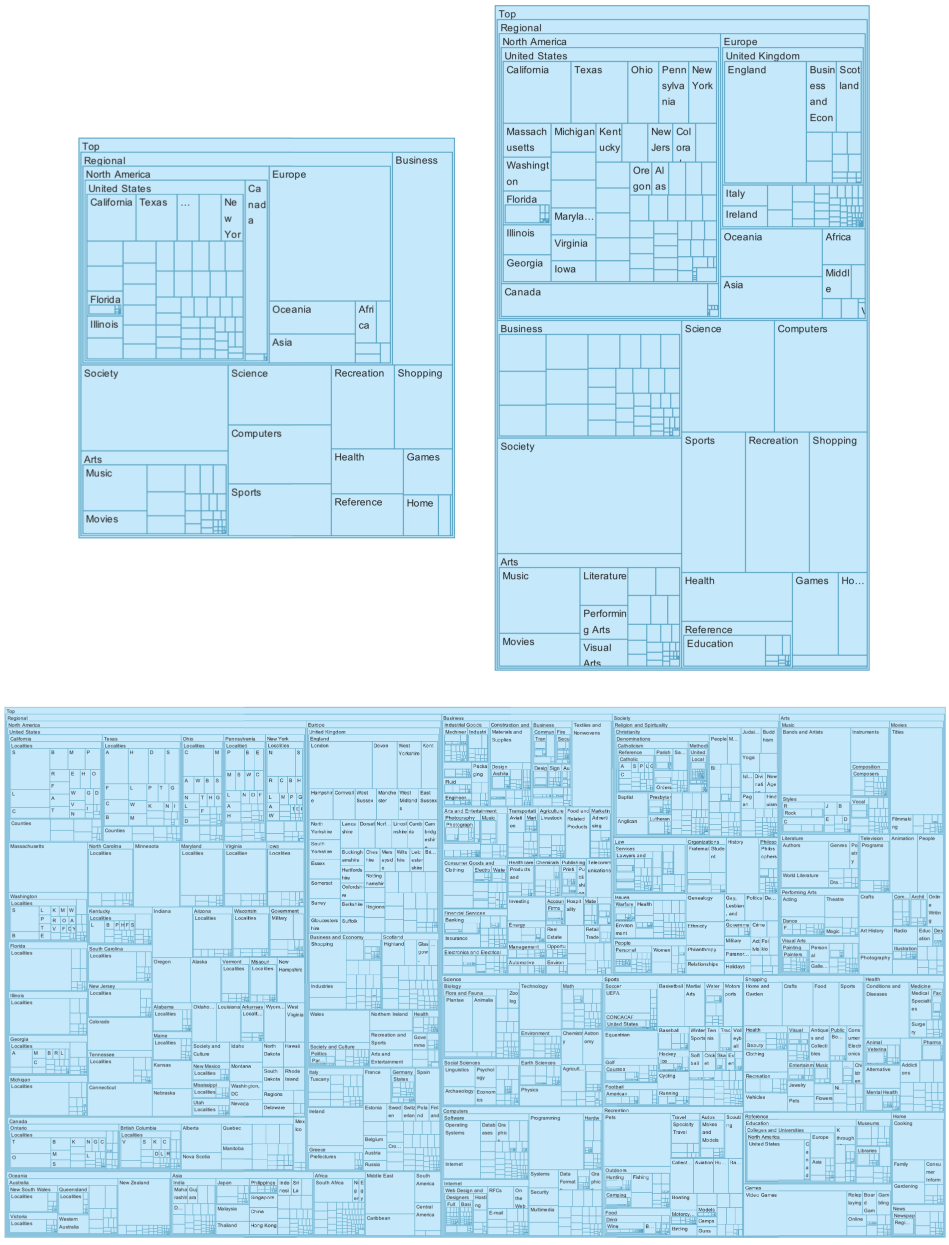


Figure 3.4: Treemap visualizations generated with the display-tailored MDL procedure, with the following resolutions: 375x400px, 375x667px and 1920x1080px. More abstract tree cuts are selected for smaller displays.

expressing mathematically any problems in this visual estimation, we can directly affect the fitness to data. My goal here is to define this estimation problem in a way that reflects the degradation of fitness when the display shrinks.

Let's define the dataset S in more detail. S is a 2-tuple (L, f), where L is the subset of classes that are tree leaves, and f is a function such that for each $l \in L$, $f(l)$ is the count of l.

Then the area of a leaf can be defined as the following composite function with respect to the display area D (in pixels):

$$(A \circ f)(l) = A(f(l)) = \frac{f(l)}{|S|} D. \tag{3.18}$$

Likewise, the area of an abstract class C is given by:

$$A(f(C)) = \sum_{l \in C} A(f(l)), \tag{3.19}$$

where $l \in C$ is the set of tree leaves dominated by C. We call $G = (L, A \circ f)$ the linearly transformed dataset using $A \circ f$. Essentially, G is the dataset with scores transformed to pixels. The probabilities of the classes are estimated based on the encoded G. For conciseness, we abbreviate $A(f(C))$ as $A(C)$ in the rest of this section.

A treemap encodes such areas as a vector of rectangle coordinates $\vec{R} = [R_1, R_2, ..., R_k]$. Formally, we describe a treemap as a 2-tuple $\hat{T} = (\Gamma, (R \mid D))$. Given D, we can refer to any point in the grid with an integer index $1 \leq i \leq D$. Thus, to transmit R_i, we need only two integers, corresponding to the indexes of the top left and bottom right corners. Since the index space is finite and the indexes are equally likely, we can use Rissanen's universal prior to arrive at $L(i) = \log_2 D$. The parameter description length is then:

$$L(\vec{R}) = 2k \log_2 D. \tag{3.20}$$

Equation 3.20 gives an approximation of the *optimal* number of bits necessary to transmit the parameters of a treemap, ignoring any factors that are constant across all treemaps. For the sake of simplicity, we consider a treemap with no colors or labels.

Since the pixel grid imposes a limited precision on the representation of areas, we approximate the encoded area of C in the treemap by rounding $A(C)$:

$$A'(C) = \lfloor A(C) + 1/2 \rfloor. \tag{3.21}$$

Note that $A'(C)$ does not account for precision lost by the fact that $A(C)$ has to be decomposable into exactly two factors. $\hat{P}(C)$, the probability of a class, is estimated simply as the ratio between the encoded class area and the total display area:

$$\hat{P}(C) = \frac{A'(C)}{D}. \tag{3.22}$$

As in (3.12), we assume that $\hat{P}(l)$ is estimated by normalizing $\hat{P}(C)$ with respect to the leaves dominated by C:

$$\hat{P}(l) = \begin{cases} \frac{\hat{P}(C)}{|C|} & \text{if } \hat{P}(C) > 0 \\ c & \text{if } \hat{P}(C) = 0 \end{cases} \tag{3.23}$$

where c is a constant representing the estimated probability of the leaves under a class whose rounded area is zero, and can be thought of as an uninformed probability. We can set c to an arbitrarily small value so as to penalize cuts featuring subpixel areas or, more sensibly, define c as the sum of the probabilities of the "invisible" classes in a cut, divided by the total number of tree leaves under such classes. The piecewise function above can also be defined in a more conservative way; for example, setting $\hat{P}(l) = c$ if $A'(C) < \delta$, in order to penalize cuts with small areas, where δ is the smallest visible or selectable area. For example, the desired minimum pixel area on a high resolution wall-sized display may be different than that on a smartphone device.

The data description length is $L(G \mid \hat{T})$, the log of the following likelihood of G (as discussed in Section 3.2):

$$\mathcal{L}(G \mid \hat{T}) = \prod_{l \in L} \hat{P}(l)^{A(l)} \tag{3.24}$$

It is worth mentioning that the expression above is not strictly a likelihood, but a power of the likelihood, since the data counts have been multiplied by a common factor that converts them to areas. Finally, the information criterion for selection of treemaps is:

$$L(\hat{T}, G) = L(\hat{T}) + L(G \mid \hat{T}) = 2k \log_2 D - \sum_{C \in \Gamma} A(C) \log_2 \hat{P}(l) \tag{3.25}$$

Sunburst

The structure of a sunburst can be thought of as a series of overlapping disks, one for each tree level. A tree cut can be represented as a vector of arcs $\vec{Q}$. The central angle of the arc of a class equals the sum of its

children's angles. Arc radius is proportional to the depth of a class in the tree: $r_j = (j+1)\Delta r$, with r_j being the radius of all classes of depth j, and $\Delta r = d/2h$, where d is the sunburst diameter and h is the number of tree levels. Δr is the "thickness" of each tree level in the sunburst diagram.

It is reasonable to assume that users decode a sunburst by estimating the ratio of the arc length of a class and the circumference of the disk that corresponds to the tree level where the class belongs. Assuming the sunburst is sized to optimally fit the screen, as more levels are displayed, Δr is reduced, and estimating the value of a class becomes more difficult. This implies that selecting the best tree cut depends on how many levels are displayed, and vice-versa. For example, a class with a relatively low frequency and depth 2, might be readable when displaying only three levels of a tree, but can be rendered invisible when eight more levels are displayed, as the level disks will shrink.

In order to avoid a *chicken or the egg* dilemma, where the tree cut depends on Δr and Δr depends on the tree cut, we need to define the true value independently of Δr. We can then calculate the description length of tree cuts that yield varying Δr with respect to this true value.

I define the following mapping of S, where function A is the area of the arc sector of radius $d/2$, which is independent of Δr. This is the true value to be estimated.

$$(A \circ f)(l) = A(f(l)) = \frac{f(l)}{|S|}\pi(d/2)^2. \tag{3.26}$$

A sunburst needs only two integers to inform each area, corresponding to the pixel indexes of the endpoints of an arc. Therefore, the parameter description length is:

$$L(\vec{Q}) = 2k \log_2 d. \tag{3.27}$$

The arc length s of a class C at depth j is:

$$s(C) = \frac{f(C)}{|S|}2\pi r_j. \tag{3.28}$$

Assume then, that the true value A is estimated based on the sector angle, which is, in turn, estimated based on the arc length of the sector:

$$(A' \circ f)(l) = A'(f(l)) = \frac{\hat{\theta}(d/2)^2}{2} = \frac{\lfloor s(C) + 1/2 \rfloor (d/2)^2}{2r_j} \tag{3.29}$$

where $\hat{\theta}$ is the estimated angle and j is the depth of class C. Note how the rounding of s implies that the decoded area of arcs with length smaller

than .5 is 0, due to the pixel resolution constraint. The estimated probability of a class is the ratio between the estimated area of the class and the full area of the sunburst (before cuts):

$$\hat{P}(C) = \frac{A'(C)}{\pi(d/2)^2}. \tag{3.30}$$

The data description length expression is the same used in the treemap case ($L(G \mid \hat{T})$ in Equation 3.25).

To select the MDL tree cut, we need to run two rounds of minimization. In the first, we select the best tree cut under each value of h; for instance, the best tree cut considering all levels up to level h, then $h-1$, and so on. In the second round, we select the best of the tree cuts from the previous step. The tree cut models are comparable, as they attempt to encode the same true value.

Proof-of-concept

To illustrate the use of the proposed display-tailored MDL procedure, I developed two prototype visualizations (treemap and sunburst) of the Directory Mozilla (DMOZ) dataset. As of November, 24, 2014, DMOZ consisted of 3,847,266 web pages, categorized under a total of 782,031 topics. I selected the subtree under the prefix "Top/World", which contains 2,083,282 pages written in English under 498,487 topics. I wrote browser-based clients that request tree cuts from a Node.js server. The parameters required by the server are display size and root node ID. The layouts are calculated in the server using D3 and rendered in HTML. Although the server has no knowledge of the algorithm used by the client to calculate the treemap, it relies on the fair assumption that the areas are calculated approximately as in Section 3.4.

The resulting visualizations, parameterized for a variety of screen resolutions, are presented in Figure 3.4. Note that as the display resolution increases, deeper tree cuts are selected. This is a consequence of fewer classes in such cuts being represented with tiny areas; hence, the likelihood of these cuts increases, while their description length decreases.

The treemaps drawn by the client allocate significant space for labels, in a way commonly known as "padding". That space is subtracted from the space available to represent each node's ancestors, and is also meant to help users understand the tree structure better. The MDL calculations do not account for this "wasted" space (in the estimation sense) and the clutter introduced by the labels; therefore, there is more complexity in the resulting views than what is encoded in $L(\hat{T})$. I consider, however, the results satisfactory.

3.5 VALIDATION

The proposed technique is based on the premise that a high-quality display of hierarchical data has a good balance between clutter and information; hence, the main question to be answered is whether the proposed approach is scalable, in the sense that it can consistently produce high-quality views under varying display resolutions and dataset sizes. It should be noted that it is not my intention to provide a comprehensive evaluation of abstraction approaches; instead, I am interested in comparing the proposed method with reasonable baselines to put its quality in perspective.

To address this, I adopted two validation approaches, following Munzner's (2014) nested model of validation. At the visual encoding level, I test performance in a comparative controlled study, and I report on a quantitative image analysis that measures clutter. At the algorithm level, I report the scalability of the approach.

User Study

Clutter is shown to correlate with response times in visual search tasks (Haroz and Whitney, 2012; Rosenholtz et al., 2010; Wolfe, 1998a); therefore, a sensible way to assess the level of clutter in a visualization is by measuring the time participants take to locate targets. In hierarchical displays, an important caveat of abstraction is hiding potentially interesting nodes; that is, if a node of interest is located deep in the hierarchy, more abstract views will require more drill downs to locate it. I designed a user study where participants were asked to find targets in treemaps abstracted with different methods, including MDL. Among other factors, I varied display resolution, target value, and target depth, and examined how each abstraction approach performed in interactive tasks.

Tasks

Participants were instructed on how to use the drill down (re-rooting) function and were given the *path* to the target (i.e., a list of the target's ancestors); for instance: *Top/Arts/Music*. A CSS hack was implemented to make labels not searchable with a browser's find tool. The following factors were varied in the tasks: abstraction technique, display size, dataset size, target depth, and target value. MDL was compared with three levels of depth threshold: t_3 and t_4, which correspond to the conservative approaches of capping nodes with depth greater than 3 and 4, and t_∞, which is equivalent to no aggregation. Display resolution has three levels: 375x667px, 1024x768px, and 1920x1080px, which match common resolu-

tions of smart phones, laptops, and desktop monitors, respectively. For dataset size, three subtrees of DMOZ were tested: *top*, *arts* and *soccer*, containing approximately 500,000, 55,000, and 3,000 categories each. Target depth (distance from root) varied among 3, 4, 5, and 7; and target value varied between *average* and *outlier*. The value of average targets was the average of the values of all categories in the target's level, while the value of outliers was ten times the average. Given these constraints, the target location in the tree was chosen randomly. Depending on the combination of factors, the target might be visible in the "overview" screen or drill down might be necessary to find it; for example, a target with depth 4 in a treemap where nodes with depth higher than 3 are hidden (i.e., t_3) can only be seen upon a drill down. The crossing of all factors resulted in 288 interactive tasks.

During pilot testing, I realized that some tasks might take a long time (over two minutes), and a long session is incompatible with participants' expectations of fairness in crowdsourcing tasks. Thus, each session consisted of one training task followed by 8 tasks. In total, each participant completed 9 tasks, which were assigned randomly within display resolution, in order to avoid participants having to interact with visualizations larger than their screen. Completion times and number of drill-down interactions were recorded. In order to minimize the effect of latency, in the interactive tasks the timer was paused whenever the user drilled down, and resumed once the new view was completely rendered.

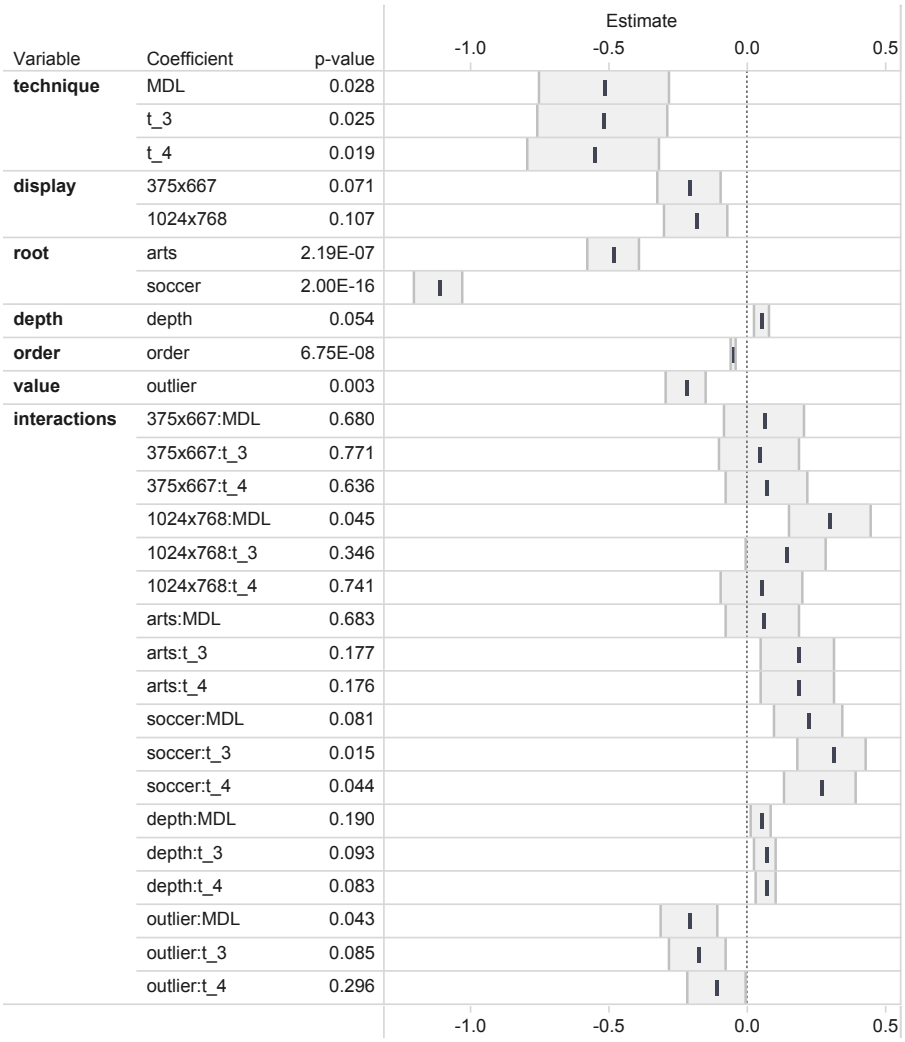
Participants

Participants were recruited with the CrowdFlower crowdsourcing platform and compensated with $2. They were presented with the instructions both on the CrowdFlower page listing my study and on the study page hosted in our servers. Participants were allowed to skip each task after three minutes and withdraw the study at any time.

Results

I analyzed 980 completed trials ($\sim$ 3.4 per task avg.) after removing 96 outliers. The median session length was 11 minutes. I used a log-linked Gamma generalized linear model, including as covariates display resolution, dataset size, target depth and target value both as main effects and in two-way interactions with technique. A new variable was created representing the order tasks are completed within the session. User was included as a random intercept. Baseline levels are t_∞, 1920x1080px, *top*, average, depth and order 0.

The model intercept is 4.37 (79 seconds). Model estimates correspond to increase/decrease in the intercept estimate, which is in log scale. For



Variable	Coefficient	p-value
technique	MDL	0.028
	t_3	0.025
	t_4	0.019
display	375x667	0.071
	1024x768	0.107
root	arts	2.19E-07
	soccer	2.00E-16
depth	depth	0.054
order	order	6.75E-08
value	outlier	0.003
interactions	375x667:MDL	0.680
	375x667:t_3	0.771
	375x667:t_4	0.636
	1024x768:MDL	0.045
	1024x768:t_3	0.346
	1024x768:t_4	0.741
	arts:MDL	0.683
	arts:t_3	0.177
	arts:t_4	0.176
	soccer:MDL	0.081
	soccer:t_3	0.015
	soccer:t_4	0.044
	depth:MDL	0.190
	depth:t_3	0.093
	depth:t_4	0.083
	outlier:MDL	0.043
	outlier:t_3	0.085
	outlier:t_4	0.296

Figure 3.5: Results from a generalized mixed linear model (Gamma, log-linked) fitted to the user study data. Response variable is completion time. Estimates are in log scale.

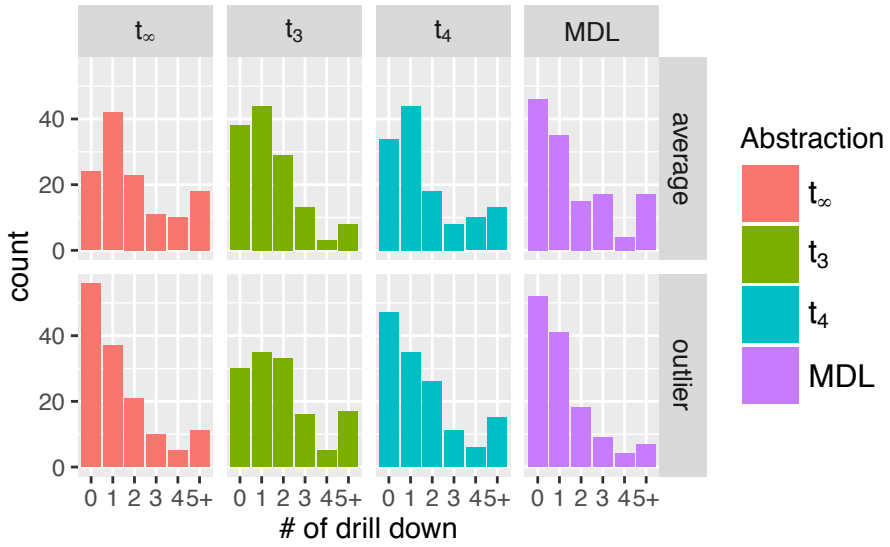


Figure 3.6: Number of drill-down interactions needed to complete a single trial of the study, grouped by abstraction technique and target value.

instance, for an intercept of 4.37, a variation of -0.1 represents a reduction of 8 seconds in mean time. The null model states that the effect is 0, indicating that the covariate has no influence on the response time. *p*-values are calculated with Wald Z-tests.

The results show that, relative to t_∞, all other techniques are responsible for a significant decrease in response times on average (Figure 3.5). Order has a small, but significant negative effect on times and so does changing the value of the target to outlier, to a larger extent. The outlier effect is significantly and slightly larger for the MDL approach, although the differences in estimates are not dramatic. Interestingly, depth does not seem to significantly affect the response variable. This may be due to not accounting for the time for new views to load when drilling down. A decrease in dataset size improves completion times only for t_∞, and in the smallest dataset condition (*soccer*), both t_3 and t_4 perform worse than t_∞. This is probably due to participants having to drill-down with t_3 and t_4, while the target is already visible with MDL and t_∞. This explains why the effect sizes are larger for *soccer* than for *arts*. Smaller display sizes are associated with a small decrease in response times, except for MDL in the 1024x768 displays, where we observe a significant increase in response times.

Figure 3.6 gives the distribution of the number of drill-down interactions needed to complete one trial, grouped by abstraction technique and target value. In the average target value condition, the distribution of values for MDL is skewed to the right compared to all other approaches;

that is, it required fewer drill downs. In the outlier condition, MDL was better than t_3 and t_4, and similar to t_∞.

Discussion

The results confirm that lack of abstraction in views of large hierarchies is detrimental to user performance, at least in visual search tasks. In that respect, even highly abstract approaches, such as t_3 and t_4, are better than unabstracted views. However, as we are not accounting for the latency between drill downs, it is possible that in high latency environments the benefits of abstraction are cancelled by the effect of latency when locating targets requires drill down. In such a case, Figure 3.6 suggests that MDL would require fewer drill-downs than t_3 and t_4. The fact that a reduction in dataset size was detrimental to user performance in all abstraction conditions suggests that abstraction for small datasets may be overkill; nevertheless, compared to t_3 and t_4, MDL was the least affected by a dataset reduction.

I expected outlier targets to be easier to spot, as their size is ten times larger than the average. The fact that MDL benefits the most of the outlier condition is likely a result of the MDL tendency to expose nodes with large model residuals.

The interaction between MDL and display size suggests a non-linear relation: response times increased with a reduction to 1024x768, then decreased with the 375x667 display. This suggests that the benefit of MDL over t_∞ is larger in the extremes of the tested display size range. In addition, it suggests that too much information may have been added to the display at 1024x768. It is possible that an adjustment in the weighting parameter that controls the importance of fitness over clutter would be beneficial.

Overall, the average response time of MDL was very similar to that of depth threshold approaches, even though the average clutter in MDL views tends to be higher. In the next section, we investigate the behavior of MDL views using an analytical measure of clutter.

Measuring Clutter

To complement the analysis of the previous section, I compared the same abstraction techniques with the feature congestion measure of clutter (Rosenholtz et al., 2010), which is based on the notion that clutter in a display is associated with degrading performance in visual search. It essentially measures the difficulty of adding a new, salient item to a display. The measure computes the local variability of color and contrast lumi-

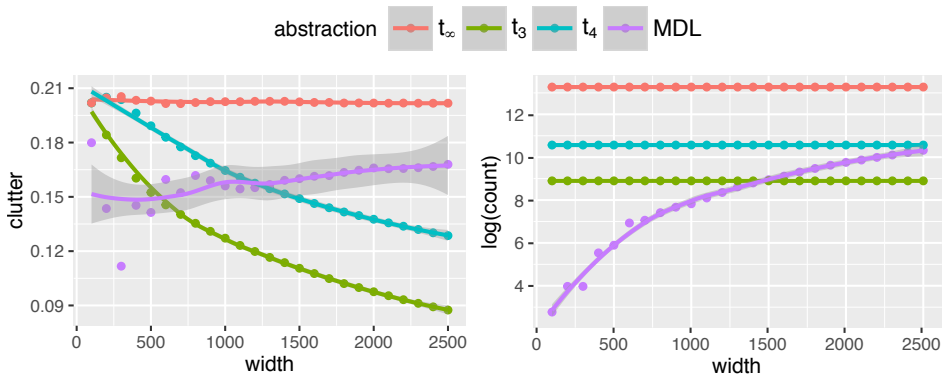


Figure 3.7: Left: Level of clutter computed with the Feature Congestion measure as a function of display size. Right: Node count as a function of display size. In both charts, the dataset is DMOZ, and aspect ratio is 1:1.

nance at multiple scales, then combines the values over space and scale to generate a scalar.

Feature Congestion Measure

To calculate the feature congestion measure, the image is represented in the perceptually uniform color space CIELab. Three spatial scales are created for the image using a Gaussian pyramid, which is a multilevel structure where each scale δ is created by smoothing and subsampling the representation at scale $\delta - 1$. In a Gaussian pyramid, a many-to-one correspondence exists between pixels in adjacent pyramid levels.

Next, color and luminance contrast features are found for each scale. For luminance contrast, a difference is computed between the results of two Gaussian filters. This common procedure, which measures the intensity of a region relative to its surroundings, captures the center-surround operation of visual receptive fields (see Itti et al. (1998) for a similar application). The color feature corresponds to local mean color, computed by pooling with a Gaussian filter.

For each of these features, local covariance is computed. From the covariance matrices, the volume of the covariance ellipsoids is calculated, which is the final *local* measure of feature clutter. Hence, we have separate 2D maps of clutter for color and luminance contrast for each scale. Intuitively, large covariance ellipsoids indicate a large utilization of the feature space, and consequent feature crowding.

In order to obtain a *global* measure of clutter, these clutter maps are pooled spatially and across scales, resulting in scalar measures of clutter for each feature. At last, the scores are linearly combined, producing a single image clutter score.

Procedure

I generated treemap views of the DMOZ's subtree "Top/World", the same used throughout this chapter, in resolutions ranging from 100 x 100px to 2400 x 2400px, with the four abstraction approaches tested in the user study: t_∞, t_3, t_4, and MDL. Then I calculated the feature congestion measure using only contrast luminance, as the treemaps do not vary color. In addition, the views were generated without labels, in order to focus on clutter caused by tree structure. Padding was kept, as it is usually necessary for understanding structure in treemaps featuring deep levels.

The results of the experiment are shown in Figure 3.7, left. The clutter of t_∞ views remains constant and high across the whole range of resolutions. t_3 and t_4 decrease exponentially as the resolution grows. Just like t_∞, the clutter in MDL views remains constant, but is lower than t_∞. Note that for small displays, MDL ends up "falling back" to t_3 and t_4. As space becomes available, the distance between MDL and the depth thresholded views becomes higher, with MDL filling the space with more data.

While clutter is often considered to be unwanted, it is positively correlated with information density, and my approach attempts to find a balance. So, while the clutter of t_3 and t_4 drops dramatically at large screen sizes, so does the information density. Clutter can be compared with the number of nodes visible in the visualizations as seen in Figure 3.7, right. MDL, t_3, and t_4 consistently reveal far fewer nodes than t_∞. The number of nodes revealed by MDL increases with screen size, while maintaining a roughly constant level of clutter. I argue that while MDL reveals a smaller number of nodes at screen width 1024px, compared to t_3, and the clutter is higher, this is due to the better (more uniform) distribution of nodes across the space, as seen in Figure 3.8.

Woodruff et al. (1998) argue in favor of constant information density (e.g., constant number of objects per area) across x, y and z dimensions of a multiscale visualization. They achieve that automatically by modifying the visual representation of data points and by adjusting the level of abstraction unevenly. Figure 3.8 (middle) demonstrates that MDL can also approximate constant information density across x and y dimensions. In addition, the results of the feature congestion experiment suggest that MDL approximates constant information density across display resolutions. Across the z dimension (drill-down) I saw variations in the information density caused by the expansion of nodes that have many children (*fan out effect* (Archambault et al., 2008)). Unlike VIDA (Woodruff et al., 1998) and GrouseFlocks (Archambault et al., 2008), which create new aggregates to minimize fan out, my method preserves the original hierarchy structure. As a result, if there are strong variations in how wide the first levels of subtrees are, the information density can vary.

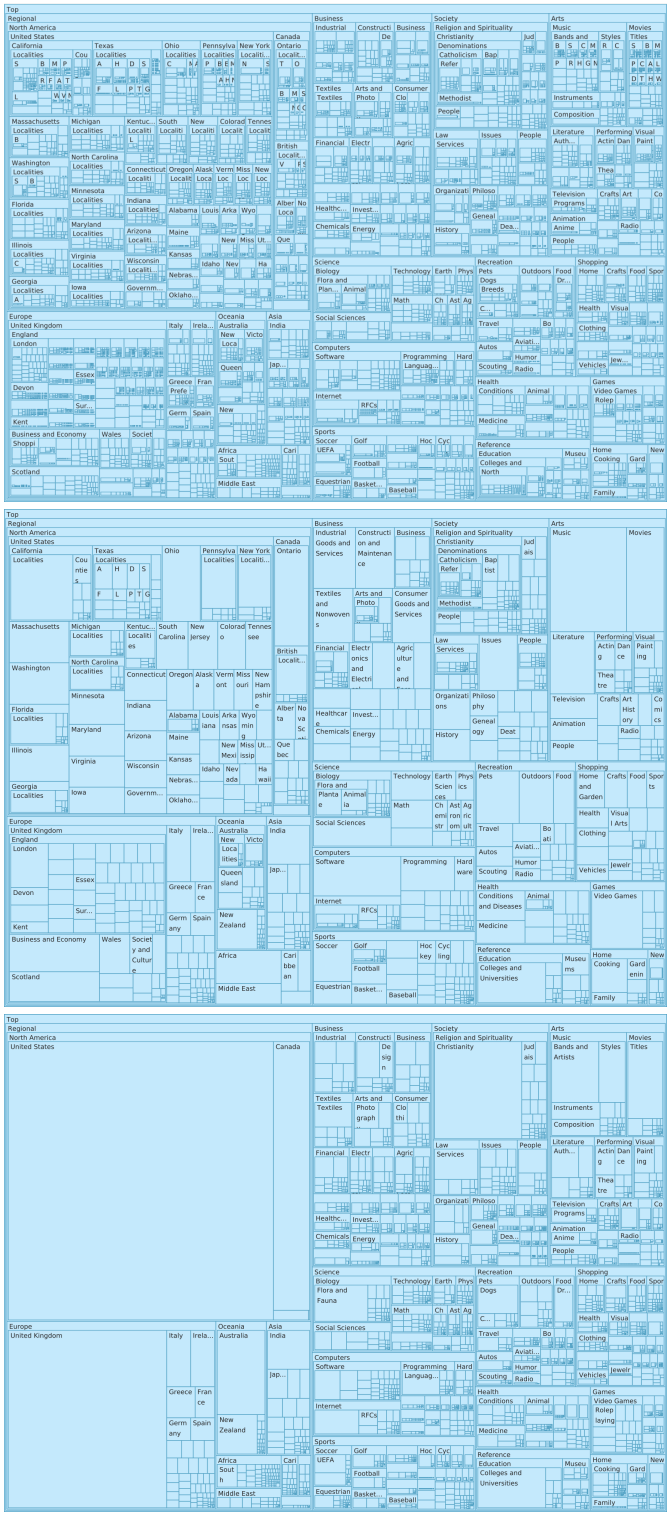


Figure 3.8: Treemap views of the DMOZ dataset for a 1024x768px display. On the top, the full view of the dataset. In the middle, the level of detail is based on the best MDL tree cut (uneven). On the bottom, an even cut is performed below level 4 of the tree. The MDL tree cuts yield a better balance of information density and clutter.

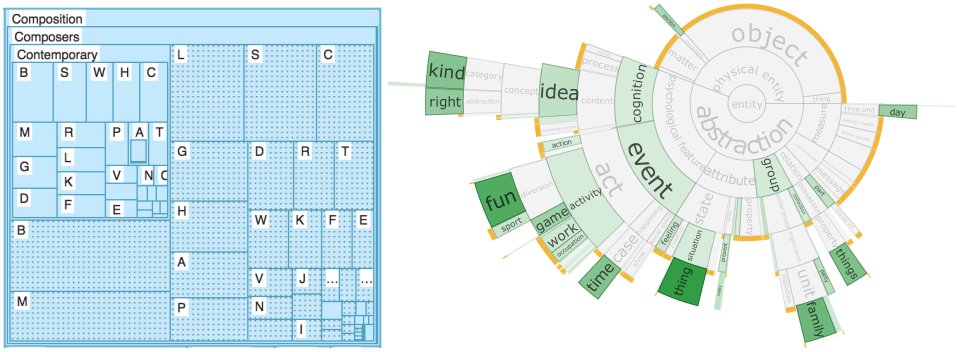


Figure 3.9: Subtle visual cues for collapsed nodes using texture (treemap) and border thickness (sunburst).

Scalability Analysis

The MDL algorithm is a customization and application of the approach of Li and Abe, with the additional optimization step of including a factor of display size. Li and Abe found that determining the MDL tree cut terminates in time $\mathcal{O}(NxS)$, where N denotes the number of leaf nodes in the input tree T and S denotes the input sample size. The algorithm I propose here increases this procedure by the transformation from the data domain to the pixel domain, and the estimation of the probabilities of leaves, both of which are $\mathcal{O}(NxS)$. As S is generally much larger than N, my algorithm scales roughly linearly with the size of the dataset.

Any overhead encountered by generating a display-optimized tree cut could be shifted to a server-side pre-calculation, for example, to pre-cache the tree cut for a variety of standard screen sizes, thereby eliminating any delay incurred by the tree cut operation. The resulting trees generally balance better information density with the number of nodes, and will render faster and consume fewer client resources than an equivalent full tree, and show a more uniform information density than a fixed-level tree cut.

3.6 DISCUSSION

Generality: With the formulae for treemap and sunburst visualizations, I exemplified how model selection criteria can be written for visualizations under the MDL framework. It is possible that good results can be achieved with MDL with many other kinds of hierarchical visualizations where (a) some visual aspect of the nodes is weighted by a score, and (b) the scores are cumulative. This might include visualizations that are not traditionally hierarchical but were augmented with multiscale function-

ality, such as aggregated scatterplots, parallel coordinates and node-link diagrams (Elmqvist and Fekete, 2010). Defining criteria for new classes of visualization involves the specification of three main expressions:

- The transformation from the data domain to the visualization domain (pixel units) ($A(C)$).
- Number of bits necessary to encode the visualization ($L(V)$).
- How probabilities of tree leaves are estimated from the visual representation of classes ($P(l)$).

Uniform distribution assumption: Behind the estimation of the probability of leaves given a class is the assumption of uniform probability. If this assumption is not reasonable in a certain application domain, $P(l)$ can be easily changed to reflect a different probability distribution. A case where this might be useful is when depicting geographic information, where the user might have a prior assumption about the distribution; for example, given a certain value for the State of New York (e.g., gross product), one might expect that value to be concentrated in New York City. In many other cases, lacking prior knowledge, I expect the uniform distribution to be fairly reasonable.

An important limitation is that if the data is uniformly distributed, the tree cut generated will be the most abstract possible (i.e. the root). This occurs, for example, if the value of every leaf is 1. Likewise, if a different distribution is used and the data conforms exactly, the tree cut will be overgeneralized. This occurs because the goal of MDL is the shortest message, and when the data conforms to the model expectation nothing stands on the way to selecting the most concise model.

Interpretation: It is especially important in models such as mine, where abstraction is calculated algorithmically, that the presence of data abstraction is made apparent in ways that are not distracting to the main task of working with a visualization. While my technique and evaluation focus on the level of abstraction, I have begun an investigation into the representation problem. Figure 3.9 suggests preliminary visual designs which subtly distinguish aggregates from regular leaves. On treemaps, aggregates are textured; on sunbursts, collapsed nodes are decorated with a colored, thicker border.

3.7 SUMMARY

I presented a technique for using the MDL Principle, extended with considerations of display space, to create optimized views of hierarchical

datasets which fit the "analyze first, show the important" first step of the visual analytics pipeline. In addition to providing overviews customized to dataset and display size characteristics, the display-optimized tree cuts can be interactively expanded by changing the weighting parameters.

The number of nodes displayed in a display-optimized MDL tree cut is similar to those in an even tree cut at a set depth, but fewer than showing a full tree. This increases the rendering efficiency, resulting in a performance gain in web-based visualization applications, where processing resources, memory, and display space may be constrained (e.g. on mobile devices). In addition, on small screens and any touch device where rendered elements are small, interaction accuracy can be difficult due to the "fat fingers" problem. My technique applies abstraction in cluttered areas of a visualization, which will likely improve target selection accuracy.

I have demonstrated my technique applied to two datasets across two different hierarchical visualization types, treemap and sunburst diagrams, and outlined the steps required to generalize the approach to other visualization types. Display-optimized MDL tree cuts may prove especially useful due to their general nature — they are not customized to dataset characteristics. However, it is also possible to tailor them to the dataset, for example, by basing the tree cut on a selected data attribute, as long as that attribute is quantitative on the leaves and cumulative in the hierarchy.

Future work includes applying the display-optimized MDL tree cut to new visualization types. In addition, I see promise in the challenge of developing new methods for representing abstraction. While I demonstrate the possibilities of interactive drill down to deeper levels of the tree cut using a fixed step size, there is promise in investigating ways to automatically tailor the drill down step based on dataset characteristics, display space, and to harmonize tree cut drill down with more traditional techniques to click and open branches manually.

4 SALIENCY DEFICIT

New visualization designs are created in academia and in industry at a faster pace than rigorous evaluation can follow. One way to inform a broad audience and validate a large number of designs at once is by running controlled experiments that examine fundamental questions. Empirical visualization research aims at laying out and continuously testing this foundation.

In this chapter, I investigate questions related to the independence of visual dimensions in animated scatterplots. We often seek to encode data in as many visual variables as possible, and this strategy has been extended to scatterplots with the use of color, size, and motion. Here we question the accuracy of the basic task of motion outlier detection in the complex scenes formed by animated multivariate scatterplots. Does the saliency of non-motion features impact the detection of motion outliers? Can we put motion outliers in a state where they are hard to detect by simply changing their color, size, or position? If so, in visualizations where observing change is a relevant task the variations in data point saliency will hinder or amplify the local perception of change, turning the encoding unreliable.

The perception literature has abundant studies on the performance of search tasks in static and moving scenes (Dick et al., 1987; Duncan and Humphreys, 1989; McLeod et al., 1988; Von Mühlenen and Müller, 2000; Wolfe, 1998b). However, psychology studies are difficult to comprehend by non-experts and their low level make it difficult to extract implications to visualization design. Nonetheless, these controlled experiments produced general results that support useful rules of thumb; for instance, targets among uniform distractors are much easier to detect than when the distractors have high variance (Dick et al., 1987). This rule captures well the results of "pre-attention" experiments with single and conjunction static features (e.g., color), and with motion components (speed and direction). Detection of speed and direction outliers in displays where no other features compete is considered efficient, and the effects of speed on direction and vice-versa are well studied (Rosenholtz, 1999). However, detecting speed and direction targets in scenes where many other channels are used is not well studied.

In the second edition of his book, Ware warned that studies on perceptual independence among three or more visual channels were rare (Ware,

2004). Almost 15 year later, our understanding of these interactions and their implications to visualization is insufficient, and fewer are the studies that involve motion in visualization. Progress recently has been made in revising rankings of encoding effectiveness (Kim and Heer, 2018; Moritz et al., 2018). While these have great practical application, they do not seek to explain the fundamental phenomena driving performance results.

Among the powerful concepts that may help us unveil the roots of problems in the visual mapping of data is visual saliency. In the next section I contribute an experiment aimed at measuring the gap in motion outlier detection accuracy between salient and non-salient outliers. I simulate animated scatterplots that contain either a speed outlier or a direction outlier. Then I vary the number of static features that, in addition to motion, are salient in these outliers.

The results show that motion outliers that have additional salient features are much more likely to be correctly identified than non-salient outliers. This suggests that motion is not immune from interference of other dimensions and that motion outlier detection is unreliable in multivariate animated scatterplots. I define the notion of *saliency deficit*: a state where the saliency profile in a visualization scene impairs the effectiveness of performing a visualization task; and suggest that saliency deficit models can help the automatic identification of saliency-boosting opportunities in visualizations.

4.1 RELATED WORK

I am interested in the role saliency plays in motion outlier identification. While this question has wide-ranging applications, I constrain my investigation to animated scatterplots. In this section I review the related work in perception for information visualization, the use of animated scatterplots, and the recent trend of developing empirical perception models for visualization.

Perception

Visual attention research investigates the limits of attention of the human visual system and has produced a number of theories that explain the mechanisms of visual information processing (see Healey and Enns (2011) for a review). Feature integration theory proposes that scenes are initially processed as many separable basic dimensions (e.g., color, motion, orientation), which are later integrated to form more complex objects (Treisman and Gelade, 1980). Without focused attention, features re-

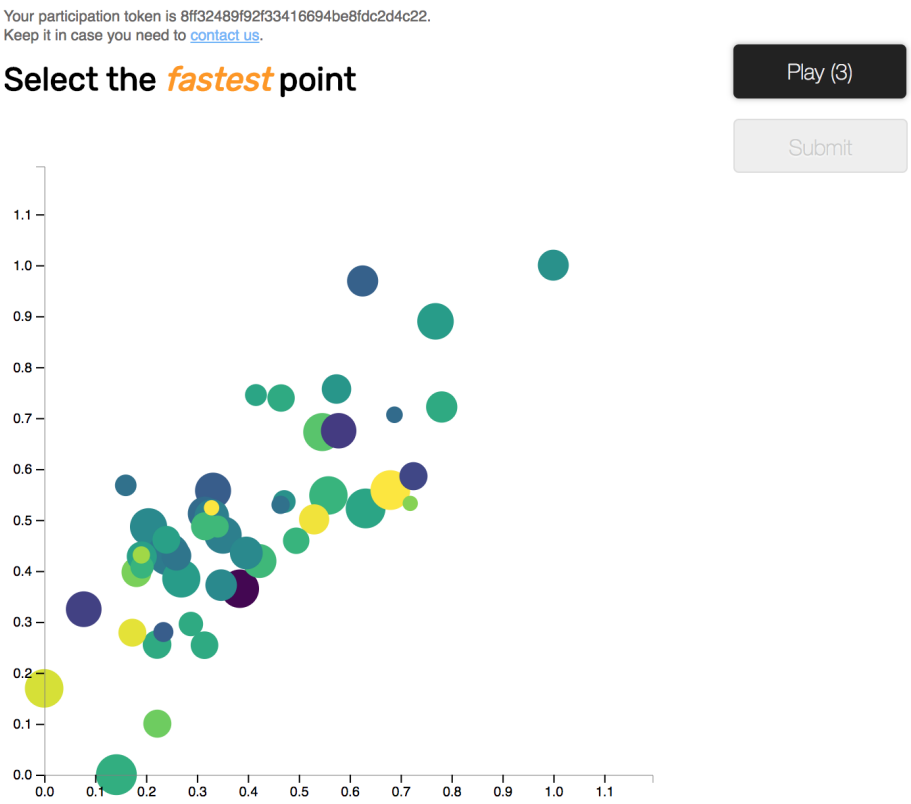


Figure 4.1: Snapshot of the interface for the speed task. The direction task asked "Select the point that moves in the most deviant direction."

main separated. As a consequence of this mechanism, searches for basic features occur in parallel and are fast, while searches for conjunction features, which involve more than one dimension (e.g., a red circle in a scene with red squares and blue circles), occur serially and thus slow down as the number of objects present in the scene increase. Visual search experiments usually ask participants to determine whether a target is present in a scene with distractors, and the number of distractors is manipulated. Reaction times (RT) and accuracy are recorded, and results are summarized as the slope of the linear relationship between the response and the number of distractors. Parallel searches have slope close to 0. Frequently, the term "popout" is used to describe the easy identification of targets in these searches.

While many experiments corroborate feature integration theory, other experiments found that some conjunction searches are too efficient to be serial searches. For instance, motion-shape targets can be detected in parallel, suggesting the existence of a *motion filtering process*, which effectively subsets the scene, reducing the search task to a simple feature

search on moving items (McLeod et al., 1988; Von Mühlenen and Müller, 2000). Aiming at explaining these problematic cases, the theory of guided search posits that the goals of the viewer play a large role in visual search, with activation maps ("heatmap" representations of the visual space storing the likelihood of locations containing a target) being constructed with bottom-up and top-down information. Top-down processes are cognitive, driven by users tasks and goals, while bottom-up processes are driven by sensory information. Guided search theory suggests that the difference in performance between single feature and conjunction tasks is due to the amount of guidance that bottom-up processes can provide (Wolfe, 1998b). Thus top-down guidance is the reason "fast" conjunction searches exist.

The impact of color on motion discrimination is well studied. Both hue and luminance have been shown to independently enable *apparent motion* of simple objects when they are displayed in different positions in successive frames, prompting debate as to whether or not color and motion are processed by separate pathways (Papathomas et al., 1991). Croner and Albright (1997) found that hue saliency and luminance saliency aid the discrimination of motion direction; that is, participants detect more accurately targets moving in the same direction among distractors moving in random directions when the targets have distinct hue or luminance, which may suggest that color segmentation of the scene occurs prior to motion discrimination, a process opposite to the motion filtering mentioned above.

The statistical saliency model (SSM) (Rosenholtz, 1999) seeks to explain motion popout phenomena with a simple statistical measure that quantifies the saliency of targets with respect to the distractors in the scene. The SSM explains the following asymmetries in motion popout phenomena: a) searching for a moving target among still distractors is easier than searching for a still target among moving distractors; b) searching for a fast target among slow targets is easier than the opposite; c) adding variability in speed when searching for a unique motion direction has little effect, while adding variability in direction when searching for a unique speed makes the search task more difficult. The SSM is compelling because calculation of the saliency of objects is trivial and efficient, and because it has been shown to explain search results in experiments where dimensions other than motion are examined. I review this model in more detail in Section 4.2.

I enumerate the following challenges in transferring the existing perception knowledge to the problem addressed in this work:

1 In the perception experiments cited above, targets are chosen arbitrarily. In this experiment, targets are outliers in the statistical sense. I ask

whether outlierness as a statistical property is preserved through the visual mapping.

2 Motion outlier detection in scatterplots is not a conjunction task. While the conjunction of motion and other dimensions is well studied, the problem here is defined as a basic feature search in the presence of many irrelevant dimensions.

3 The dimensions in my stimuli encode continuous data attributes, while in perception studies they are often discretized to some degree (e.g., moving / still, fast / slow, bright / dim) (Croner and Albright, 1997; McLeod et al., 1988; Papathomas et al., 1991; Von Mühlenen and Müller, 2000).

Animated Scatterplots

Scatterplots are one of the most effective visualizations because they employ position along a common scale, which was found to be the representation with which people can most accurately perform visual judgments (Heer and Bostock, 2010). Less important dimensions are commonly mapped to color, size, and shape. Gleicher et al. (2013) demonstrated that people can accurately compare means in multiclass scatterplots despite the addition of one discrete irrelevant cue (shape). This work shows that people can comfortably extract a summary statistic confined to a single dimension in the presence of an irrelevant dimension. Here, I investigate whether another summary statistic (outlierness) can be extracted from *motion* in correlated scatterplots with more than one irrelevant dimension (color, size). A key difference is that my scatterplots do not feature discrete dimensions that would enable the visual segmentation of the scene.

Szafir et al. (2016) argue that ensemble coding allows us to visually extract statistical information from scatterplots, such as outliers and statistical summaries, but acknowledge that attentional control may be problematic when multiple variables are encoded simultaneously, although the empirical basis is still lacking. Robertson et al. (2008) found that animated scatterplots were not superior to static trend visualizations in analytical tasks (error rates) focused on trajectories. Huber and Healey (2005) devised precise discriminability lower limits for motion (in displays with no competing visual channels): a target-distractor difference of a least 20 degrees is necessary for direction oddballs to be detected accurately; for speed, the difference needs to be at least 0.43 degrees of visual angle. Our outliers satisfy these conditions (Section 4).

Albeit designed to devise guidelines for notification design, Bartram et al.'s study of visual cues came to conclusions that relate to visualization design. Subjects were asked to perform a task in a window while glyphs overloaded with various encodings were scattered in the periphery (Bar-

tram et al., 2003). The authors measured how accurately subjects could detect change in the glyphs. Motion was found to be the most reliable cue, better than changes in shape and color. They concluded that motion "does not seem to interfere with existing color and form coding" and that motion detection is effective even in visual periphery and with small amplitudes.

Etemadpour et al. (2014) and Etemadpour and Forbes (2017) used motion as a solution to clutter on the assumption that motion does not suffer interference from other channels. They reported a large improvement in the accuracy of ranking cluster density when motion was used as an encoding for cluster density. The improvements were relative to scatterplots where density was not explicitly encoded (implicitly encoded as position); plus, density is necessarily correlated to position, which makes motion-position a double encoding for density. Similarly, animated scatterplot matrices that encoded density with flickering were found superior to conventional ones in density judgement tasks (Chen et al., 2018).

4.2 SALIENCY

The statistical saliency model (SSM) (Rosenholtz, 1999) is a model of visual search based on the intuition that the visual system is interested in unusual things. Rosenholtz represents a visual scene in an appropriate feature space and then computes the saliency of a target as the number of standard deviations between its feature value and the mean of distractors. Their model can be seen as a formalization of Duncan and Humphreys's (1989) rule of thumb that states that search is easier when target-distractor similarity decreases, or when distractor-distractor similarity increases.

Formally, saliency is defined as following in the SSM. Given a set of feature vectors, the saliency, S, of a target vector is defined as its Mahalanobis distance to the mean of the distractors (Rosenholtz, 1999):

$$S = \sqrt{(T - \mu_D)' \Sigma_D^{-1} (T - \mu_D)} \tag{4.1}$$

where Σ_D is the covariance matrix of the distractors, T is the target vector and μ_D is the mean of the distractors. Mahalanobis distance is a measure of the distance between a point and a distribution, and is commonly used to find outliers in multivariate data. In the one-dimensional case, the Mahalanobis distance is equivalent to a z-score, that is, the number of standard deviations a point is from the center of the distribution, while in the multivariate case, it corresponds to the number of covariance ellipsoids from the center.

In qualitative terms, Rosenholtz defines the saliency of an item or a region as the ease of search if that item or region were targets in a scene; alternatively, it can be defined as the likelihood of an item attracting eye movements, assuming zero influence of the task. These notions of saliency are compatible, and are consistent with the use in similar vision science and information visualization models (Itti et al., 1998; Matzen et al., 2018).

The use of search tasks and reaction times as proxies for attention relies on the premise that search for salient items should be faster than search for items that do not draw attention. Rosenholtz's study of visual search is directly relevant to motion outlier detection in visualization, and to ranking, indirectly, if we assume that ranking points defaults to finding the most outlying point in increasingly narrow search spaces. For our purposes, however, the existing empirical validation of the SSM is limited. First, the scenes used to test it are usually distractor arrays of constant density (as in a uniform grid) (Dick et al., 1987); second, no more than two features (speed and direction of motion) are varied. In information visualization displays, especially scatterplots, the x and y positions of points are commonly correlated, forming point clouds with varying density and levels of occlusion, and the points may be overloaded with multiple visual encodings, such as color, size, and shape (Szafir et al., 2016).

A subsequent paper demonstrates how the SSM predicts asymmetries in colour search in the presence of non-neutral backgrounds (Rosenholtz et al., 2004). The model is also the foundation for the feature congestion model of visual clutter (Rosenholtz et al., 2007), where separate pixel-level saliency maps of color and contrast luminance are linearly combined to produce clutter maps for raster images. The maps can be further aggregated to produce a scalar measure of overall display clutter. To evaluate the feature congestion model, the authors compared its predictions for a clutter-ranking task on a collection of 25 maps with the rankings elicited from 20 people. Spearman's rank-order correlation was high (0.83, $p < .001$) and approximated the average correlation between pairs of subjects (0.70).

Critically, it is not clear how low-level dimensions should be composed for the calculation of saliency in complex visualizations. In Rosenholtz's study of motion outlier detection (Rosenholtz, 1999) it was suggested that the Mahalanobis distance should be calculated on the 2D space formed by speed and direction of motion, whereas in the feature congestion model saliency is calculated as a linear combination of 1D saliencies. It is likely that the latter is the appropriate method in a scene where motion and static features are varied, in which case we need to learn the dimension coefficients.

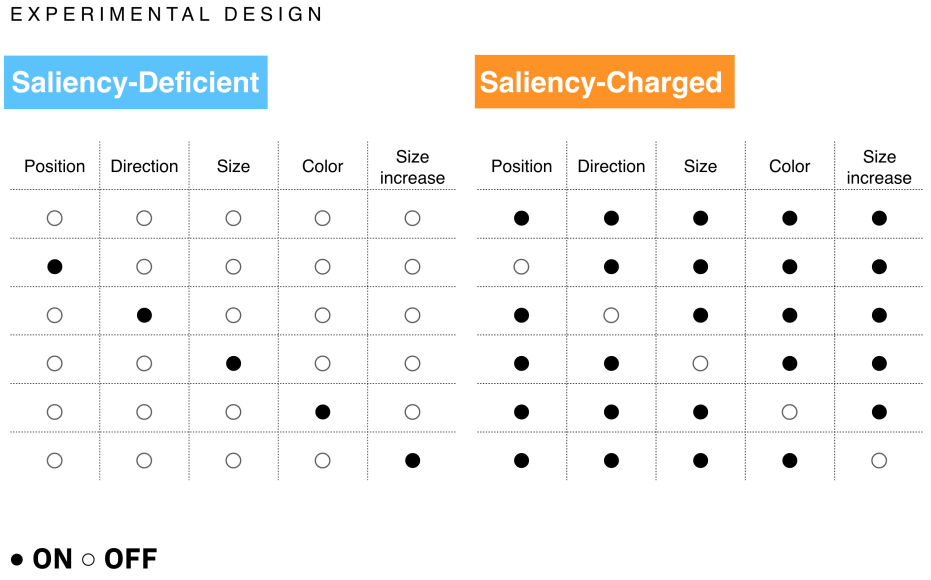


Figure 4.2: Illustration of the experimental design. The saliency-deficient group has one baseline condition where no target features are salient and conditions where only one feature is salient. The saliency-charged group has one baseline condition where all target features are salient and conditions where all but one feature are salient.

The pixel-level saliency maps employed in the feature congestion model and in many other saliency models (Judd et al., 2012) are not compelling for visualization applications because they operate after rendering, a late stage of the visualization pipeline, and because they are commonly tuned for natural images (Bylinskii et al., n.d.). Recently, saliency models for data visualization were proposed (Bylinskii et al., 2017; Matzen et al., 2018) that owe their performance mostly to accurate predictions of fixations on text elements (e.g., labels) in static visualizations.

In the next section I will explain how the stimuli were created with salient and non-salient targets following SSM's definition of saliency.

4.3 EXPERIMENTAL DESIGN

I designed an experiment to find whether saliency predicts accuracy of motion outlier detection tasks in animated multivariate scatterplots. In particular, the experiment investigates whether saliency in irrelevant dimensions influences accuracy. Irrelevant dimensions are those that are not part of the task; for instance, when participants are instructed to find

the fastest point, all dimensions (color, position, etc.) but speed are irrelevant.

The experiment is split into two *tasks*, a direction task and a speed task. The former asks participants to select the point with the most deviant direction, the latter asks them to select the fastest point. From now on I will refer to visual channels as *dimensions*, and to specific values in these dimensions as *features*. I will also call direction and speed the *relevant dimensions* in their respective tasks. Each animated scatterplot display (a *scene*) produced has 12 *conditions*, where only the target is varied: a baseline where the target has no irrelevant salient features, plus five instances where it holds a single irrelevant salient feature (position, color, size, direction/speed, or size increase); a second baseline where the target has five irrelevant features at once, plus five instances where one irrelevant feature is held out. Thus, half the stimuli follows a one-at-a-time design, and the other half follows a hold-one-out design. These condition groups are called *saliency-deficient* and *saliency-charged* (see Figure 4.2). The following notation is used to refer to individual conditions: in the saliency-deficient group, + conditions refer to the added irrelevant salient feature. For example, *+position* refers to a stimulus where the only irrelevant salient feature is position. In the saliency-charged group, - conditions refer to the removed irrelevant salient feature. For example, *-position* refers to a stimulus where only position *is not* salient. In all stimuli, the target has outlying value in the relevant dimension.

The reader may question why I do not vary dataset size, correlation, or the parameters of the sampling distribution. When distribution and dataset size are manipulated, the fundamental quantity that is being varied is the saliency of the target. For instance, a scene with more point spread results in less target saliency, and the same with a more crowded scene. As the goal is to find the effect of saliency on accuracy and saliency is already being varied through manipulation of visual features, varying the factors in question would be redundant. Therefore, I see no reason in increasing the complexity of the experiment by adding additional variables.

I generated ten different scenes per task, across 12 scene conditions, for a total of 120 stimuli per task. I collected 20 judgments of each stimulus for a total of 2400 judgments collected for each task, 200 per task-condition. I am interested in measuring the differences in error rates between the saliency-deficit and the saliency-charged baselines, and the impact of introducing or removing features.

Stimuli

The procedure for generating realistic stimuli is inspired by animated scatterplots of the Gapminder data. The Gapminder plots map an often correlated pair of variables to the x and y coordinates, use size to encode a time-varying quantitative variable (usually population), and map a categorical variable (continent) to color. In our scenes, we simulate instead a *continuous* variable mapped to color because it allows fine-grained control of the saliency.

A scene has 50 data points and is composed of two frames that are linearly interpolated to produce the animation. Motion is decomposed into distance, which determines how much the point moves in the 2D plane (Euclidean distance), and direction. I sampled the features for the initial frame and calculated the positions in the final frame based on sampled values for distance and direction. x_1 and y_1 are sampled from a multivariate normal distribution with correlation 0.7. The values for color, size increase, and distance are sampled from independent normal distributions. Direction (angle) is sampled from a beta distribution ($\alpha = 9.55, \beta = 10$) that has shape similar to a normal, but produces values that are more concentrated around the mean. This pattern was chosen to preserve the correlation of the plot; that is, the point cloud, *as a whole*, should be moving in a well-defined direction. Due to the animation duration being constant for all points, distance is effectively a measure of speed.

After all points are sampled, a target is selected according to the condition. If position is salient, then I select the point with the highest Mahalanobis distance (i.e, the most distant from the center of the point cloud); otherwise, the point *closest* to the center is selected. If color is salient, I assign to the target the maximum color in the color range; otherwise, I assign it the mean color. This pattern is followed for all the other irrelevant visual dimensions.

All targets are outliers detectable through the interquartile range method (Tukey's fences, k=1.5); thus, an analyst using boxplots to analyze the distributions of speed and direction would clearly identify the target as an outlier (positioned beyond a boxplot's whiskers). I produced outliers by assigning to targets a constant value outside the sampled distribution range. On average, direction and speed outlier values were 3.11 and 3.82 standard deviations from the mean. For comparison with Huber and Healey's discriminability thresholds, in average, the trajectory of speed outliers was 0.95 degrees of subtended visual angle longer (40% higher) than that of the next fastest point on a 113ppi laptop screen (e.g., Macbook Pro 13in.) at typing distance (20in.). The difference between direction outliers and the next most deviant points was 52 arc degrees (43% higher), in average.

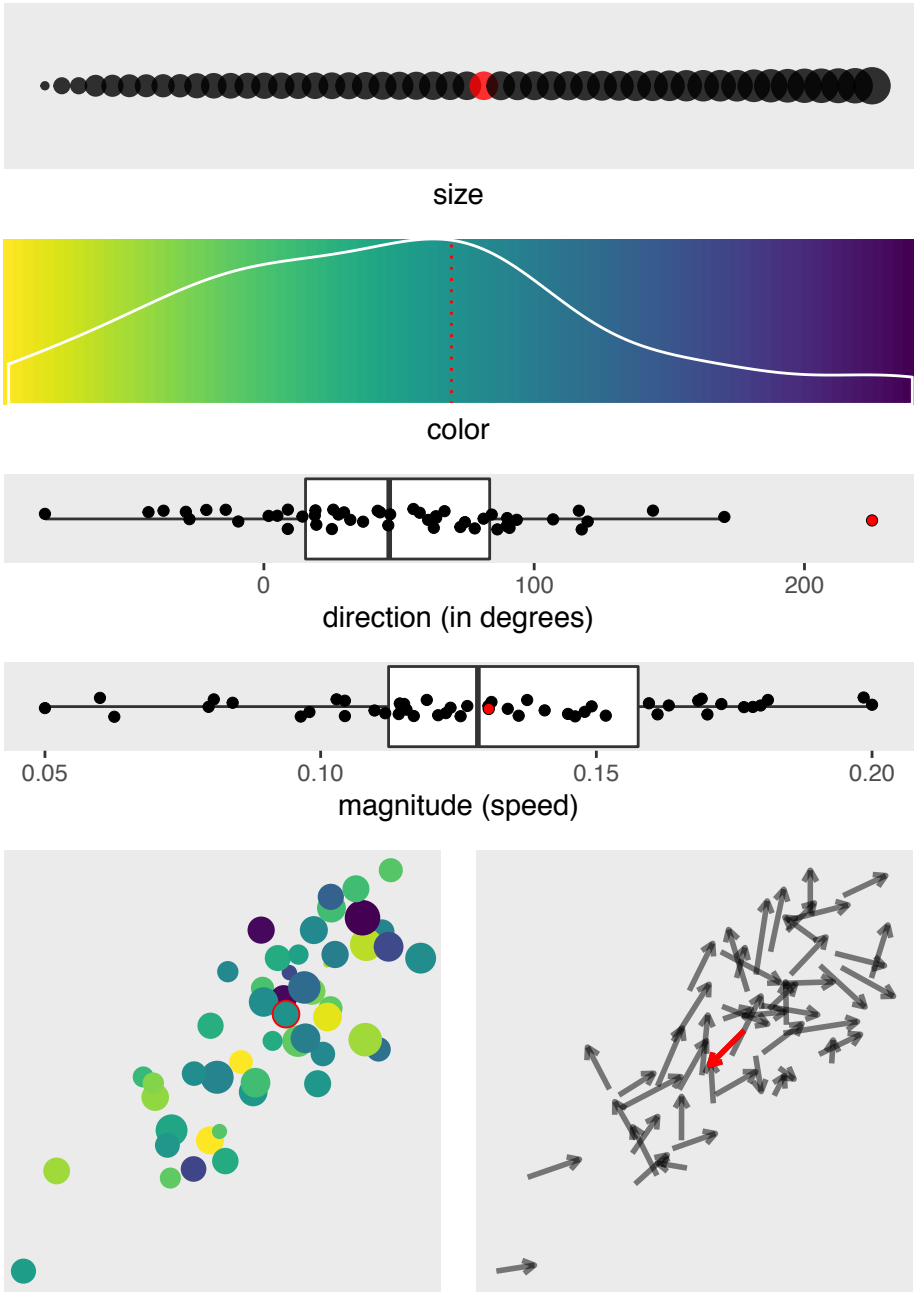


Figure 4.3: Feature distributions in a typical direction stimuli. Values are sampled from independent distributions. The target point (marked in red) receives an outlying value in the dimension of interest (either direction or speed). Depending on the condition, the target can have mean or maximum (salient) values in the irrelevant dimensions. For instance, in the condition *+color*, the target has salient color. The scatterplot in the bottom left is the first frame; the arrows in the bottom right represent the displacement between the initial and final frames, which is animated.

Table 4.1: Feature ranges. When speed is the task, the target is assigned an outlying distance value and mean or salient value for the other features. When direction is the task, the target receives an outlying direction. The color spectrum is defined by matplotlib's Viridis colormap.

Feature	Range	Unit	Mean	Salient value	Outlier value
x, y	[0, 500]	px	250	variable	
color	[●, ●]		●	●	
size (area)	[100, 600]	px	350	600	
size increase	[1, 2]	multiplier	1.5	2	
distance	[25, 100]	px	62.5	100	150
direction	[-81, 171]	degree	45	171	225

Table 4.1 lists the dimension ranges for the sampled points, as well as the mean and salient values. I use the inverted version of matplotlib's Viridis colormap (Smith and van der Walt, 2015), where higher values are darker (bright points on a white background would not "pop out"). Viridis was found to have superior performance, measured in time and accuracy of relative similarity judgments, in comparison with other popular colormaps (Liu and Heer, 2018). I chose the direction range again respecting the principle that the plot trend should not be overly disrupted. The size range was chosen so as to not cause too much occlusion. In addition, the render order on the screen (from largest to smallest) also reduced occlusion. The stimuli were inspected to make sure that the targets were not occluded. Size increase is a multiplier of the initial area. Figure 4.3 displays a scene for the direction task in the saliency-deficient baseline condition. The target moves in an outlying direction but has average values for speed, color, size and position.

Procedure

The stimuli was presented embedded in the Mechanical Turk interface (Figure 4.1). The page presented the first frame of the animation until the play button was pressed. After the end of the animation, the visualization was stationed in the second frame, allowing participants to select the target and submit the response or replay the animation up to two times before submission. The animation duration was 500 milliseconds. When play was pressed the second or third time the points faded to a blank screen then reappeared in their first frame positions before the animation took place. This sequence is illustrated in Figure 4.4. The variable number of views was introduced as a measure to mitigate errors due to interruptions, as these can be a problem in crowdsourced studies where I have no

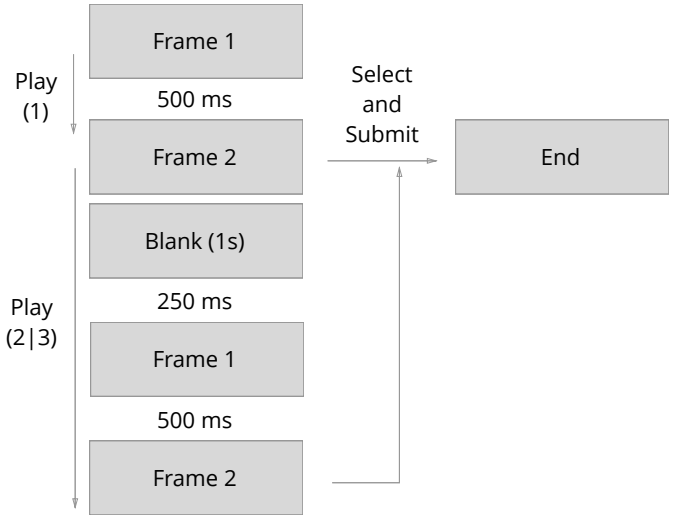


Figure 4.4: Flow diagram illustrating the sequence of screens in the study interface. Participants could replay the animations twice. Blank screens were place in-between replays.

control over the environment. The number of views was capped at three to prevent the task becoming too easy to the extent no differences can be detected between the conditions. Trials were published as two separate groups of HITs on Mechanical Turk (speed and direction). Within each group, trials appeared in random order. Participants were not limited in the number of tasks they could complete. I recorded time, accuracy and number of views.

Participants were instructed to find the fastest point ("find the fastest point") in the speed task and the most deviant point ("find the point that has the most unique trajectory compared to the rest") in the direction task. Therefore, the task is to “find the maximum”, with all targets being outliers. This mitigates the risk of participants not comprehending the outlierness concept or the study being affected by different notions of what an outlier is. Participants had the opportunity to perform test trials, as it is common on MTurk, but these trials did not provide feedback.

4.4 EXPERIMENT RESULTS

I collected 4800 observations from 67 participants, who performed an average of 71.6 tasks (sd = 42.4). The median completion time was 10.3s. Figure 4.5 displays the accuracy distribution per task-condition. Accuracy is calculated per stimulus (a scene-condition pair) as the ratio correct/incorrect. In the following sections I examine the odds of a participant

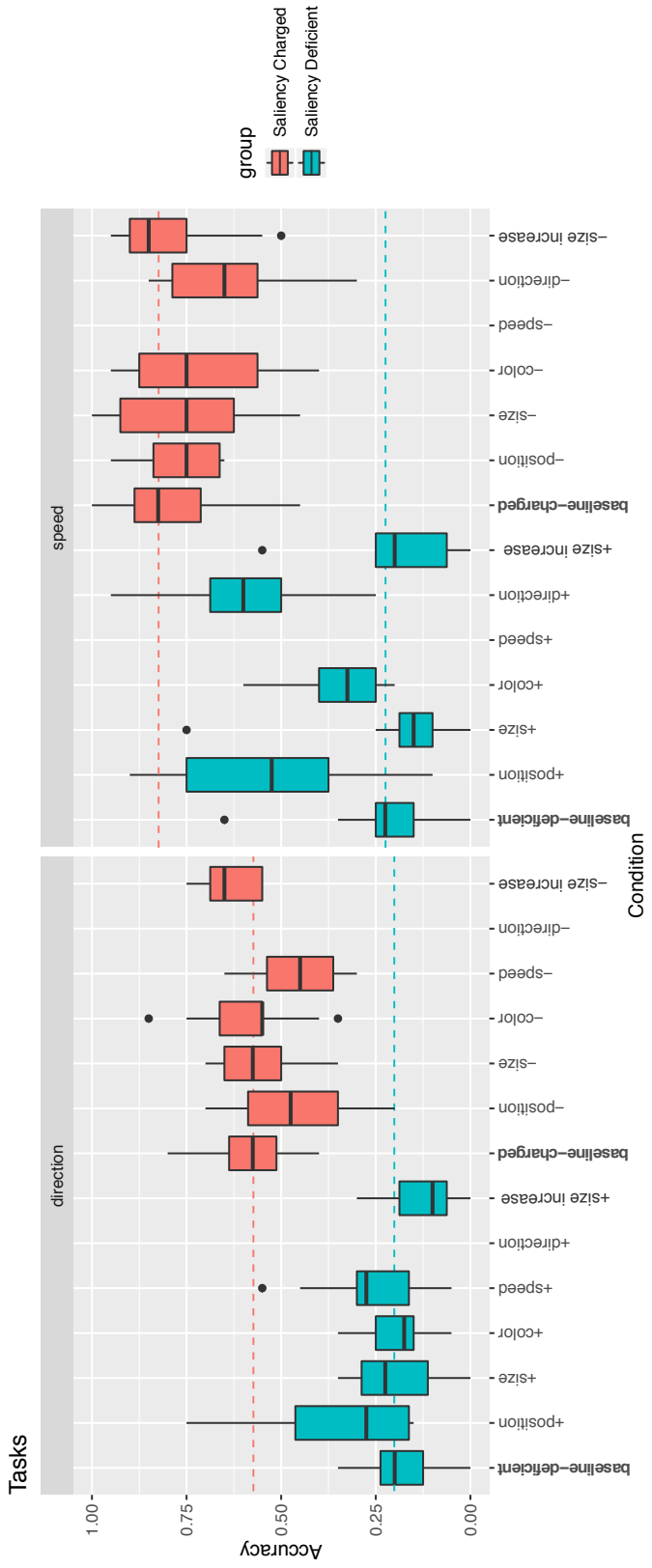


Figure 4.5: Distribution of accuracy for each condition. For each stimulus, accuracy is calculated over 20 judgments. There are 10 stimuli per condition, one for each scene.

selecting the outlier and which features contributed most to incorrect selections.

Channel Contributions

I used the R package *lme4* (Bates et al., 2015) to fit a pair of generalized linear mixed models (GLMM), one for each task (speed and direction). The models were specified with a binary response variable (correct = [true, false]) and five binary covariates [salient, non-salient]: position, color, size, speed/direction, and size increase. This model is also known as a binomial logistic regression. In order to account for scene-specific and participant-specific effects, I inserted the variables *scene* and *subject* as random effects. As such, the random impact from scenes that happen to be more or less difficult, or participants that are more or less accurate, is reduced. Figure 4.5 shows the data, and Figure 4.6 shows the model estimates. The null model has an odds-ratio of 1; that is, irrelevant visual dimensions do not influence the probability of an outlier being correctly detected. p-values are computed for each visual dimension with Wald Z-tests. Below I discuss the main findings.

Motion outlier detection is not well supported

The mean accuracy is lower than 25% in the condition *baseline-deficient* in both tasks. This condition is where the motion outlier does not have salient features other than motion. Low accuracy suggests subjects were mostly unable to separate motion from other dimensions in order to correctly identify the motion outlier. In other words, motion detection in multivariate scatterplots suffers interference from irrelevant dimensions.

Accuracy depends on saliency of irrelevant features

Most conditions where the motion outlier had irrelevant salient features recorded higher accuracy. In particular, subjects achieved averages of 78.5% and 58.5% accuracy in the *baseline-charged* condition, in the speed and direction tasks, respectively. Removing one salient feature at a time generally caused a drop in accuracy; conversely, adding one salient feature generally increased accuracy, but not by much, especially in the direction task, which suggests that in crowded displays motion outliers can only reliably be extracted if they have multiple salient features. More generally, animated scatterplots may reliably support only the detection of *global* outliers.

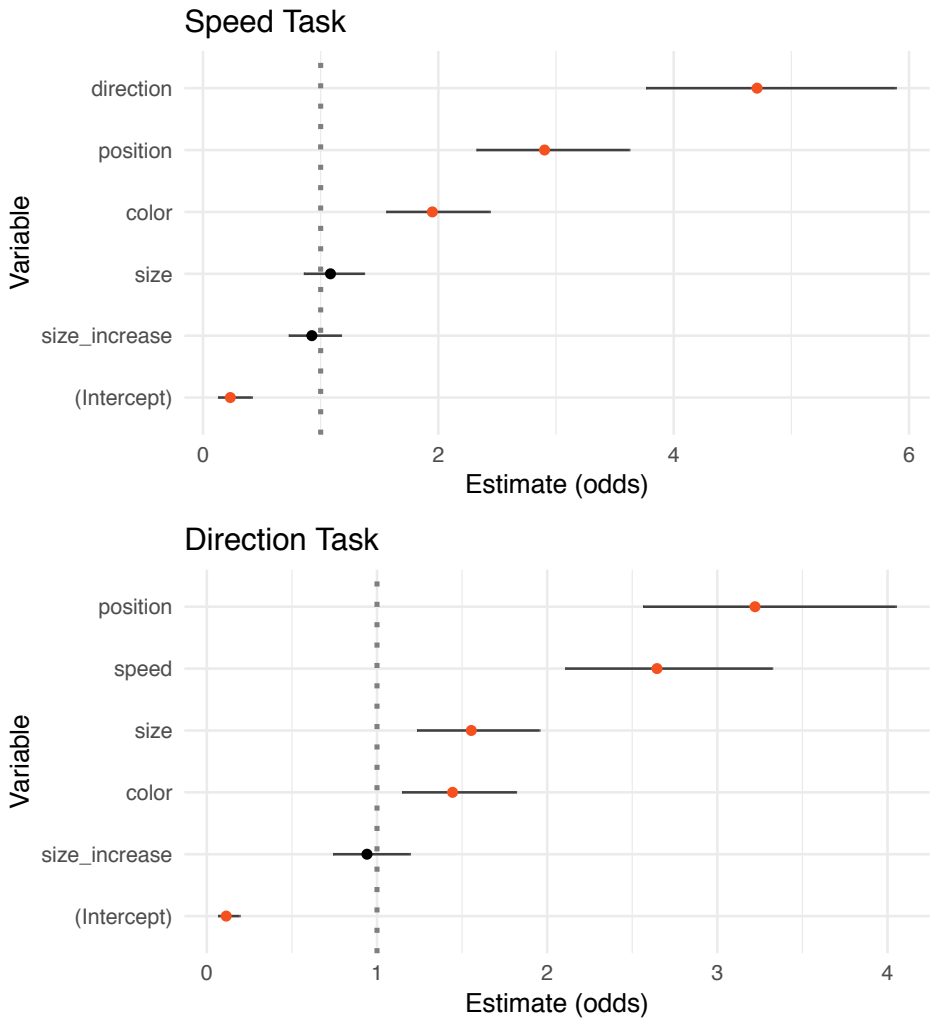


Figure 4.6: Estimates for the effect of irrelevant salient features on the odds of a speed (top) and direction (bottom) outlier being identified. Binary covariates and multiplicative coefficients. Red denotes statistical significance ($p < .05$).

Direction plays the largest role in the speed task

The fitted model indicates that direction saliency accounts for an increase of 4.7 times in the odds of correct speed outlier detection, which corresponds to a shift in probability from 0.19 (intercept) to 0.52. This result is somewhat aligned with previous findings that direction variability degrades searching for a unique speed. Targets with salient direction might have allowed subjects to segment the scene, cancelling some of the noise that impacts accuracy.

Position plays the largest role in the direction task accuracy

Position is estimated to account for an increase of 3.2 times in the odds for the direction task, which is equivalent to a shift in probability from 0.10 (intercept) to 0.26. This result is not trivial: while targets in salient positions (surrounded with blank space) are more visible, they are arguably more difficult to compare, due to their distance from other points. In addition, this result highlights the effect of clutter on this task. The sampling process I used produces a point cloud with a high density center. Points with low spatial saliency are located in these cluttered regions.

Size and color have small influence in the direction task

Both size and color contribute modestly to the outcome. The results contain no evidence of difference between the odds estimate for these dimensions, as their confidence intervals largely overlap. In general, there is a precedence of spatial attributes (position, speed, and direction) over form attributes (color and size).

Size makes no difference in the speed task

Size and size increase did not alter the odds of correct detection in the speed task (these variables have odds ratio approximately 1). This is in contrast to a small, but significant effect in the direction task. It is possible that this can be explained by larger points being perceived as moving slower, which would degrade the performance relative to the baseline; however, the model did not point to a negative effect. It is also plausible that the distribution of values mapped to size did not produce enough saliency. Weber's law predicts a non linear relation between area change and perceived area change, which may have caused points with maximum area to appear closer to the mean and less salient.

Which Features Mislead?

When examining the incorrect choices of participants one would normally expect that the points they selected are close to the target in speed or direction; that is, more incorrect selections should be recorded for faster or more deviant points. This expectation was contradicted by the low correlations observed between task dimensions and selection counts: 0.23 for speed and 0.28 for direction. The correlations were calculated on the subset of non-target points with selection count greater than 0. This suggests that incorrect selections are not necessarily due to the proximity to the outlier value in the target dimension; that is, irrelevant dimensions may be leading participants to make mistakes.

To find which dimensions play a role in the number of times a non-target point is selected I fit generalized linear models (GLM) to the subset of 1,530 non-target points that were selected at least once. Since the observed response variable—selection count—is skewed and lies in the interval $(1, \infty)$ I set the models with a Gamma response variable. The covariates are saliency measures (SSM) on speed/direction and on all other dimensions. I use the saliency measure here because unlike targets, which were made either salient or not, non-target features lie within a saliency spectrum. Likewise, I split position saliency into saliency in the first frame (xy_1) and in the second frame (xy_2).

I included terms for interactions of all saliency measures with speed or direction. In order to make the estimates comparable and easier to interpret all covariates were standardized (zero-mean and unit-variance). In Figure 4.7, the effects are multiplicative; that is, $y = \beta_0 \times \beta_1 x_1 \times \beta_2 x_2 \times \beta_{12} x_1 x_2 ...$, where β_0 is the intercept, β_i are fixed effects, β_{ij} is an interaction term, and x_i are dimension values. The null model states that β is equal to 1, which equates to a visual dimension having no effect on the selection count. p-values are computed for each visual dimension with Wald Z-tests. The interaction plots in Figures 4.8 and 4.9 depict the curve that represents the relationship between speed/direction and the response variable (count), and how this curve is changed as a function of the interacting variable. Below I report the main findings.

Position and direction saliencies boost the effect of speed

In the speed task, the model estimates reveal, not surprisingly, that speed is a confuser and that the interactions of speed with direction saliency and position saliency in the first frame are significant. The interaction terms are positive: the misleading effect of speed increases as a function of the saliency of these irrelevant dimensions. In Figure 4.8, this is shown as an increase in slope: when the values of either direction saliency or position saliency increase by one standard deviation, the effect of speed on

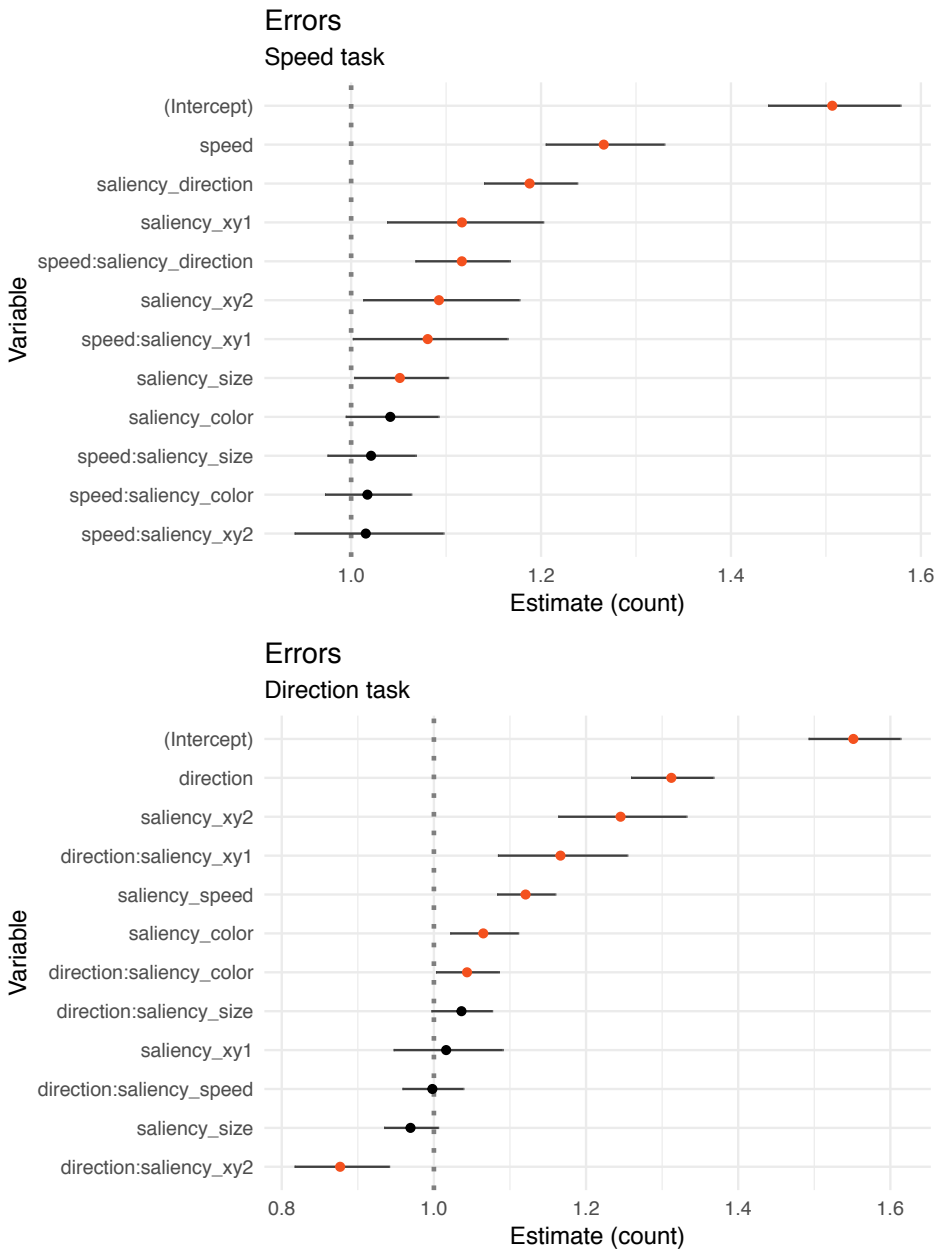


Figure 4.7: Estimates for the effect of feature saliency on the number of times a non-target is selected (erroneously). Red denotes statistical significance ($p < .05$).

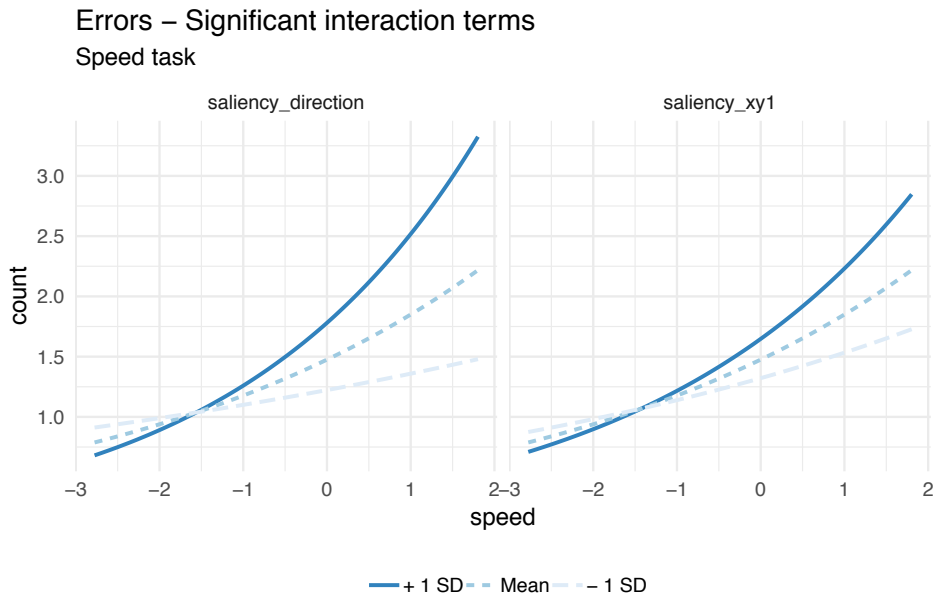


Figure 4.8: Interaction plot depicting the modulation of the effect of speed and direction by irrelevant features in the speed task.

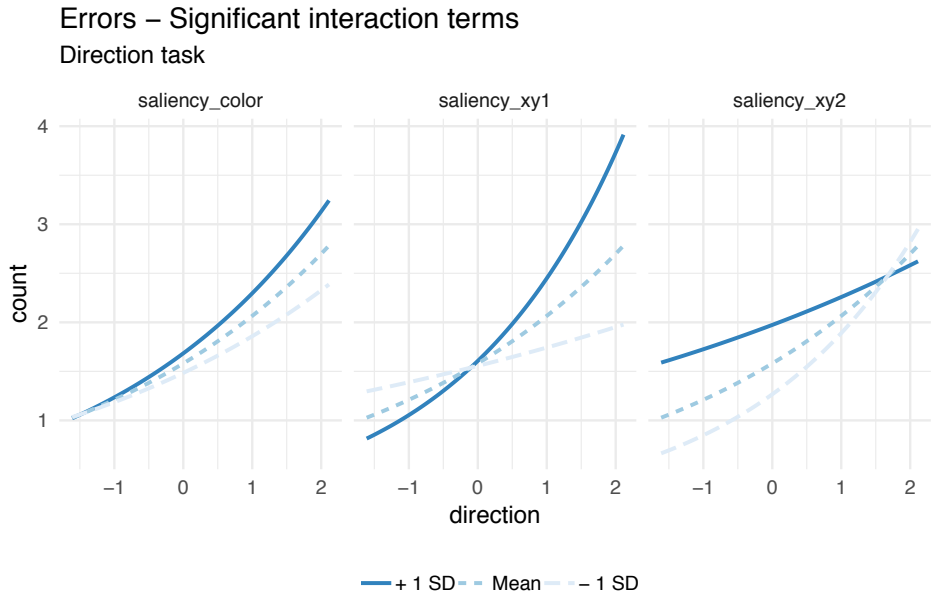


Figure 4.9: Interaction plot depicting the modulation of the effect of speed and direction by irrelevant features in the direction task.

the response becomes steeper. In practice, this indicates that fast points moving from blank regions and in unique directions tend to be mistaken for true speed outliers. This result is aligned with the channel contributions observed in the previous section: position and direction have the highest impact on the odds of a target being correctly identified.

Position saliency in the first frame and color saliency boost the effect of direction

In the direction task the misleading effect of direction saliency is boosted by position saliency in the first frame. In Figure 4.9 this is seen as a slope increase when the value of *saliency_xy1* increases. Color saliency also interacts with direction, but to a lesser extent. In addition, the effect of speed is significant and independent from that of direction. Considering the results above, it appears that position saliency in the first frame is consistently a major factor for selection. Motion outliers that are inside the point cloud might be overlooked if there is a confuser departing from a salient position.

Position saliency in the second frame degrades the effect of direction

Surprisingly, position saliency in the second frame has a negative interaction with direction. This appears in Figure 4.9 as a decrease in the slope of the curve when *saliency_xy2* increases. Participants are thus less likely to erroneously select a point moving in a salient direction the more salient its final position. I hypothesize that this effect may be due to points moving out of the cloud clearly having direction perpendicular to the trend. As participants were instructed to select "the point that moves in the most deviant direction", they may have been looking for points that were in the opposite direction of the mean. Points moving in the opposite direction would likely be inside the cloud, not moving out of it.

Replays

In this section I examine the number of times participants viewed the animation before selecting their answers. I analyze the distribution of correct and incorrect selections across the three possible values for number of views. Figure 4.10 shows this distribution split by task, condition, and whether the trial was completed correctly. Due to the study being deployed on Mechanical Turk, I am unable to separate divided attention from task difficulty as the cause for replays. A reproduction of this experiment in a controlled setting is necessary for establishing a causal relationship.

Figure 4.10: Distribution of number of views divided by task and condition.

Overall, there is a prevalence of a V-shaped distribution, suggesting that participants were more likely to watch the animation either the minimum or maximum allowed times. In the saliency charged group, speed task, there is a clear pattern of correct answers coming more often from 1-view judgments. This pattern is not present in the direction task. In the saliency deficient group we see the opposite pattern: correct judgments are more likely to come from 3-view judgments, with a few exceptions; namely, targets with salient position in the direction and speed tasks and with salient direction in the speed task seem to require less effort than targets in the other deficient conditions. These patterns are consistent with the coefficients found in the above analyses, suggesting task difficulty may be behind them. The V-shaped symmetrical pattern also appears in incorrect answers, especially in the direction tasks, suggesting confidence in wrong selections.

4.5 DISCUSSION

Motion outlier detection was found to be unreliable in multivariate animated scatterplots. The accuracy of motion outlier detection is degraded in the absence of other salient cues. This suggests a level of interference of spatial (position) and form (color, size) encodings over motion, and between the individual components of motion (speed and direction).

Furthermore, the experiment produced evidence that while people were selecting outliers based on the relevant features (speed, direction), irrelevant features may have acted as "boosters," leading people to select the wrong target. I hypothesize that this may be due to people's attention getting caught by near-outliers that have high global saliency; since the animation is short, they would not have enough time to revise a first impression.

Spatial saliency, which is closely tied to clutter, had a large impact on accuracy in both speed and direction tasks. Here, I emphasize the distinction between occlusion and clutter. I inspected the stimuli for occlusion and adjusted the z-order of elements to prevent small points hiding under larger points. Instead of an effect due to inability to *see* the targets, I believe the effect is due to a difficulty of *allocating attention*, in the sense of feature congestion: as the feature space becomes crowded there is less chance for a single object to stand out (Rosenholtz et al., 2010).

The results suggest that it may be possible to predict scenes where outlier detection is difficult on the basis of saliency measurements. A linear model with a binary response variable and feature saliency coefficients such as the one I fit can output the odds of correct detection given a "scene." A linear model of saliency (for clutter measurement) was used also by Rosenholtz et al. (2010). A threat to the generality of this approach is the fact that the statistical saliency model is invariant to scale (due to the use of Mahalonobis distance); for instance, points mapped to a very narrow color range yield the same saliency values as if they were mapped to a wide color range.

At a more general level, the results expose a failure of mapping data outliers to visual outliers, which I refer to as a *saliency deficit*. A data point or a group of data points is saliency deficient when its importance in the data space is not reflected in the visualization due to a lack of saliency. Saliency deficit is thus a condition of imbalance between data and visual importance. In Kindlmann and Scheidegger's (2014) algebraic model, such a failure is classified as a violation to the visual-data correspondence principle: important changes in the data should yield important visual changes.

The notion of saliency deficit is task dependent: here I examined motion outlier detection, but it is possible that other tasks in other visualization types may suffer from the same problem. Interference between visual channels is not new in visualization research, which often points to the theory of separable and integral dimensions (Garner, 2014). When a pair of visual dimensions is integral, information from an individual dimension cannot be accessed easily. However, these studies have been traditionally restricted to the task of class-separation and with static features. For instance, in a point cloud with varying hue and size, it is not

easy to separate points based on each dimension independently. Motion has generally been regarded as a superior dimension, immune to interference from static features.

It is plausible that the mechanism behind saliency deficit depends on the number of visual channels employed. That is, the more visual channels, the harder it becomes to perform tasks that rely on saliency along a single dimension. This sends us back to the feature congestion model of clutter, which predicts difficulty in creating salient targets within a crowded feature space. In order to assert this mechanism with confidence, further research needs to examine this effect with a variable number of visual channels.

4.6 LIMITATIONS

The present experiment could be extended in many ways. I controlled the outlierness of the targets, animation speed, and the distribution of the features and their correlation in order to isolate the effect of feature saliency. This imposes limitations on the scope of inference of the experiment. It is plausible that interactions exist between the controlled factors and the response variables; in particular, as the outlierness of the target increases, the effect of other features probably decreases. The effect of animation speed may be complex: fast transitions may make tasks more difficult, but studies in the topic of change blindness have found that large changes can also go undetected when introduced gradually (Simons et al., 2000).

I have investigated only positive outliers. Due to a known asymmetry in motion target detection—it is easier to find fast targets among slow distractors than the inverse—I cannot extend the conclusions to slow outliers.

As stated in Section 4.5, I would like to measure accuracy in an experiment where the number of irrelevant dimensions is manipulated. This could generate insights on the number of dimensions beyond which some tasks start to lose accuracy. Likewise, it would be interesting to measure the effect of motion on other encodings. Finally, it is possible that the estimates for size and color do not generalize to other ranges. In particular, the color saliency may vary depending on the direction of the colormap (bright to dark or inverse) and the background.

4.7 SUMMARY

In this chapter I reported the results of a controlled experiment designed to test the effect of irrelevant visual dimensions on the accuracy of motion outlier detection in multivariate animated scatterplots. I found that color, size, position, speed, and direction influence the accuracy with which people detect the fastest or the most deviant data point. In particular, spatial visual dimensions, such as position, speed, and direction have larger influence than form attributes, such as color and size. Mean accuracy in detection of speed outliers was higher than 75% only when targets had multiple salient features. When detecting direction outliers, mean accuracy was never higher than 30% when targets lacked salient features.

These results suggest a saliency deficit effect that prevents motion targets from being detected accurately when their overall saliency is low; as a consequence, animated scatterplots should be used with caution if outlier detection is a critical task. It is plausible that saliency deficit may affect tasks in other multivariate visualizations. Models of task accuracy that rely on foundational variables, such as saliency, in conjunction with models of user intent may inform the introduction of automated interventions when the predicted accuracy of a task given a plot is low.

5 DISCRIMINABILITY

Visualization research has its origins in HCI, statistics, vision science, and design, just to name a few disciplines. Each of them contributed methods that together define how research is done in the present day. Of importance to this chapter is the role that human-centred design and experimental research with human participants, both coming from HCI, have on the way visualization research and products are tested.

To demonstrate the predominant process, let's examine Munzner's influential nested model for visualization design and validation (Munzner, 2009). This model prescribes nested steps for visualization design and methods for validating each step: a) domain problem and data characterization; b) operation and data type abstraction; c) visual encoding and interaction design; d) algorithm design. At the first level, the designer "must learn about the tasks and the data of target users in some particular domain". Also known as elicitation of requirements, this phase borrows methods from human-centred design, such as ethnographic studies. I argue that, in practice, this step is conflated into learning about the tasks of the users *in detriment* of the data. We don't need to leave Munzner's text to observe this happening. The output of step (a) is a "set of questions asked about or actions carried out by the target users for some heterogeneous data". Note how the characterization of data disappears from the output. In the next level, operation and data type abstraction, the output is a description of operations and data types. Characterizing data is thus reduced to descriptions of its type. This gap gives rise to what I call *exemplary datasets*, a small collection of datasets taken as representative of the population and which the rest of the design process becomes based upon. The outcome of the design process is commonly overfit to these few datasets. In fact, many visualizations are tested against the same datasets used for their design.

In statistical terms, the exemplary dataset is a single outcome of size N of the random process that governs the data. This outcome is more or less characteristic of the process depending on the complexity of the process, which can be measured by the number of parameters and the variance. That is, the more parameters, the broader the universe of possible datasets and the less representative our example. The more variance, the lower should be our expectation that future samples will resemble our example.

In many cases, overfitting should not be a problem. Custom visualizations that appear in journalism or are commissioned by institutions and whose purpose is to communicate, or to expose, do not need validation because they are not meant to be used with other data. Overfitting affects visualizations or techniques that are expected to be effective over a large collection of datasets. For instance, teams that develop custom tools for clients with specific needs should account for large variations in the data if their tools are to last. If the designers model products after "common" data, how can they guarantee that people can make sense of rare data when it occurs?

5.1 THE INDUCTION PROBLEM

Karl Popper discusses in great depth the difficulty of proving universal scientific statements through the induction method (Popper, 2005). He argues that there is no logical basis for the argument that knowledge can be derived from experience, for experience has no limit, so we are never able to exhaust the observations needed to prove deductively that a statement is universal. Popper then proposes falsifiability as a criterion for deciding if a statement is or is not scientific. Under this criterion, a scientific statement has to allow one to reject it based on observations. For instance, the universal statement "all swans are white" can be rejected upon the observation of a single black swan.

Most visualization research is not in search of universal statements and being merely falsifiable does not guarantee a claim will even be accepted for publication. We rely on the strength of the evidence to legitimate findings. When new layouts, visual representations, or applications are proposed in our field, claims are made about their efficacy that span both a universe of users and a universe of datasets. How can we accept the validity of such claims? The more datasets and the more people are observed in our experiments, the stronger the evidence in favour of a contribution. In fact, the community has given increasing importance to the number of people a technique is tested with, and the background of these people: whether they are students or professionals, for instance. However, we have not recognized that failure to characterize data comprehensively imposes limitations to the scope of inference of visualization research, and threatens its validity.

It is not a stretch to say that data is the forgotten random variable in visualization research. Rare are the examples of research that vary data to an extent that enable us to infer that the proposed method generalizes to other datasets. For instance, Rodrigues and Weiskopf (2018) presented a layout for visualizing highly skewed distributions, motivated by citation

data, which often contains a few data points with tens of thousands of counts and most points with counts near zero. They tested the technique with 4 visualization experts and 3 datasets and concluded that it's "well prepared to visualize countable data samples for data sets with a large range of frequencies". When backed by observations of few datasets, claims that a technique or tool is adequate to visualize data are just as fragile as those that are based on observations of few people using the technique.

Whether or not authors should test and scope their work more rigorously is open to debate. Research is incremental and the community is free to collect more observations of the technique and attempt to discredit it. From the perspective of the author, it may make sense to delimit a scope within which it is hard to prove a claim wrong. In fact, Popper acknowledged that one can amend a statement in order to prevent falsification ("all swans are white except Australian swans"). These he calls *ad hoc* hypotheses. In a similar fashion, a researcher can state the limits wherein the technique is deemed good.

5.2 VISUALIZATION DISCRIMINABILITY TESTS

The culprits for undertesting in visualization research and practice are the current evaluation methods. The most scalable method at our disposal is crowdsourcing, and it may not be scalable enough because of the cost. If a new technique can be deployed in a production environment with real users, collecting field logs could be a suitable method. Laboratory experiments, expert evaluations, and field observations are all less scalable methods.

I propose a combination of *data simulation* and *quality measures* to perform stress tests on visualization techniques. Simulation can solve the problem of data characterization by forcing designers and researchers to document the data parameters and boundaries wherein the visualization is expected to produce high quality plots and by generating comprehensive test sets. The most prominent use of simulation in the VIS community has been through the VAST challenges (Cook et al., 2014), which are based on large synthetic datasets. In this case, however, we have a single synthetic dataset and the challenge is to build a tool capable of extracting the answer to a problem. In the spirit of what I propose, an interesting twist would consist in publishing a set of datasets that covers a broad area of the parameter space and asking for a tool that is capable of answering a problem question given *any* of the datasets.

Quality measures constitute the other leg of the stress tests. While simulation forces us to specify data, quality measures require us to spec-

ify what the tool or technique is hoping to achieve, and how success is measured. This sounds trivial, but there is anecdotal evidence that researchers often do not know how to state what the contribution of a tool is. For instance, the aforementioned non-linear dot plots, by Rodrigues and Weiskopf (2018), are motivated by lack of bandwidth found in common histograms when the distribution is skewed. In other words, we can state that a large family of skewed datasets produces the same histogram, which points to image similarity measures as a possible way to verify that non-linear dot plots are effective over a large spectrum of simulated, skewed datasets.

Here, I propose one type of stress test that is intended to evaluate the perceptual scalability of visualizations: **discriminability tests**. These are semi-automated tests based on the notion that when we state that a design is more scalable than other, we are saying that this design allows us to distinguish a larger family of datasets as N grows. We can define, thus, scalability in terms of this discriminability criterion:

SCALABILITY The relation between dataset size and discriminability.

We can then make this general definition more useful by specifying a data scope and a concrete way to measure discriminability.

DISCRIMINABILITY Given a collection of datasets, the average perceived distance between the corresponding visualizations.

Alternatively, discriminability could be defined in terms of the average data distance needed to produce a just noticeable difference in the visualization. Or, given a seed dataset and corresponding visualization, the effort needed to produce a second dataset (beyond a certain data distance) that yields an ambiguous visualizations. If an intelligent agent is trained to generate such ambiguity inducing dataset pairs, the effort could be measured in terms of model complexity. This ambiguity induction is conceptually the same procedure proposed by Matejka and Fitzmaurice (2017) to generate wildly different scatterplots that have the same statistics.

There are many reasons why a visualization design may lack scalability, the most common being clutter. Under very high clutter, a large family of different datasets will be mapped into very similar images. However, clutter is not general enough. There are many situations where datasets will be mapped to ambiguous low clutter images; for instance, skewed histograms tend to display a few bars on either extreme of the horizontal axis. Discriminability is a more general criterion because it attacks not the resulting image, but the *mapping* of data to image.

In summary, a discriminability test takes as input a collection of datasets, and a visualization function. It outputs a measure of the discriminability of the datasets given the visualization function. A stress test using the discriminability criterion performs discriminability tests at different scales, and outputs a curve describing the relation between scale and discriminability.

In this chapter, I investigate in depth the possibility of an analytical measure of similarity that can match human perceived similarity. With such a measure we could perform large scale discriminability tests, involving not only variation in dataset size, but also in dataset distribution, entropy, etc.

5.3 THEORETICAL BACKGROUND

An alternative to set theory as a foundation for mathematics, category theory is a general mathematical theory of structures and systems of structures. It allows us to define families of structures and see how structures of different kinds are related without having to deal with their details (Marquis, 2015). Using category theory, Kindlmann and Scheidegger (2014) formalized in mathematical terms the minimal quality criteria for visualization, which has appeared previously in the literature in various forms; for instance, the expressiveness and effectiveness criteria of Mackinlay (1986).

Three objects are defined in Kindlmann and Scheidegger's algebraic process: data (D), representation (R), and visualization (V). If r and v are structure preserving maps from D to R and from R to V then their composite $r \circ v$ is a structure preserving map from D to V. The notion of structure preserving maps (homomorphisms), which can be composed, is central to category theory (Cheng, 2008). Morphisms can be depicted as arrows, and their composition as concatenation of arrows in commutative diagrams:

$$\begin{array}{ccccc} D & \xrightarrow{r_1} & R & \xrightarrow{v} & V \\ \downarrow{\scriptstyle \alpha} & & & & \downarrow{\scriptstyle \omega} \\ D & \xrightarrow{r_2} & R & \xrightarrow{v} & V \end{array}$$

The diagram above states that the mapping v acts on a representation of the data to produce a visualization. The maps α and ω are called data and visualization symmetries, respectively. An important consequence of this formulation is the *principle of unambiguous data depiction*. Consider a composition $D \xrightarrow{r_1} R \xrightarrow{v} V$. If a data symmetry is applied on D, the only way for the diagram to commute is through a visualization symmetry on V: $D \xrightarrow{\alpha} D \xrightarrow{r_1} R \xrightarrow{v} V \xrightarrow{\omega} V$. If α was not the identity mapping then

ω cannot be the identity mapping, or the visualization is *ambiguous*. The principle of unambiguous data depiction is satisfied if the following holds: $\omega = 1_V \Rightarrow \alpha = 1_D$, where 1_D and 1_V are the identity mappings. Given a dataset and its corresponding visualization, only the identity mapping on the data should result in the same visualization. Thus, *confusers* are changes in the data that are invisible to the viewer of a visualization.

Similarly, the correspondence principle states that changes in the data are followed by changes of equivalent magnitude in the visualization ($\alpha \cong \omega$); that is, when an important change in data is not followed by a salient change in the visualization, the principle has been violated.

Note that structure preserving mappings are formally defined in other areas of mathematics. In vector spaces, for instance, linear maps preserve addition and scalar multiplication. In category theory we do not care about the specific ways in which structures are preserved and this is convenient to study visualizations, because in visual data analysis structure preservation is task-dependent.

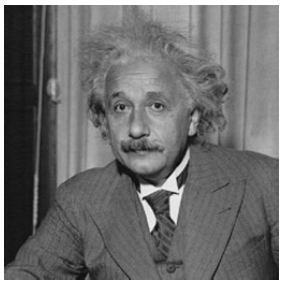
Discriminability tests can be seen as computational tools to verify the principles described above. Given two distinct datasets we can verify the principle of unambiguous data depiction by computing the similarity between the corresponding visualizations. Given two datasets with known data distance, we can verify visual-data correspondence by comparing the data distances with the perceived visualization distances.

A scalability test as defined in the previous section is a test of visual-data correspondence as a function of scale. It is also possible to define scalability tests in terms of ambiguity tests: given a large collection of datasets, we can search for pairs (α, ω) that violate the ambiguity principle.

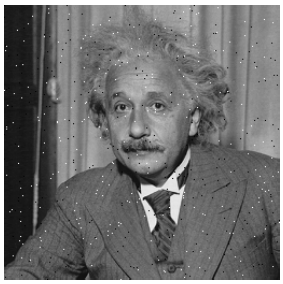
This mathematical representation of the visualization process and its failures suggests that we can *search* for principle violations by varying α and testing ω. α can be varied with random sampling methods or non-random methods such as a parameter sweep, and ω can be tested with image similarity or information theory methods (Rigau et al., 2008).

5.4 STRUCTURAL SIMILARITY INDEX

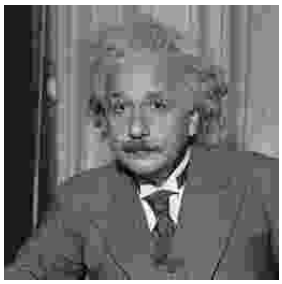
The Structural Similarity Index (SSIM) was developed for quality assessment of compressed images (Wang et al., 2004). Different than previous measures (e.g., mean squared error, and peak signal-to-noise ratio) that assumed that the perception of image quality depends on the visibility of errors, SSIM assumes that image quality depends on the preservation of structural information. As such, image quality can be quantified by a general measure of structural similarity between the original image and



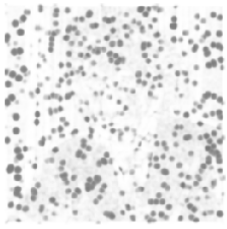
(a) Original



(b) Random noise
MSE=160
SSIM=0.79



(c) JPEG compressed
MSE=117
SSIM=0.71



(d) SSIM similarity maps

Figure 5.1: The mean squared error (MSE) scores the JPEG compressed image (c) as the most similar (lower error value) to the original (a). SSIM correctly scores the image distorted with random noise (b) as the most similar (higher SSIM value). (d) displays the similarity maps computed with SSIM, where gray is error.

the compressed images. While the error-sensitivity paradigm tries to reproduce early-stage, low-level processing of the human visual system, such as thresholding informed by psychophysical experiments, the structural similarity paradigm tries to emulate the hypothesized *function* of the overall human visual system. This function consists in probing the structures of observed objects. Figure 5.1 displays MSE and SSIM scores calculated between an image and two distorted versions of it, one with random (salt and pepper) noise and the other with distortion introduced by JPEG compression. The MSE "prefers" the JPEG compressed image, despite it clearly having lower quality. The SSIM is robust to distortions that do not compromise an image's spatial structures, and correctly rates the image with random noise as the most similar.

The SSIM is defined as the weighted product of luminance similarity, contrast similarity, and structural similarity.

$$SSIM(\boldsymbol{x}, \boldsymbol{y}) = l(\boldsymbol{x}, \boldsymbol{y})^{\alpha} c(\boldsymbol{x}, \boldsymbol{y})^{\beta} s(\boldsymbol{x}, \boldsymbol{y})^{\gamma} \tag{5.1}$$

where $\boldsymbol{x} \in \boldsymbol{R^D}$ and $\boldsymbol{y} \in \boldsymbol{R^D}$ are vectors (of the same size) containing the grayscale pixel intensities of each image. The SSIM calculation normalizes the images with respect to luminance in the contrast similarity calculation, and then normalizes the images with respect to contrast in the structural similarity step. This way, the similarity components are made independent. We can think of equation 5.1 as a pipeline (from left to right) where a feature is subtracted after it has been the subject of a similarity assessment.

Luminance μ is the mean pixel intensity:

$$\mu_{\boldsymbol{x}} = \frac{1}{D}\sum_{i=1}^{D} x_i \tag{5.2}$$

and luminance similarity is defined as follows:

$$l(\boldsymbol{x}, \boldsymbol{y}) = \frac{2\mu_x\mu_y}{{\mu_x}^2 + {\mu_y}^2} \tag{5.3}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are vector representations of the images. Contrast is estimated as the standard deviation of the pixel intensities. Note that the standard deviation σ inherently subtracts the mean intensity (luminance) from the signal.

$$\sigma_{\boldsymbol{x}} = \sqrt{\frac{1}{D}\sum_{i=1}^{D}(x_i - \mu_{\boldsymbol{x}})^2} \tag{5.4}$$

Contrast similarity is defined analogously to luminance similarity:

$$c(\boldsymbol{x}, \boldsymbol{y}) = \frac{2\sigma_x\sigma_y}{{\sigma_x}^2 + {\sigma_y}^2} \tag{5.5}$$

Finally, the structural similarity function operates on the signal normalized by luminance and contrast: $(\boldsymbol{x} - \mu_x)/\sigma_x$. Readers familiar with machine learning will recognize this operation as *standardization*, which yields a z-score. The structural similarity is the correlation (inner product) of these normalized vectors:

$$s(\boldsymbol{\sigma_x}, \boldsymbol{\sigma_y}) = \frac{1}{D-1}\sum_{i=1}^{D}\frac{(x_i - \mu_x)}{\sigma_x}\frac{(y_i - \mu_y)}{\sigma_y}. \tag{5.6}$$

The SSIM is then computed in a local fashion (per pixel) with a 3x3 Gaussian window. This yields a similarity map over the image. The overall image similarity measure, a scalar value, is the mean similarity of this map:

$$\text{Mean-SSIM}(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{M}\sum_{j=1}^{M}\text{SSIM}(\boldsymbol{x}_j, \boldsymbol{y}_j) \tag{5.7}$$

where M is the number of Gaussian windows, $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the images, and $\boldsymbol{x}_j$ and $\boldsymbol{y}_j$ are the image patches defined by each of the M windows. When zero-padding is used $M = D$. Despite the parent-child relation, the acronym SSIM usually refers to Mean-SSIM, and the distinction is rarely in effect. In this chapter, I follow this convention. When the context suggests SSIM is a scalar value, it refers to the Mean-SSIM.

The SSIM is symmetrical, bounded, and has a unique maximum. The index lies in the interval $[-1, 1]$ and a comparison between two identical images will always yield 1.

5.5 MULTISCALE-SSIM

Recall that the SSIM was created to measure the encoding quality of natural images, which depends on the impact of imperfections introduced by the encoding. Clearly, the perception of quality depends on the viewing distance, given that some imperfections are only noticeable at close inspection. In general, we can say that the perception of quality and similarity depends on the *scale* of the image, which varies with viewing distance or image size. Recognizing the challenges of assessing image quality at a single scale, Wang et al. (2003) proposed Multi-Scale SSIM. This technique is a straightforward extension of SSIM where the contrast and structural similarities are computed at K image scales. The original image is subject to low-pass filtering and downsampling by a factor of 2 in each of $K - 1$ steps.

$$MS\text{-}SSIM(\boldsymbol{X}, \boldsymbol{Y}) = l(\boldsymbol{x}, \boldsymbol{y})^{\alpha} \prod_{i=1}^{K} c(\boldsymbol{x_i}, \boldsymbol{y_i})^{\beta_i} s(\boldsymbol{x_i}, \boldsymbol{y_i})^{\gamma_i} \tag{5.8}$$

The weights indexed by i are adjusted according to the desired relative importance of the scales to the similarity judgement. For simplicity, and following Wang et al. (2003), I always set $\alpha = 1$, and $\beta = \gamma$ within each scale:

$$MS\text{-}SSIM(\boldsymbol{X}, \boldsymbol{Y}) = l(\boldsymbol{x}, \boldsymbol{y}) \prod_{i=1}^{K} \Big(c(\boldsymbol{x_i}, \boldsymbol{y_i}) s(\boldsymbol{x_i}, \boldsymbol{y_i}) \Big)^{w_i} \tag{5.9}$$

Throughout this chapter I will use vector notation to communicate the scale parameters; for instance, in the parameter array $W = [w_1, w_2, ..., w_n]$, w_1 is the weight on the lowest scale (largest image), while w_n is the weight on the highest scale (smallest image).

5.6 COMPARING SSIM AND MS-SSIM

To begin assessing the utility of SSIM as a measure of visualization similarity I designed a small sanity test. I chose two visualizations from the Vega-lite visualization gallery (Interactive Data Lab, 2018), a bubble chart and a stream chart, and produced data perturbations of different magnitudes. Then I measured the similarity between the visualizations of the perturbed data and the original visualization. These visualizations have encodings of different nature: point and area. I added also a third set of visualizations, which consists of plots of graphical models of password lists (Zheng et al., 2018). They were chosen because they are dense representations that tend to form distinct shapes.

Figure 5.2 shows the mean squared errors (MSE) computed on the dataset pairs, and both MS-SSIM and SSIM computed on the corresponding visualization pairs. The MSE summarizes the differences in values from one dataset to the other. In this experiment, it represents the baseline or true dataset difference. Most charts of unaggregated data where clutter is not an issue should allow us to recover, with some effort, the MSE between two datasets by mapping the visual marks back to data values and computing the measure. In fact, there are tools designed with the specific purpose of extracting data from existing visualizations (Harper and Agrawala, 2014; Méndez et al., 2016).

The SSIM produced similarity rankings that mirror the MSE rankings in the bubble chart and stream chart cases: larger SSIM values should correspond to lower MSE values. In the dense graph case the true data similarity is unknown, so I'll resort to a qualitative assessment. It is rather clear that two of the plots feature an extremely dense central region that forms a solid red blob, while the other two plots, including the reference plot, feature a more well-distributed pattern. The output of the SSIM comparisons indicates that this notion is not captured by the measure; the graph that is most similar to the reference received the lowest similarity score.

It appears that the similarity of plots is judged at different scales depending on the kind of plot. For instance, dense graphs form distinct global shapes that override local similarity comparisons. Other visualizations, such as scatterplots, may or may not form global shapes. When a global shape is not formed, the similarity judgement is done at a lower level, by scanning the scene in search of differences, a process that is well captured by the windowed calculation of SSIM.

MS-SSIM is built on the premise that *viewing conditions* determine the right scale. I instead posit that at *identical* viewing conditions the scale in which similarity judgements varies with the chart type. As such, I customized the weights as following, so as to give more importance to

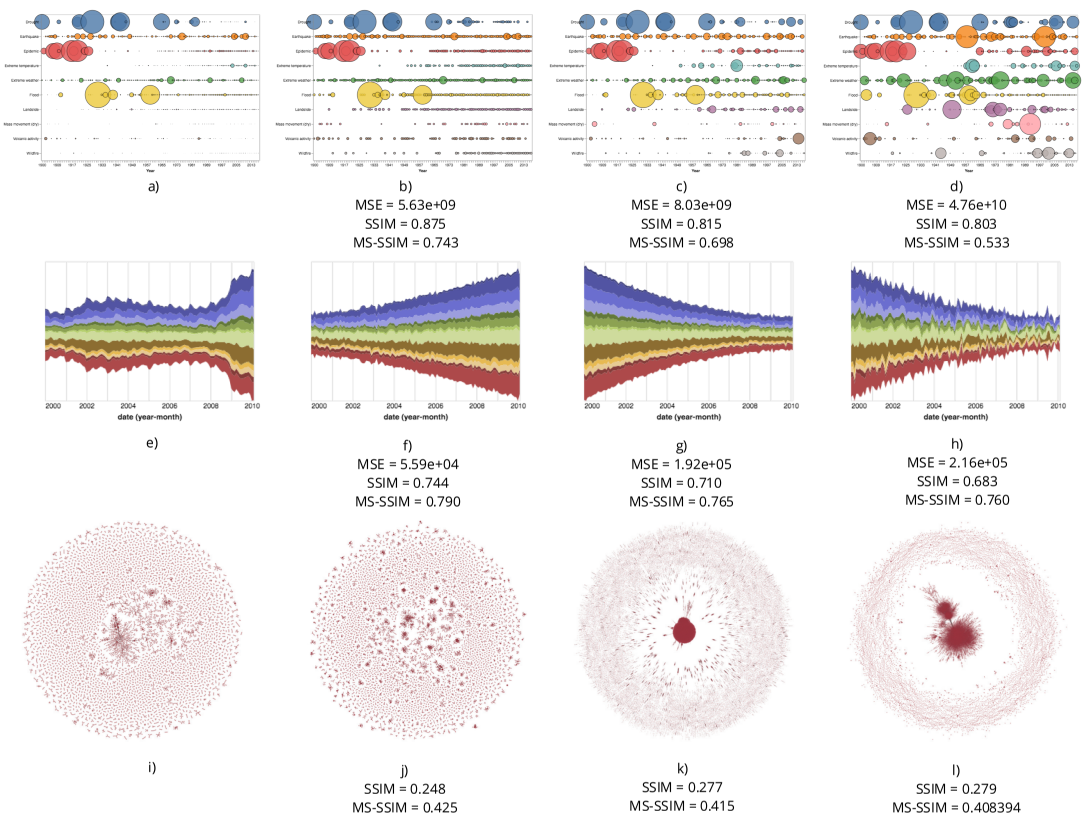


Figure 5.2: Data and image similarity measures: Mean-Squared Error (MSE), Structural Similarity Index (SSIM), and Multi-scale SSIM (MS-SSIM). Leftmost images in each row are the references. Top: global deaths from natural disasters (Vega-lite gallery) and simulated perturbations. Middle: unemployment across industries (Vega-lite gallery) and simulated perturbations. Bottom: graphical models of passwords (Zheng et al., 2018). MSE is inversely proportional to similarity. MS-SSIM weights: [0.1, 0.1, 0.1, 0.2, 0.5].

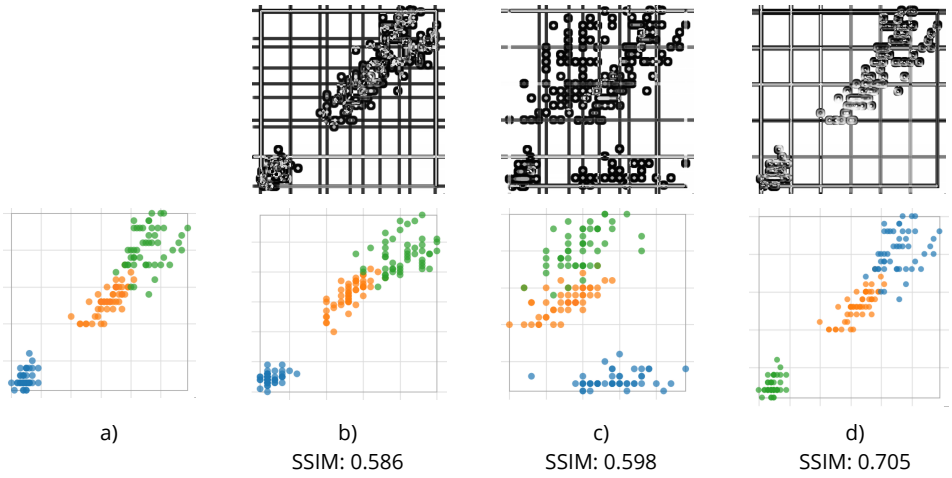


Figure 5.3: Bottom: SSIM measurements relative to (a). Top: Local SSIM values (brighter is higher).

features at the highest scales: [0.1, 0.1, 0.1, 0.2, 0.5]. The resulting scores (Figure 5.2) reflect the correct similarity ordering of the dense graphs. As a bonus, the MS-SSIM scores also comply with the correct data MSE ranks for the stream charts and bubble charts.

5.7 LIMITATIONS

Fundamental limitations arise when the SSIM is applied to data plots. In natural images every pixel counts towards a similarity judgement, although some extensions of the SSIM recognize that some regions matter more than others and attempt to weigh their importance based on saliency (Moorthy and Bovik, 2009), recognized objects (Ninassi et al., 2007), and information theoretic measures (Wang and Li, 2011). In data plots, this characteristic manifests adversely as a hypersensitivity to visual accessories, such as grids and labels. Figure 5.3 displays scatterplots of the Iris dataset that feature a grid. Note how the SSIM values do not correspond to the visual similarity of the plots. Upon close inspection we see that the grids, which are not consistently positioned, contribute disproportionately to the measurement.

In the context of the proposed use of the measure, the discriminability tests, the tester has control over the production of the images, so the hypersensitivity problem can be completely disregarded if we assume that for testing purposes, the plots are generated without grids, labels, and other accessories. Of course this entails that the viewer is capable of separating the accessories from the data mapping when performing a similarity judgement and that such accessories do not hinder the discrim-

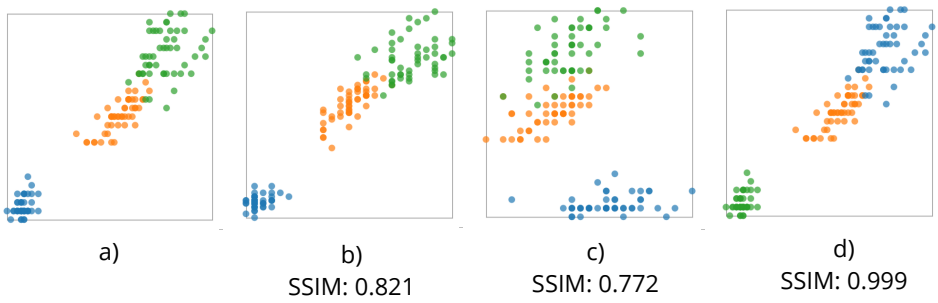


Figure 5.4: Iris dataset without grid. SSIM measurements relative to (a).

inability of the visualizations. That is, by removing the accessories, we artificially make the views less cluttered. This brings us to an important point regarding the target of discriminability tests and, consequently, the similarity measures: they are not intended to measure and test the *clutter* levels of the visualization; instead, the tests target the discriminability of the *mapping*, which precedes concerns with clutter due to labels, grids, and other annotations. Clutter is only a factor when it arises from the data-visual mapping.

Better measurements are achieved by simply turning the grid off (Figure 5.4). However, this figure illustrates a more complicated limitation. The scatterplot labelled (d) is a clone of (a) that had the color mapping inverted (blue became green, and vice-versa), therefore, (d) in fact depicts the most different dataset to (a), contrary to the SSIM value, which places it as the most similar to (a). The SSIM operates on grayscale images and it is not capable of capturing changes in hue.

The color limitation does not affect color encodings of numerical, continuous data attributes, which employ color schemes that vary luminance and saturation, such as the Viridis color scheme. It affects exclusively visualizations that use categorical color mappings, which normally use nearly equiluminant color palettes. In the next section I propose a modification to SSIM that addresses its "color blindness".

5.8 COLOR-SENSITIVE SSIM

The basic idea behind my modification of the SSIM is to segment the image in a manner that produces independent layers, each of which stores objects of a single hue, then average the layer-wise similarities. This strategy exploits the fact that categorical color palettes are often designed to maximize the perceptual distance between the colors. For instance, if we know that the color palette has 3 values, a segmentation of the image based on 3 regions of the hue spectrum of equal size is likely to yield the

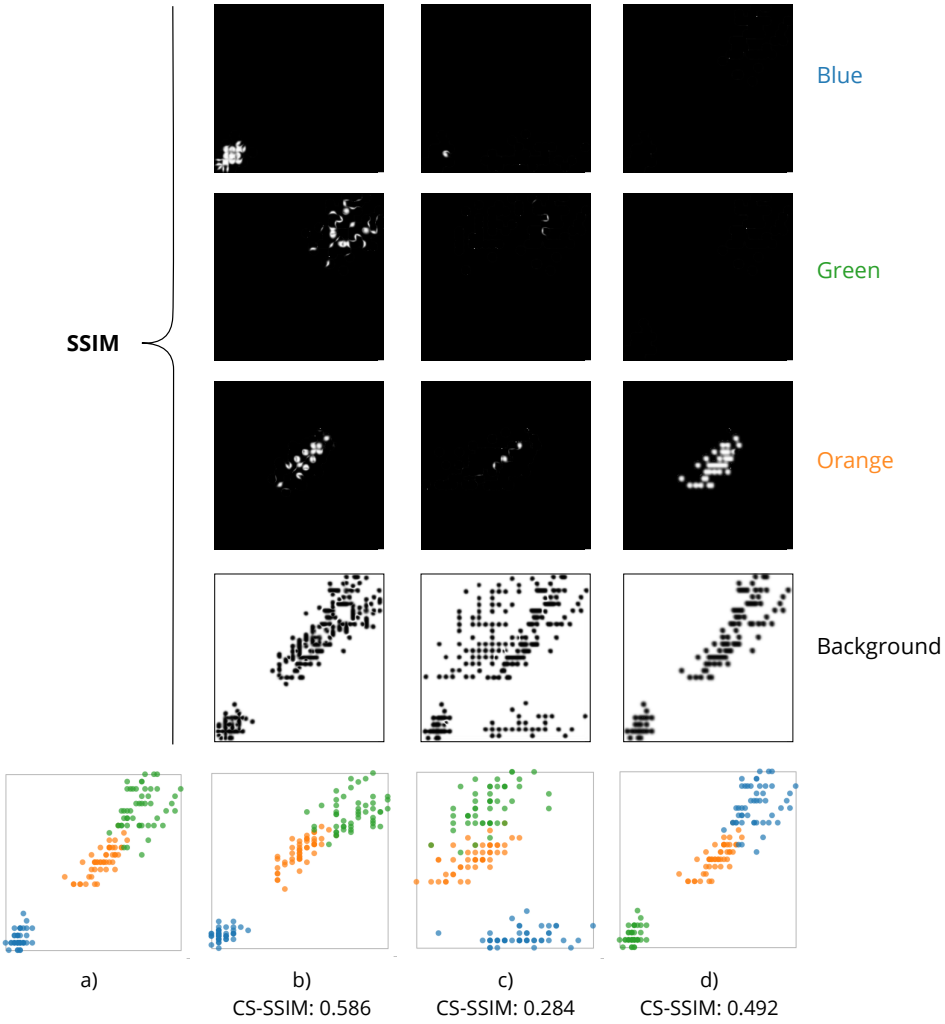


Figure 5.5: Color-sensitive SSIM applied to Iris scatterplots. On top, the decomposition of the similarity measure into color layers and background. Similarity is encoded with brightness (more similar is lighter).

object segregation that we need for computing the correct similarity. The color-sensitive SSIM is thus defined as:

$$\text{CS-SSIM}(x, y) = \sum_{k=1}^{L} \frac{\text{SSIM}(x_k, y_k)}{L}, \tag{5.10}$$

where L is the number of layers. For convenience, I obtain the HSV representation of the images, which allows the color segmentation to take place on a single dimension (hue). The color layers are defined as follows:

$$\boldsymbol{x}_{ki} = \begin{cases} x_i, & \text{if } h_{ka} < x_i < h_{kb} \\ z, & \text{otherwise,} \end{cases} \tag{5.11}$$

where x_{ki} is a pixel in layer k, h_{ka} and h_{kb} are the upper and lower hue bounds that define layer k, and z is the background color. As a result, each layer is obtained by replacing every irrelevant pixel with the background value.

Remember that the SSIM is computed locally with a Gaussian window followed by pooling. Pooling the local SSIM ad-hoc in each layer would result in overrepresentation of the background; instead, the background is cancelled by computing the pooling step as a weighted mean with the following weights:

$$\lambda_{kj} = \mathbb{1}(x_{kj} \neq z \vee y_{kj} \neq z) \tag{5.12}$$

The indicator function above works as a boolean mask that cancels overlapping background pixels. The weights are applied to the calculation of the mean in Equation 5.7:

$$\text{Mean-CS-SSIM} = \frac{1}{L} \sum_{k=1}^{L} \frac{1}{\sum_{j=1}^{M} \lambda_{kj}} \sum_{j=1}^{M} \text{SSIM}(x_j, y_j) \lambda_{kj} \tag{5.13}$$

While the effect of the background is undesirable in the layerwise comparisons, the background should be taken into account; therefore, I define a separate layer for the background. As such, $L = |C| + 1$, where C is the color palette. Figure 5.5 illustrates the application of CS-SSIM on the downsampled Iris scatterplots (scale .25).

This layerwise strategy is conceptually simple and well-founded, as the semantics of SSIM are preserved. The only difference is that we decompose the image into as many layers as the number of hues. In cases where a uniform division of the hue spectrum does not yield a good color segmentation, the algorithm can be easily adapted to accept a list of hues and threshold band. The problem with this strategy is that it can only be used to compare visualization encodings that share a categorical color mapping.

For instance, suppose we would like to compare the discriminability afforded by a force-directed layout with that of a spectral layout. We simulate a number of datasets and calculate their average pairwise similarity

under each layout. If both representations use a categorical color encoding, these averages are comparable and we can use them to decide which layout produces the most discriminable images. However, if one of the encodings uses a continuous color map, there are two obvious options, none of which is well-founded:

1. Calculate CS-SSIM on the visualization with continuous color mapping.
2. Calculate SSIM on the visualization with continuous color mapping and compare it with the CS-SSIM computed on the visualization with categorical color mapping.

In the next section, I discuss a second strategy for computing similarity in color images.

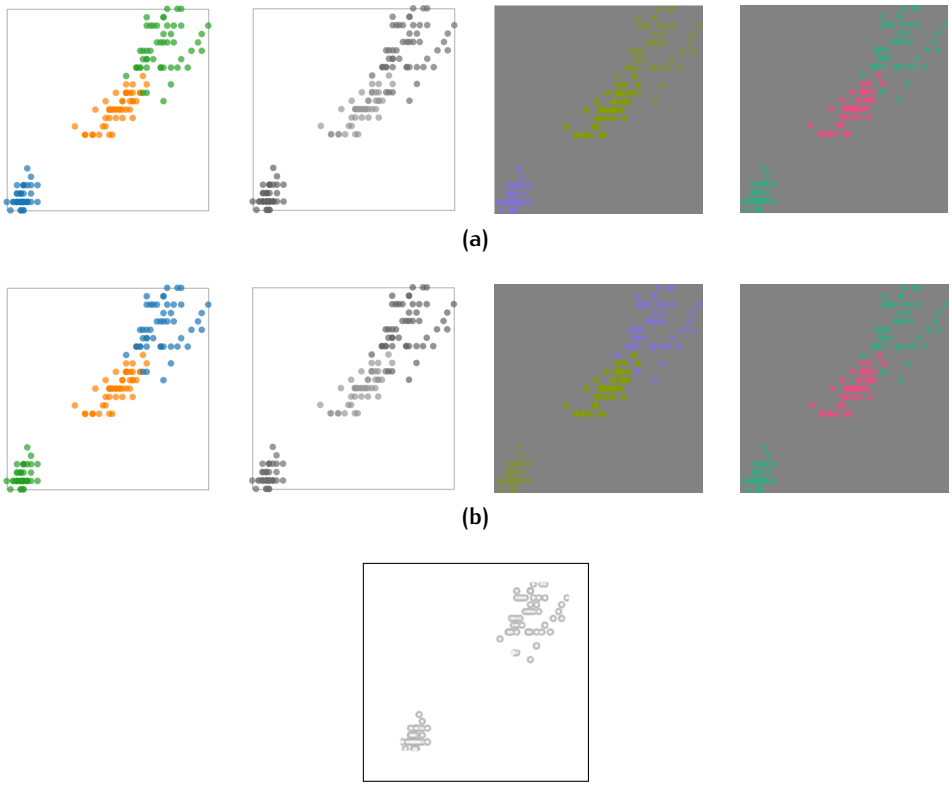
5.9 SSIM ON YUV COLOR SPACE

Considering that in the next sections I will be investigating the discriminability of a broad set of encodings with various color mappings, it is important to establish a more general use of SSIM that can accommodate both categorical and continuous color mappings.

My goal is to introduce *some* sensitivity to color by using a color space where color components are represented independently from luminance. The YUV color space is well aligned with this goal. It consists of a luminance component (Y), and two chrominance components (UV). Black and white images use only the Y component.

I compute the SSIM on the YUV space by simply averaging the similarities computed in each color space component (Y, U, and V) independently. The original SSIM is equivalent to the computation on the Y channel (black and white). The computations on U and V can be interpreted as an assessment of the similarity existing in color structure.

In the pathological example depicted in Figure 5.6, where two groups had their color swapped, this strategy is enough to prevent the visualizations from being scored identical. However, it preserves SSIM's characteristic of being driven by spatial structure. In a data analysis context, there are two plausible readings for the change between images 5.6a and 5.6b. On one hand, the green and blue points could have all been translated across the plane; on the other hand, the points could have remained still and changed the values mapped by color. Assuming the former reading implies very low similarity between these datasets, the SSIM on YUV cannot be expected to capture it, as the CS-SSIM does. The CS-SSIM between the color-swapped scatterplots is 0.492, while the SSIM on YUV is 0.968.



(a)

(b)

(c) SSIM = 0.968

Figure 5.6: SSIM applied on YUV image representations. (a) and (b) are images in their original form, and decomposed into Y, U, and V channels of the YUV color space. (c) is the similarity map resulting from averaging the similarities computed on each channel independently. Note how the color difference in the original images appears in the final similarity map.

Thus, SSIM on YUV is more compatible with the second reading, which implies higher similarity.

However, note that the evidence for the correct interpretation and level of similarity in this case is lacking, so I do not treat any of these behaviors as limitations. I choose to work with SSIM on YUV when color is involved because it is more applicable to a wide range of visualizations. It does not require additional parameters (e.g., color palette, number of colors) and it yields comparable scores regardless of the color map used.

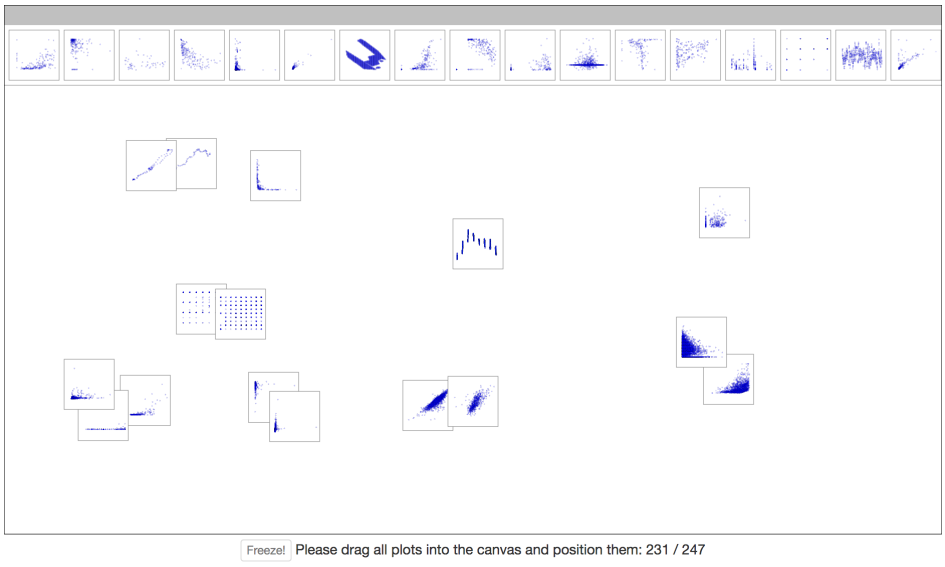


Figure 5.7: Spatial arrangement interface used to collect human similarity judgements of a set of scatterplots. Study participants were instructed to position images into groups according to perceived similarity, and then explicitly delineate group boundaries.

5.10 EMPIRICAL VALIDATION

Scatterplot Similarity

In this section, I compare SSIM judgements with empirical similarity judgements. My goal is to test if a parameterization of SSIM is capable of approximating empirical judgements for a certain visualization type. A positive result in this validation should indicate that other parameterizations can help us approximate judgements for other visualization types, assuming that the judgments will vary mostly with respect to scale and the use of color. If instead we find that no parameter set can approximate well empirical judgements, that should prompt discussion about what factors are involved in similarity perception of data plots. This applies in particular to spatial encodings. If a measure that stems from pixel correlation cannot be tuned to model human similarity judgements, then what visual features are people taking into account?

For this analysis I chose the data collected by Pandey et al. (2016), which consists of human similarity judgements (13 participants) for a set of 247 single-color scatterplots. The scatterplots were produced from 84 real-world datasets, and were selected to maximize diversity using the scagnostics descriptors of scatterplot shape (Wilkinson and Wills, 2008). The similarity judgements were collected with a spatial arrangement in-

terface (Figure 5.7) in which scatterplot thumbnails are displayed in an "image carousel" and can be dragged and dropped into a large, initially empty, canvas. Participants were instructed to arrange the scatterplots into groups according to their similarity, and then explictly mark the boundaries of each group, and finally, assign labels to them. They were told not to worry about within-group or between-group distances; that is, only group membership mattered.

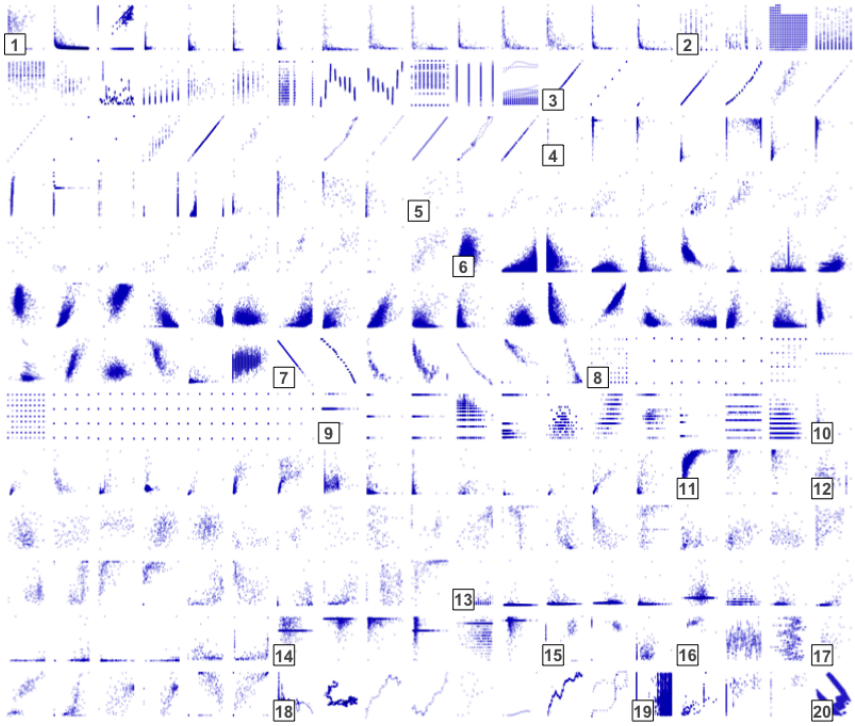
Pandey et al. (2016) calculated the consensus distances for each pair of plots as the complement of their probability of co-occurrence averaged across participants:

$$d_{i,j} = \frac{1}{N} \sum_{k=1}^{N} \left(1 - \frac{c_{i,j}}{min(c_i, c_j)}\right)_k \tag{5.14}$$

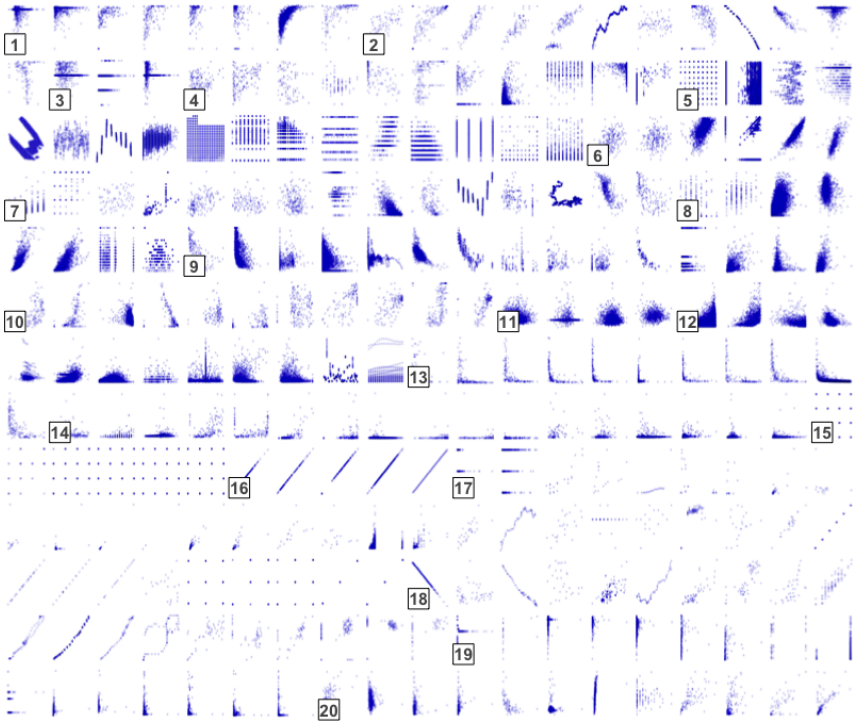
where N is the number of participants, $c_{i,j}$ is the number of clusters that contain both plots i and j, and c_i and c_j are the number of clusters that contain the plots i and j, respectively. Note that the interface allowed participants to assign plots to multiple groups. A hierarchical clustering of the plots based on the consensus perceptual distance matrix was calculated, and it is displayed in Figure 5.8a. This process was repeated with similarity judgements derived from scagnostics scores, the analytical similarity method under scrutiny. The authors used correlation between the pairwise distances to assess the correspondence of scagnostics to empirical judgements, and concluded that, with $r < 0.26$, scagnostics is not a good match.

I consider this comparison method inappropriate because participants were explicitly instructed to disregard distances and the distance calculation above, based on probability of co-occurrence, does not capture fine-grained distance information. For instance, suppose all participants are 100% consistent, and there is a single non-overlapping clustering. In such case, the plot distances are either 1 or 0, and no information is learned about the distances between clusters. With scagnostics, the Euclidean distance is computed for a pair of plots based on the numerical feature vectors that are the output of scagnostics. The matrix holds distance information irrespective of cluster membership. In summary, the data collection procedure of Pandey et al. does not yield 2D embeddings but cluster assignments. For this reason, set membership, not matrix correlation, should be taken as a measure of correspondence.

Here, I compare SSIM and empirical judgements using cluster quality measures, which are traditionally used to quantify the agreement between two independent label assignments on the same dataset. I selected the following measures, all of which assume the ground truth is known: adjusted mutual information (AMI), normalized mutual informa-



(a) Empirical scatterplot clustering.



(b) MS-SSIM scatterplot clustering.

Figure 5.8: Empirical and MS-SSIM clusterings of the scatterplots from the study of Pandey et al. MS-SSIM parameters were tuned to the empirical data via gradient descent.

Table 5.1: Cluster quality measures for clusterings of 247 scatter plots based on MS-SSIM. The quality measures are relative to the clustering based on human similarity judgements reported by Pandey et al. (2016). Each row corresponds to a parameter set ($w_1..w_5$). The parameters in the first row were obtained through gradient descent.

w_1	w_2	w_3	w_4	w_5	**ARI**	**RI**	**AMI**	**NMI**
0.32	**0.73**	**0.82**	**1.00**	**1.00**	**0.20**	**0.90**	**0.35**	**0.51**
0.10	0.10	0.10	0.30	0.40	0.16	0.86	0.30	0.46
0.10	0.20	0.20	0.20	0.30	0.13	0.83	0.25	0.42
0.10	0.15	0.15	0.30	0.30	0.10	0.81	0.22	0.40
0.20	0.20	0.20	0.20	0.20	0.13	0.81	0.24	0.42
0.40	0.20	0.20	0.10	0.10	0.13	0.81	0.26	0.44

tion (NMI), Rand Index (RI), and Adjusted Rand Index (ARI). A summary of the measures' properties is provided in Appendix A, Table A.1. All measures except RI assign values close or equal to 0 to random clusterings and assign 1 to perfect clustering (relative to the ground truth). Change adjusted measures (AMI and ARI) do not exhibit a dependency between the number of clusters and the number of samples; such dependency could boost the score of random clusterings that have many groups.

I compared clusterings based on the multiscale version of SSIM parameterized with six naively defined weight vectors, chosen manually to represent different weight balancing strategies, plus one special weight vector tuned via gradient descent. The weight vectors are presented in Table 5.1, ordered by importance on the finest scales. The parameter set in bold was obtained with the tuning approach described in detail in the next section. The clustering method was fixed to hierarchical clustering under the Ward agglomeration strategy, with even-height tree cuts that yielded 20 clusters (the same number of clusters in the ground truth, although none of the quality measures requires an equal number of clusters).

The results can be seen in Table 5.1. The parameters found through gradient descent achieved the best fitness to the empirical clustering, as observed in all of the quality scores. The plot arrangement resulting from clustering with this best MS-SSIM parameter set is presented in Figure 5.8b, and the corresponding dendrogram in Figure 5.9. The fitted parameters and the plot arrangement comparison tells us much about the protocol used to collect the empirical measurements. First, the participants had only the chance of interacting with thumbnails, forcing them to make high-level perceptual judgements. This fact is expressed in the weights discovered with gradient descent, which clearly emphasize coarser judgements.

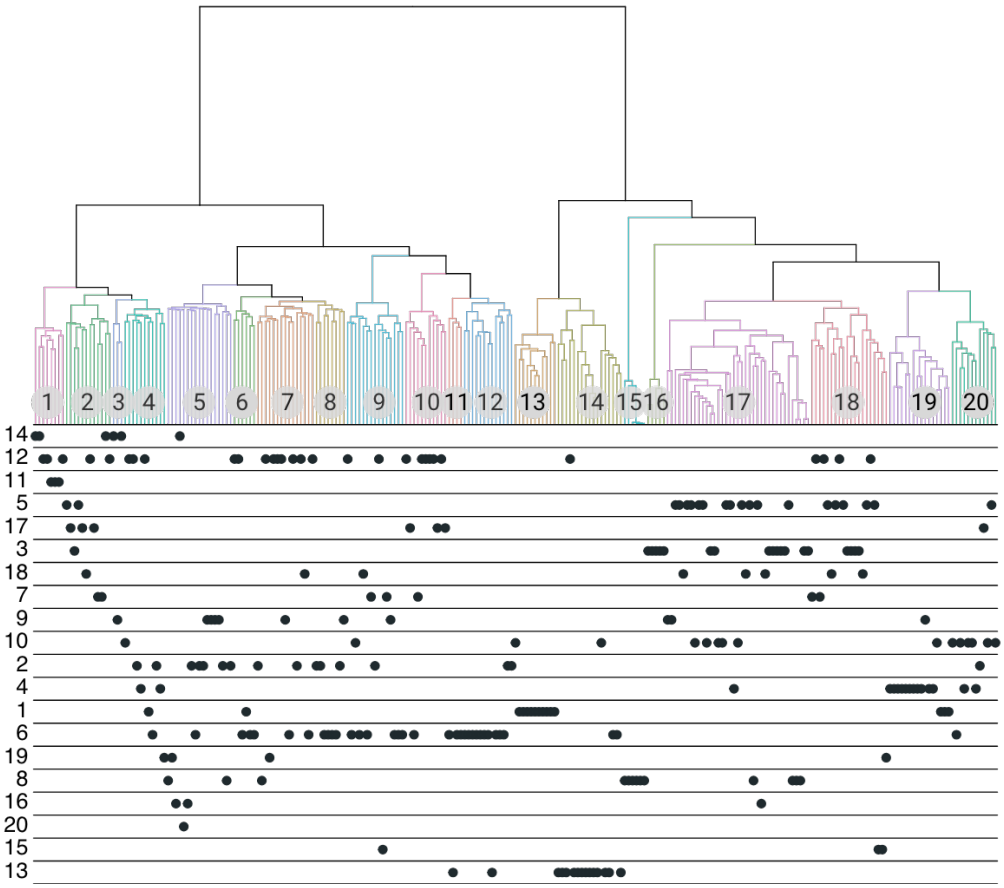


Figure 5.9: Dendrogram representation for the MS-SSIM clustering of Pandey et al's scatterplots. Each row in the bottom represents an empirical cluster, with each dot representing a plot. Dots are aligned with the dendrogram, allowing us to observe how the empirical clusters are disrupted by the dendrogram arrangement. If the clusterings were identical, all dots in each row would be adjacent. Rows are ordered according to leftmost match with dendrogram.

Second, distances were not taken into account. The MS-SSIM clustering imposes a partition between dense and sparse plots (around cluster 13), while the empirical clusters have a fuzzier organization. In addition, some pairs of plots that are very similar are distant in the empirical arrangement. In fact, even if the participants were instructed to optimize distances, the procedure consisted in the organization of 247 scatterplots in a plane, and that would likely have discouraged participants from doing fine-grained adjustments: in addition to much energy being spent, the size of the canvas, in turn limited by the display size, would be a bottleneck.

These limitations prevent me from taking these empirical judgements as an *absolute* ground truth. The most important difficulty arises from the *cognitive interaction problem* (Wang et al., 2003), by which different user goals can result in very different judgements. Participants were not instructed to cluster plots based on *dataset similarity*. In a real-world scenario, analysts are making judgements about the data, with the visualization being a proxy. Some pairs of plots that bear some visual resemblance (in terms of shape) and are in the same empirical cluster, are unlikely to have been found similar if the question was about the underlying data. For instance,  and  have both a T-like shape, but represent very different relationships between the variables. We can attribute much of the difference between the clusterings to this misalignment of goals. Empirical cluster number #5, the one whose elements are spread the most across SSIM clusters, comprises elements with wildly distinct data patterns, but similar density. Density-based agglomeration is still present in the SSIM clustering, but divided according to the position of the point-cloud. Likewise, empirical cluster #11 has plots with similar amount of "ink" but very different spatial arrangements; it is also divided in several pieces in the MS-SSIM clusterings.

5.11 TUNING

In this section I describe the development of a tuning procedure for the multi-scale SSIM. The goal of this procedure is to adjust the scale weights so as to minimize the discrepancy between SSIM similarity and a set of empirical judgements. As no hypothesis is being tested, the procedure is relatively free from bias, and these empirical judgements can be performed by a designer, by a group of colleagues, or deployed on Mechanical Turk.

I assume a visualization designer or engineer in her testing workflow should be able to determine whether or not her similarity judgements suffice. For instance, if the product is being designed for a broad audience

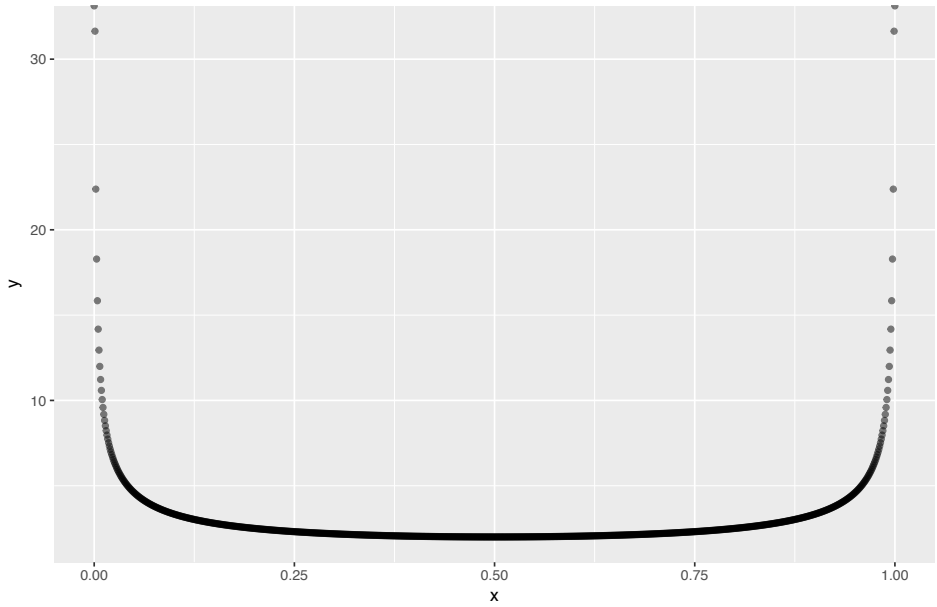


Figure 5.10: Regularization loss function used for gradient descent tuning of MS-SSIM parameters.

and analytical needs, then judgements from a large sample are advised. If the product is being designed for a specific audience in a narrow problem domain, then the designer has access to the audience and can collect judgements, or has learned enough to the extent she can perform the judgements on their behalf, knowing that the audience's interpretation of the visualization will not deviate significantly from the expectation.

For the tuning, I used a stochastic numerical gradient descent algorithm, whose code is presented in Appendix B, Listing B.1. The algorithm, at each iteration, evaluates the gradient of the loss function with respect to the current parameters, then updates the parameters in the directions that reduce the loss.

Let's define a dataset of images $x_i \in R^D$, and a similarity function $s : R^D \times R^D \to R^1$. With the multi-scale SSIM, s has the following form:

$$s(x_i, x_j) = \text{MS-SSIM}(x_i, x_j, W) \tag{5.15}$$

The above equation can be read as the similarity of x_i and x_j given the vector of weights W, which determines the importance of each scale to the overall similarity score, as seen in Section 5.5. Next, let's the define a binary function that takes an image triplet (x_i, x_j, x_k) and decides whether x_i is more similar to x_j than x_i is to x_k:

$$f(x_i, x_j, x_k) = \mathbb{1}(s(x_i, x_j, W) \geq s(x_i, x_k, W)) \tag{5.16}$$

This equation embodies a triplet matching task and enables the definition of a loss function for comparison of SSIM scores with a ground truth that is *independent* of the protocol used to collect the ground truth judgements. For example, the judgments could be collected using triplet matching, triplet discrimination, spatial arrangement, or pairwise ratings on a Likert scale. Compare that with a loss function based on distances, such a matrix correlation: unless the judgement protocol yields a spatial embedding, the comparison with SSIM, or any other analytical measure, would be difficult.

The loss function is defined as follows, where f_{ijk} is an abbreviation for $f(x_i, x_j, x_k, W)$, the SSIM binary label, and Y_{ijk} is the ground truth label:

$$L_{ijk}(W) = \sum_{f_{ijk} \neq Y_{ijk}} \Big(s(x_i, x_j, W) - s(x_i, x_k, W) \Big)^2 + R(W) \tag{5.17}$$

The loss defined in the equation above is composed of two terms, the data loss and the regularization loss. The data loss is simply the squared difference between the similarity scores when they are wrong. For instance, if $s(x_i, x_j, W) = 0.8$, $s(x_i, x_k, W) = 0.6$, and the ground truth is $s(x_i, x_j, W) < s(x_i, x_k, W)$, that is, $Y_{ijk} = 0$, then the loss is 0.2^2. The regularization loss (or penalty) is a function of the weights and embeds our preference for weights in a certain range. In this case, the weights need to be between 0 and 1. The regularization loss has the following form:

$$R(W) = \sum_{i=1}^{|W|} (W_i)^{\alpha-1}(1 - W_i)^{\alpha-1} \tag{5.18}$$

where α is a parameter that controls the steepness of the penalty as the values approach 0 or 1. In Figure 5.10, the shape of this function is depicted with $\alpha = 0.5$.

5.12 DISCRIMINABILITY OF BASIC ENCODINGS

In Section 5.10 I presented a validation of the MS-SSIM against an empirical study of scatterplot similarity. It was useful for understanding the extent to which we can expect human similarity judgements to match MS-SSIM scores, but it did not shed light on the usefulness of discriminability as a quality criterion. We do not know if discriminability scores derived from similarities have any relationship to the effectiveness of visualizations. In this section I seek to fill this gap.

Fortunately, there are a few empirical studies of the effectiveness of visualization encodings. I will base my investigation on the most recent of

Table 5.2: Kim and Heer's experiment was divided into four tasks. Q_1 is a continuous variable.

Read value	What is the Q_1 of the data point A?
Compare value	Which data point has more/less Q_1?
Find maximum	Which state has the data point with the highest Q_1?
Compare averages	Considering all data points for the State, which of the following two States has greater average Q_1?

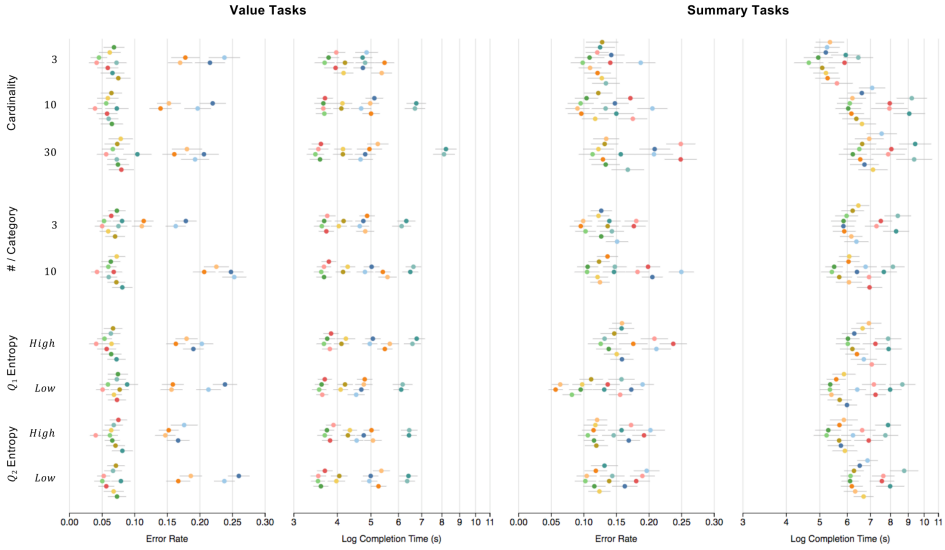


Figure 5.11: Error rates and completion time (log-transformed) for each encoding, along with 95% confidence intervals. Reproduced, with permission, from the paper of Kim and Heer (2018).

these studies, which has all materials publicly available (Kim and Heer, 2018). As a plus, this study focused on the effect of data scale and distribution on performance, so it aligns with my interest in scalability. Kim and Heer (2018) tested the effectiveness of twelve trivariate encodings, shown in Figure 5.12, where Q_1 and Q_2 are numerical, continuous variables, and N is a categorical variable..

The data consists of 2016 U.S. monthly weather measurements, which are published as part of the Global Historical Climatology Network-Daily Database (GHCN) (Menne et al., 2012), and contains the categorical variables State and Month, and the following numerical variables: Maximum Temperature, Minimum Temperature, Average Wind Speed, Wind Direction, Strongest Gust Speed, Precipitation, Snowfall, and Snow Depth.

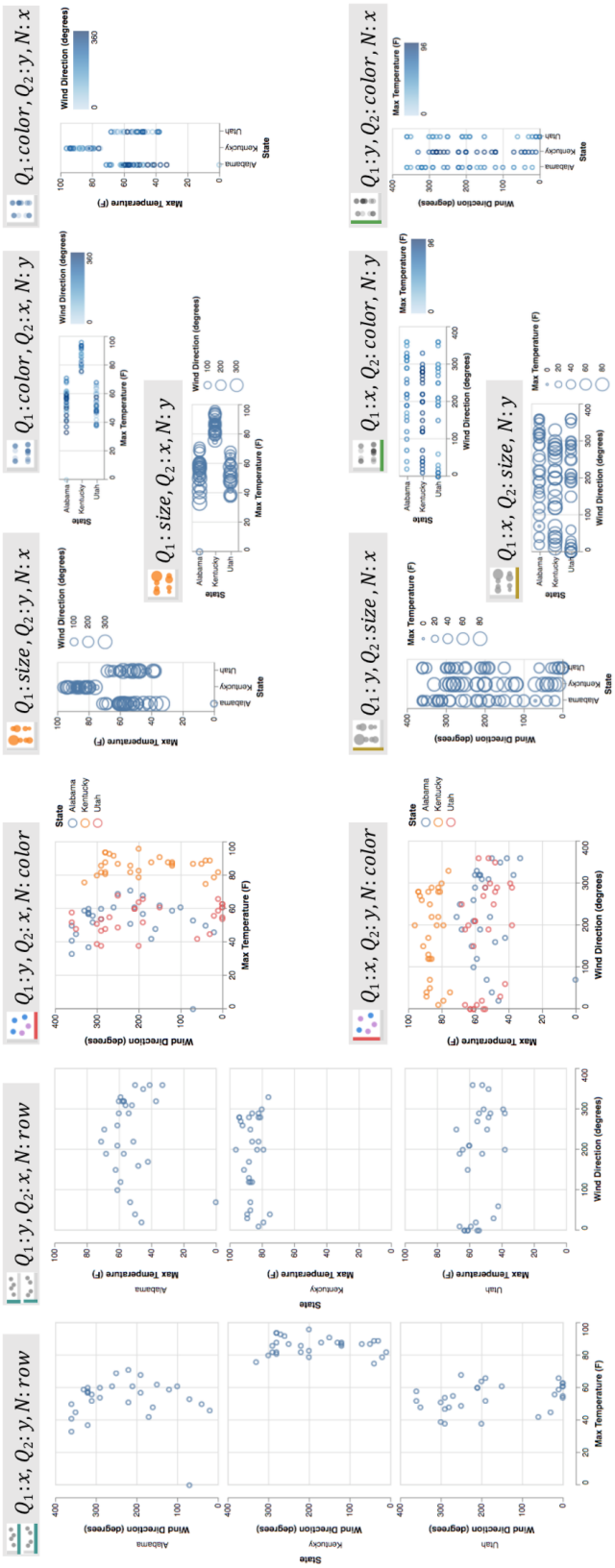


Figure 5.12: Encodings evaluated in Kim and Heer's experiment. Reproduced, with permission, from the paper of Kim and Heer (2018).

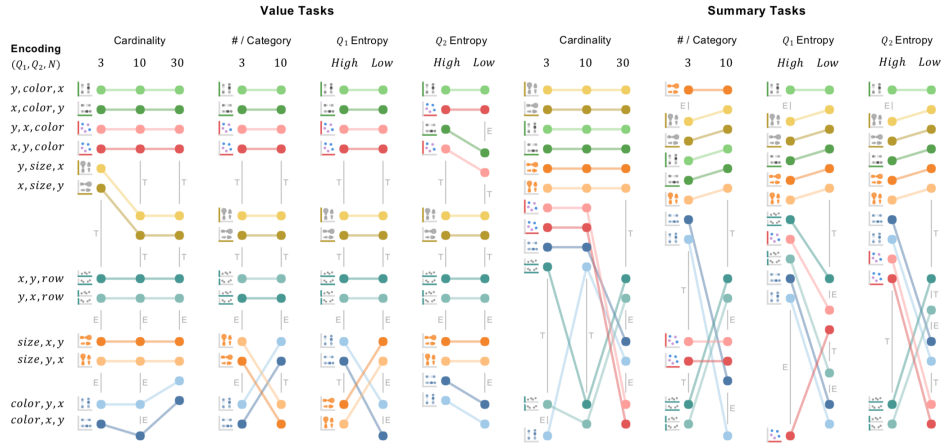


Figure 5.13: Rankings of effectiveness divided by task and factor level. Gaps labelled T and E represent statistically significant differences in completion time and error rate, respectively. Effectiveness is calculated as mean error rate. Reproduced, with permission, from the paper of Kim and Heer (2018).

The stimuli of that experiment were produced by sampling from GHCN and they were divided into 24 experimental conditions that result from the crossing of the following factors: Cardinality (3, 10, 20), where cardinality is the number of categories N, #/Category (3, 30), $Entropy_{Q_1}$ (Low, High), and $Entropy_{Q_2}$ (Low, High). The specific variables Q_1 and Q_2 were not factors; thus, they vary randomly across stimuli. N is always a derived variable resulting from the conflation of State and Month (as in TX-03), although in the stimuli it appears simply as State; that is, participants are not exposed to Month.

Study participants were asked to perform tasks that involved questions about Q_1. The tasks were of the following types: value tasks, further split into read value and compare value; and summary tasks, further split into find maximum and compare averages. Table 5.2 lists the question templates for each task. Error rates and completion times were measured, and rankings of encodings were created based on the error rates.

The results of this experiment reveal that the effect of encoding on error rates depends on the task and on the various factors manipulated in the experiment (Figure 5.11); therefore, a different ranking of encodings is created within each task group and factor level (Figure 5.13). Furthermore, the differences in error rate and completion time for the encodings are not always statistically significant; for instance, in summary tasks involving datasets with three and ten categories, the ten best ranked encodings did not score significantly different error rates.

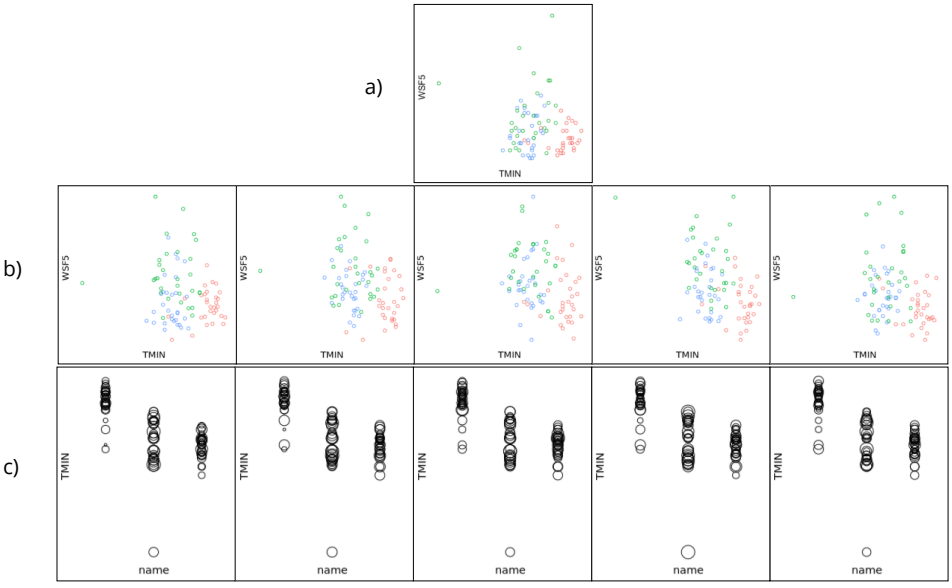


Figure 5.14: Images generated for the global discriminability test. a) Original plot used by Kim and Heer. b) Plots depicting variations of the original data, resulting from sampling from statistical models fitted to Kim and Heer's data. Only the question variable Q_1 (WSF5 in this example) is simulated. c) The same simulated data depicted using size encoding (*size_y_x*) for Q_1.

Measuring Discriminability

These rankings of effectiveness are useful to visualization practitioners but they do not enhance our understanding of what drives the effectiveness of a visualization. They are digestible guidelines, not elementary quality criteria; as such, they only vaguely help us predict what would work in a new visualization design. In their popular science book, Cham and Whiteson (2017) argue that while decades ago the periodic table of elements represented our best understanding of the building blocks of matter, it contained clues that suggested that the actual building blocks were smaller. Many elements shared commonalities and were grouped together, forming patterns that seemed coincidental. Today we know that the relationships between elements are driven by patterns in smaller particles: quarks and leptons. Yet, patterns in the electrical charge of these particles, which today seem entirely coincidental, suggest that something even more elementary is behind them. Likewise, the fact that a more elegant explanation exists for the visualization rankings is undisputed, but we currently only speculate the reasons why some encodings are better than others. My goal here is to examine if discriminability can be considered a good candidate for this explanation.

Table 5.3: Structure of the global discriminability experiment. Note how encodings within each experimental condition are tested on the same collection of 20 datasets. The datasets are different across conditions.

Cardinality	#/category	Entropy(Q1)	Entropy(Q2)	Q1	Q2	Encoding	Dataset ID
3	3	High	High	TMAX	SNOW	x_y_color	1...20
3	3	High	High	TMAX	SNOW	size_x_y	1...20
3	3	High	High	TMAX	SNOW	...	1...20
10	3	High	High	TMAX	SNOW	x_y_color	21....40
10	3	High	High	TMAX	SNOW	size_x_y	21...40
10	3	High	High	TMAX	SNOW	...	21...40
...	...	...	...	...	...	...	...

In the next sections I'll report two experiments. The first experiment is a *global* discriminability test, of the kind someone would run without a specific task in mind. It generates a variety of datasets then computes the average similarity across visualizations of these datasets for each encoding being considered. In essence, it measures the sensitivity of each encoding, or how much *overall* visual change we can expect of each encoding, in average. The link to effectiveness is in the assumption that the less sensitive an encoding, the harder it is to decode information: reading and comparing values is more difficult when the visual range is narrow.

The second experiment is task-specific. In Figure 5.13, we can see that the rankings for the summary tasks (mean comparison and find maximum) are somewhat different than the value rankings. In the mean comparison tasks, participants are instructed to select the state with the highest mean out of only two options. It is safe to assume that in these tasks what matters is how easily people can segregate the values of the two states in question and compare their values. In experiment 2, I devise a scheme to test *local* discriminability.

Experiment 1 - Global Discriminability

The experiment of Kim and Heer is structured as follows: 8 different datasets were sampled from the GHCN records for each combination of factors *cardinality* $\times$ *#/category* $\times$ $entropy_{Q_1}$ $\times$ $entropy_{Q_2}$ $\times$ *encoding*. That is, within each condition, each encoding was tested with a different collection of datasets, all with similar characteristics (dictated by the experimental condition). The datasets vary randomly in Q_1, Q_2, and the specific data

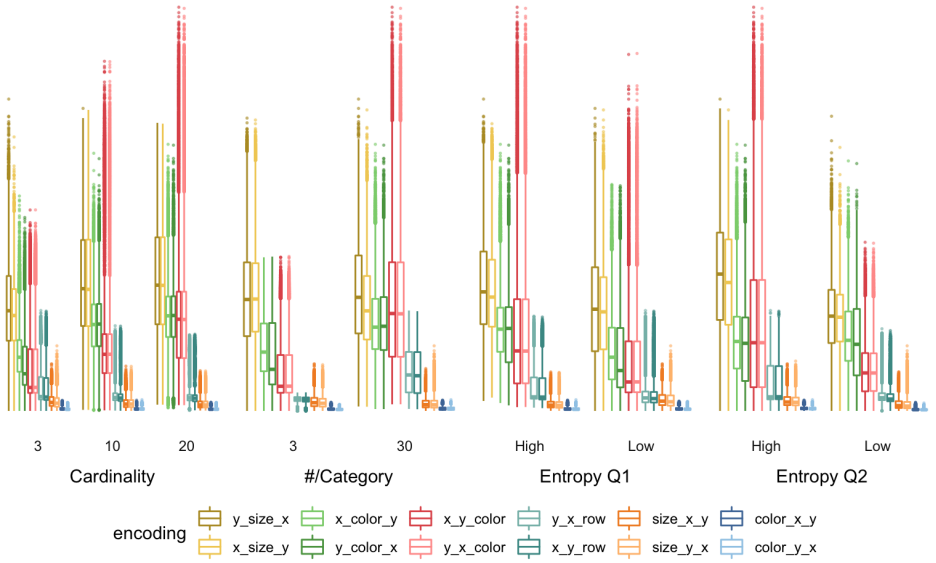
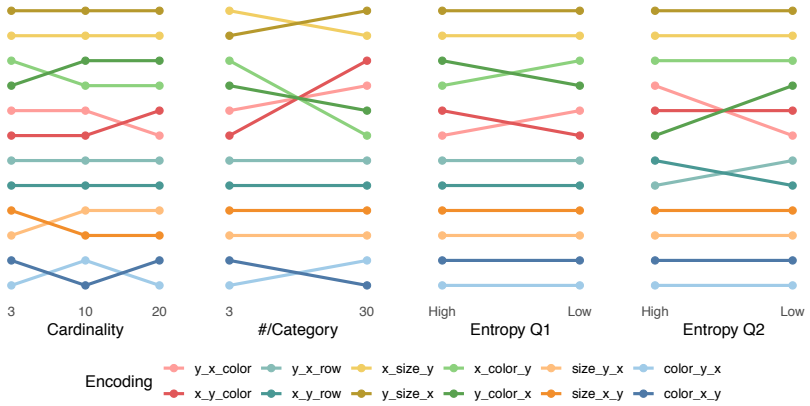


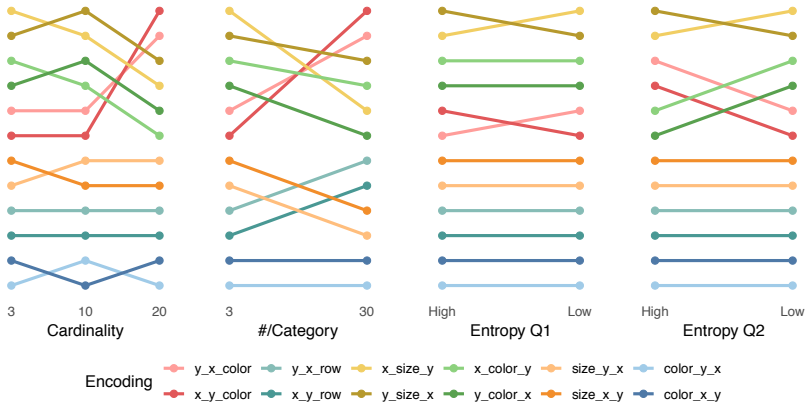
Figure 5.15: Global discriminability (Experiment 1). MS-SSIM weights prioritizing coarsest scale. $W = (0, 0, 0, 0, 1)$

points and states that the questions center on, in order to avoid a combinatorial explosion of conditions. In the discriminability tests, I prioritized symmetry by testing all encodings within a given experimental condition on the *same* datasets. Furthermore, Q_1 and Q_2 were not varied randomly; instead, they were a factor in the experiment (between-encodings). These changes were made because the scale of the test is not a problem here, so I can test every possible cross between $Q1$, $Q2$, and the rest of the factors. In summary, I created 20 datasets by simulation for every combination of factors $\textit{cardinality} \times \textit{\#/category} \times \textit{entropy}_{Q_1} \times \textit{entropy}_{Q_2} \times Q_1 \times Q_2$. Table 5.3 demonstrates this structure.

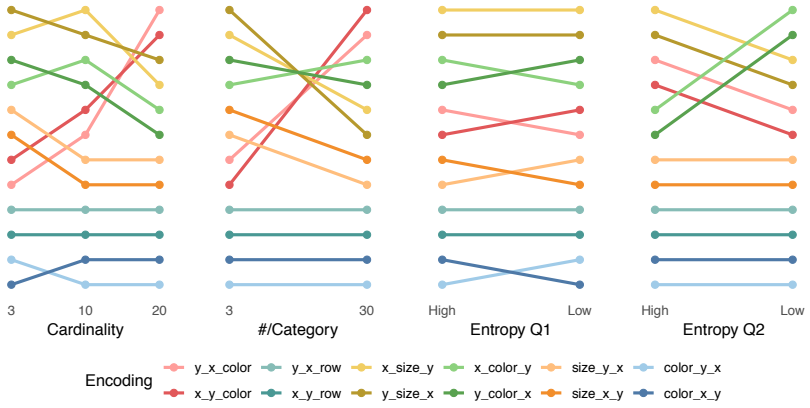
In order to simulate data that are similar to the data used by Kim and Heer (2018), I sampled values from generalized linear models (GLMs) fitted to the GHCN data. The simulation consisted in randomly drawing a dataset that matched the given experimental condition, then replacing its Q_1 values by values sampled from the model. The replacement step was repeated 20 times. The GLMs were fitted as follows. Given a condition, all records in Kim and Heer's data that match Q_1 were collected. Then a GLM was fitted to these records with Q_1 as the response variable and State as the covariate. Since all datasets have low correlation, Q_2 was omitted from the model; thus, the GLMs simply learn one distribution for each state. Figure 5.14 shows a reference dataset and simulated datasets visualized with two different encodings.



(a) High-level ranking. $W = (0, 0, 0, 0, 1)$



(b) Uniform ranking. $W = (1, 1, 1, 1, 1)$



(c) Low-level ranking. $W = (1, 0, 0, 0, 0)$

Figure 5.16: Discriminability rankings for visualization encodings resulting from Experiment 1, which measures global discriminability. Each ranking was produced with different MS-SSIM weights.

For each encoding, pairwise similarity judgments were computed with the MS-SSIM on the YUV representations of the images. Each tuple (*cardinality*, *#/category*, $entropy_{Q_1}$, $entropy_{Q_2}$, Q_1, Q_2, *encoding*) yields a discriminability score computed as the average pairwise similarity between the 20 images. These scores are then aggregated to produce scores per factor level, used in the rankings of encodings. Three rankings were computed, each with different scale weights: uniform ($W = [1, 1, 1, 1, 1]$), high-level ($W = [0, 0, 0, 0, 1]$), and low-level ($W = [1, 0, 0, 0, 0]$). Figure 5.16 shows the three rankings using the same color scheme as Kim and Heer's rankings.

Results

The high-level ranking matches almost entirely the Value Task ranking, with the only difference being the position of the pair of encodings *x_size_y/y_size_x*; this difference disappears when we consider that no statistically significant difference was found between the four best ranked encodings for the Value Tasks. The uniform and low-level rankings also match the Value Task rankings to a lesser extent. The summary task ranking, which is characterized by a drop in the effectiveness of the encodings *x_y_color* and *y_x_color*, and an increase in the effectiveness of the encodings *size_x_y* and *size_y_x* is not matched well by the discriminability rankings.

Furthermore, the boxplots in Figure 5.15 reveal a similar partition of encodings as the one found by Kim and Heer, with the first four pairs of encodings exhibiting distinctively higher discriminability compared to the two lowest encodings.

Experiment 2 - Local Discriminability

As mentioned earlier, we cannot expect a general test as the one presented in Experiment 1 to explain accurately the effectiveness of a task that requires the comparison of two sections of a visualization, because that experiment evaluated global discriminability. At first, it seems reasonable to simply extract the data of the two categories in question (the States in the weather data) and plot them independently, each in its own plot, then measure their similarity. This could produce good results for encodings where categories are spatially segregated, after all, extracting and comparing the categories is what people *need* to do in order to answer the mean comparison question. However, we should not expect this strategy to match well the effectiveness of encodings like *x_y_color* (the multi-category scatterplot), where data for different categories share the same axes. In these encodings, it can be difficult, in human perception terms, to separate categories if there are many of them and if the plot

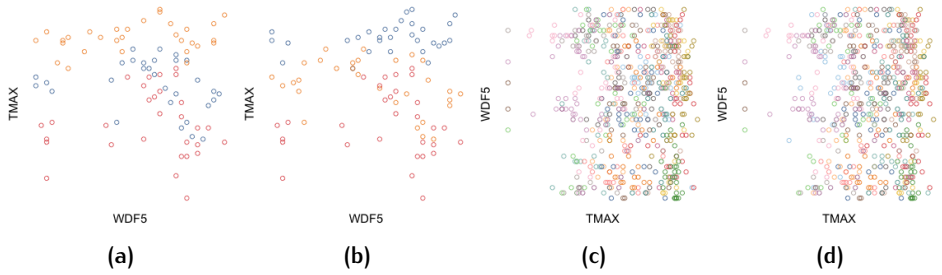


Figure 5.17: Pairs of colored scatterplots (*y_x_color*) with y values swapped between two categories. a) and b) have 3 categories in total, while c) and d) have 30 categories. These pairs are used to measure the visual discriminability of two categories (other categories fixed) along one variable.

is crowded. A test based on a procedure that isolates the categories in different layers would ignore this difficulty.

In order to test the discriminability of the visual representations of the two categories *within the context of the whole plot* I devised the following testing scheme. Given a plot, a subset of two categories, and the variable Q_1 subject to the mean comparison, a second plot is generated where the values of Q_1 are swapped between the two categories. The values for Q_2 in both categories remain fixed, as well as all data points in all other categories. The similarity is then computed on this pair of images, effectively measuring the visual similarity of the two groups of data points in the context of the rest of the data.

This test did not employ statistical simulation. I modified the same datasets that served as stimuli in Kim and Heer's experiment, which had 2,304 mean comparison tasks. In my experiment, each of these datasets was modified once, resulting in 4,608 datasets. Discriminability was calculated as the average similarity (MS-SSIM on YUV) between source and modified datasets. As in the section above, I present rankings that correspond to three very distinct MS-SSIM parameterizations.

Results

The local discriminability ranking resulting from Experiment 2 (Figure 5.19) correctly captures the main change observed in the Summary Tasks rankings: the encodings that map $Q1$ to size become highly effective, while the multiclass scatterplot becomes ineffective. While the rankings do not deviate drastically, this time the *low-level* ranking is the one that matches better the Summary task ranking of effectiveness. This is not surprising, since the summary tasks require local judgements. To be more precise, the summary judgements require visual aggregation, but that

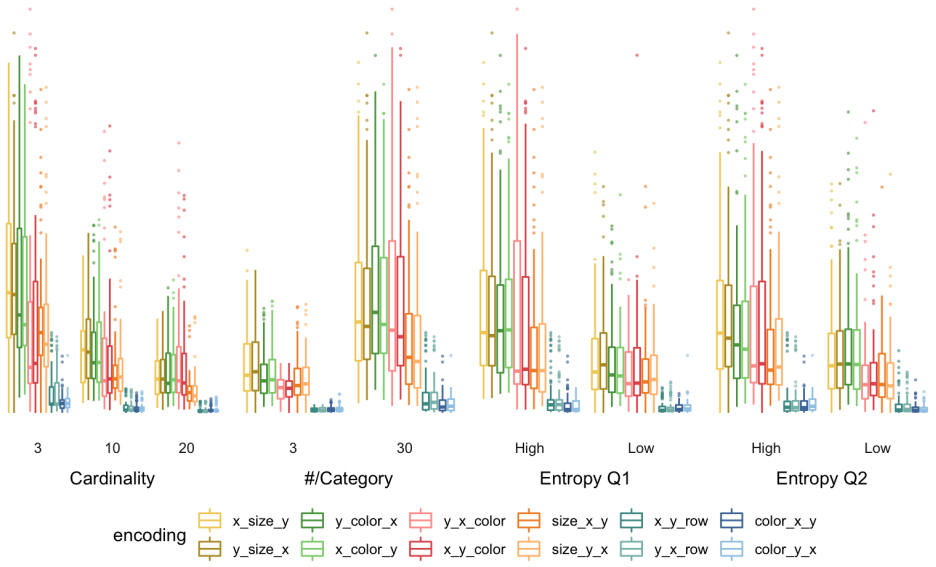


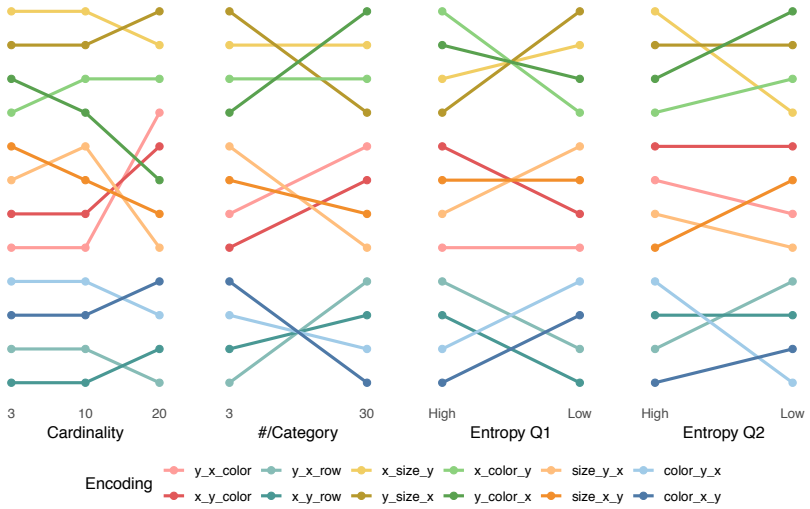
Figure 5.18: Local discriminability (Experiment 2). MS-SSIM weights prioritizing finest scale. $W = (1, 0, 0, 0, 0)$

cannot be considered a global judgement because the question covers only two categories. In order words, we expect the differences between two groups of points in a plot to disappear if viewed from afar when in the context of more categories, especially when the number of "distractor" categories increases. In fact, this is exactly what we observe in the high-level ranking (Figure 5.19c), where *size_y_x/size_x_y* and *x_y_color/x_y_color* switch back to their global discriminability ordering.

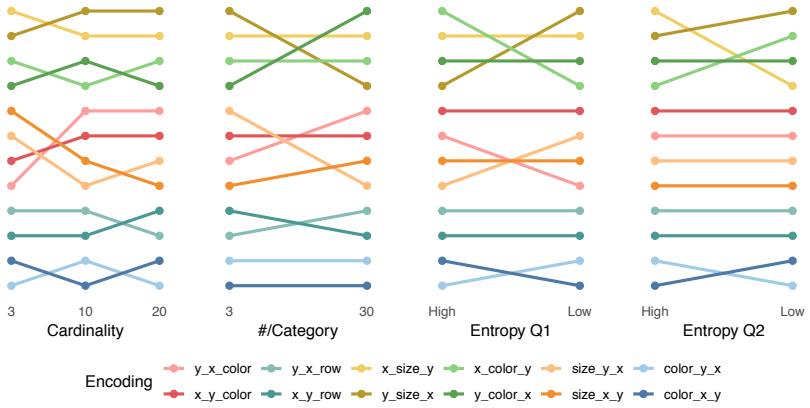
5.13 CONCLUSIONS

The correspondence between the rankings of discriminability and empirical effectiveness suggest that the effectiveness of the encodings is, to a large extent, driven by encoding discriminability. The results show that the discriminability tests based on MS-SSIM are useful as tools to assess the discriminability of visualizations. While discriminability has been a quality criterion in visualization for a long time, it has been mainly confined to theoretical discussions. This work constitutes the first methodical application of the discriminability criterion to the evaluation of visualization encodings.

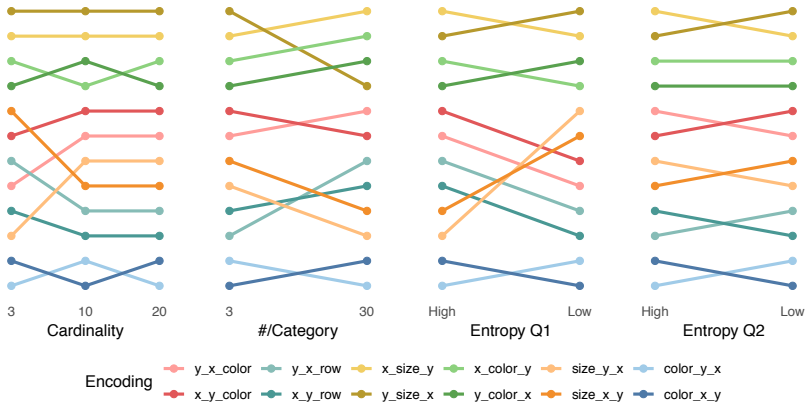
Fine grained changes in Kim and Heer's rankings due to entropy and scale were not matched by the discriminability rankings. This suggests that discriminability cannot fully explain the rankings. This is to be ex-



(a) Low-level ranking. $W = (1, 0, 0, 0, 0)$



(b) Uniform ranking. $W = (1, 1, 1, 1, 1)$



(c) High-level ranking. $W = (0, 0, 0, 0, 1)$

Figure 5.19: Discriminability rankings for visualization encodings resulting from Experiment 2, which measures local discriminability. Each ranking was produced with different MSSSIM weights.

pected, since other factors are known to influence people's judgements. Among these factors are saliency and distortions in the perception of brightness, contrast, length, and area (as described by Steven's law). But more importantly, discriminability seems to be the strongest factor behind effectiveness in these experiments, as it explains the majority of the patterns.

It is worth noting the limited scale of Kim and Heer's experiment. Although it can be considered a very large controlled experiment involving human participants, it is tiny compared to what is possible to accomplish using a computational measure like MS-SSIM. Namely, to make the study amenable, data size, correlation, and entropies were discretized into at most three levels. With discriminability tests, if necessary, it is possible to construct a discriminability *surface* over these dimensions.

Finally, the MS-SSIM score can be interpreted as an inverse measure of the strength of the visual difference generated by a visual encoding. The successive downsampling steps simulate the increase of viewing distance. Intuitively, differences that are preserved at a large distance are easy to read at normal viewing distance, and judgements that depend on evaluating these differences are expected to be more accurate with less difficulty.

5.14 SUMMARY

In this chapter, I examined the problem of automated evaluation of visual encodings. I started by reviewing the current evaluation practice, and argued that a commonly narrow scope in the definition of test data results in new visual encodings and techniques being undertested. I also highlighted the low scalability and high cost of evaluation approaches that rely on human judgements, and pointed to automated evaluation as a solution to improved the scale and coverage of evaluation. I proposed *discriminability tests* as tools to evaluate the quality of visualizations with a large collection of datasets with varying characteristics. Such tests consist in simulating an array of different datasets and scoring the discriminability of the corresponding visualizations.

In order to guarantee the scalability of discriminability tests, I proposed the use of an image similarity measure (SSIM) as a substitute for human judgements. The appropriateness of SSIM for rating plot similarity was evaluated in an experiment where SSIM scores for a set of 247 scatterplots were compared with scores derived from empirical data. The results revealed a notable overlap between the approaches, suggesting SSIM could be used to replace human judgements.

Finally, I conducted an experiment to answer whether there is a link between the discriminability and the effectiveness of visual encodings. I computed discriminability scores for several encodings and compared them with empirical effectiveness measures published in the visualization literature. My comparative analysis shows that there is a large overlap between the discriminability computed with SSIM and empirical effectivenes; in other words, the more discriminable encodings tend to offer better support to tasks such as reading values and comparing means.

6 FUTURE WORK

In this chapter I discuss how the research presented in the previous chapters creates opportunities for new advances in the visualization field. I pose questions that arise from technical challenges I encountered while developing this thesis and from the findings of my research. Occasionally, I suggest concrete paths for investigating such questions.

6.1 MODEL SELECTION

Model selection, the elegant and general statistical framework I used in Chapter 3 to find good hierarchical views, is well suited to information visualization. In that chapter, data plots were defined as statistical models of the data (whose parameters were encoded visually), and an information theoretic criterion was used to select the best model-plot. This way of treating visualizations is intuitive only when the data is very large. When the data is small, there is no penalty for seeing a visualization as a faithful "reflection" of the data. It becomes apparent that a visualization is a rough model when we can notice various artifacts (e.g., overlap, clutter) that emerge with large scale data.

Many visualization techniques that rely on feature extraction, such as splatterplots, are difficult to use because of the need to manually tune several parameters. However, under the model selection framework these techniques could become powerful, because information criteria could be used to automatically tune parameters. Compared to guideline-based constraints for visualization selection (Moritz et al., 2018), information criteria is more appropriate because it measures the fitness of a view to the data at hand. With guidelines, the recommendations are based on coarse data descriptors, such as "low entropy" and "high correlation".

Nevertheless, expressing empirical perception results (possibly task-specific) in information theoretic terms is difficult within the model selection framework, and is a topic that merits further research. For instance, how to integrate the knowledge acquired in Chapter 4, that motion outlier detection is difficult under certain circumstances, into an optimizer? To begin with, it is not clear which parameters should be optimized, but suppose there exists a set of parameters that are to be tuned to make animated scatterplots better *overall*. How can we integrate the specific model

we have for outlier detection into a criterion that accounts for other qualities (clutter, cluster detection effectiveness, mean comparison effectiveness, etc.)?

In the application of knowledge from perception into other fields lies another trap: the attempt to emulate the mechanisms of the brain to make perception predictions. In my work, the model complexity component of the information criterion (the other component was fitness) consisted of a calculation that boils down simply to the number of elements in the screen. As shown in my validation using the statistical saliency model, this was sufficient to control the level of clutter. The SSIM score that I used in Chapter 5 is another example of perceptual measure that *is not* based on the emulation of perceptual processes; instead, it is designed to mimic the hypothesized function of the human visual system while disregarding its *modus operandis*. It is shown to outperform measures that reproduce at every step the filters known by vision science (see Wang et al. (2003) for a review).

6.2 ELICITING SOFT KNOWLEDGE

Feature extraction techniques and visualization tuning approaches that require a formal notion of data relevance face the challenge of eliciting an appropriate representation of users' knowledge, expectations, and goals. In the visualization literature, this has been referred to as soft knowledge (Kijmongkolchai et al., 2017). In the technique I proposed for hierarchy summarization, an important assumption is made about users' expectations of the data: the value (size) of an aggregate category is expected to be proportional to the number of children nodes. In other words, small subtrees are expected to have smaller value than large subtrees. Moreover, the distribution of values within a subtree is expected to be uniform. Whenever categories fail to meet this expectation, the algorithm pushes the visualization to expose them. Therefore, the algorithm produces the desired effect of exposing data that contradicts expectations.

While this expectation is customizable in my technique, in practice we (the visualization community) do not know a good way to elicit it, and it is not practical to ask users to "upload" a statistical model that describes their expectations. Surprise Maps (Correll and Heer, 2017), a technique that computes and highlights surprise in datasets, also relies on models of expectation. The authors suggest a number of default models, including uniform and Gaussian, but acknowledge that selecting expectation models demands domain knowledge and statistical expertise.

Ideally, some kind of interface would ask users about their prior beliefs, or perhaps infer the beliefs of a group of users from their collective inter-

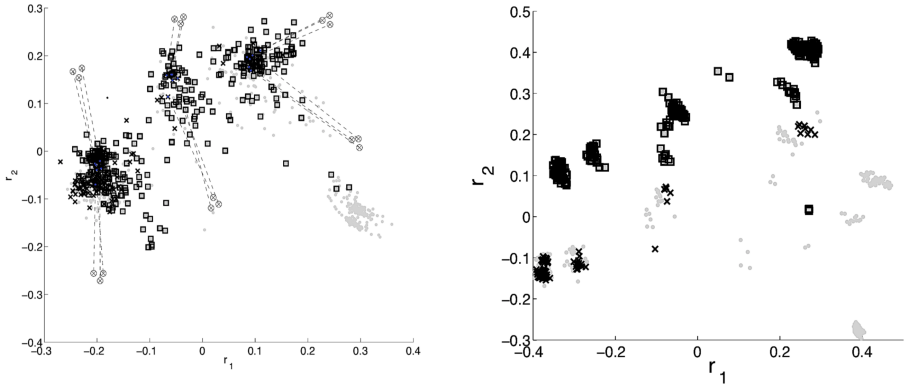


Figure 6.1: Illustration of semantic interaction. Left: Initial document embedding. Grey points and links represent users interactions moving points to new locations according to their own knowledge. Right: New view resulting from model update, with users soft knowledge integrated. ©2011 IEEE. Reprinted, with permission, from A. Endert, C. Han, D. Maiti, L. House, C. North. Observation-level interaction with statistical models for visual analytics, 2011.

actions. A simpler way would consist in inferring an expectation model from past or simulated data; but even then, this could not be construed as a trivial task. Endert proposed a method where users update the parameters of a model by directly manipulating elements in a visualization (Endert et al., 2011, 2012). Their technique is particularly suited to 2D embeddings of a high-dimensional data. In one example of what they call *semantic interaction*, an interface displays a spatial embedding of large documents and lets users correct individual document representations according to their own similarity judgement (Figure 6.1). The model then updates to reflect changes, effectively absorbing users' soft knowledge.

With the approach of Endert et al. users could easily provide feedback to the hierarchy summarization model by collapsing and expanding nodes; however, it is unlikely that updating individual parameters manually would suffice (the DMOZ hierarchy in Chapter 3 has more than half a million nodes/parameters). How can users state comprehensive hypotheses, such as "I expect the value of health care stocks to depend on their cash flow", or "I expect the occurrence of animal words to be highly skewed"?

6.3 ERROR PREDICTION

In Chapter 4, I used logistic regression models to analyze the data collected in the motion outlier experiment. Each model estimates the ex-

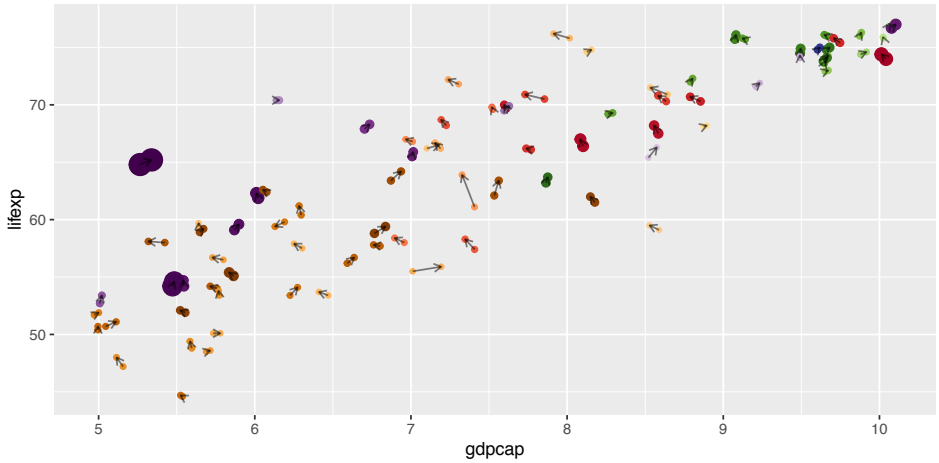


Figure 6.2: Illustration of a transition in the Gapminder dataset. Countries are represented by bubbles; arrows indicate changes in countries' values from one year (1981) to the next. In animated scatterplots the positions are smoothly interpolated over time, creating the perception of motion. The model learned in Chapter 4 predicts that it is difficult to identify the motion outlier in this transition, El Salvador (large arrow in the middle).

pected accuracy of an outlier detection task (speed or direction of motion) given the visual features of an outlier. When the models are used to make *predictions*, many design opportunities arise. The accuracy predictions can inform visual interventions that boost the saliency of outliers only when they are likely to be missed. In this section I will briefly discuss how such predictive models could be used to make outliers more evident in animated scatterplots of the popular Gapminder dataset.

Gapminder is a foundation that aggregates world development data published by multiple organizations, such as the United Nations. Each dataset distributed by Gapminder features a numerical indicator of development for every country and year, and can be joined to form richer multivariate datasets. This data has motivated several studies in information visualization. Part of the research is dedicated to techniques that enable better tracking of countries of interest (Hu et al., 2016; Kondo and Collins, 2014), while the other part is made of contributions that help users observe large structures (Collins et al., 2009b) and trends (Robertson et al., 2008).

Gapminder scatterplots usually encode two numerical indicators in the axes, continent as color, and population as size. They are animated across years, smoothly displaying changes in all variables over time. Since the axes are often correlated (as in life expectancy and GDP), a bubble cloud tends form. Figure 6.2 demonstrates how outlying changes in a data point

Table 6.1: Speed and direction of motion outliers with lowest predicted probability of detection (p) in a time-varying scatterplot of Gapminder.

	Speed			Direction		
Rank	Year	Country	p	Year	Country	p
1	1965	China	0.08	1998	Turkey	0.11
2	2008	Zambia	0.1	1993	Venezuela	0.11
3	1999	Liberia	0.1	2008	Paraguay	0.12
4	2000	Liberia	0.18	1999	Chad	0.14
5	2010	Zimbabwe	0.21	1991	Cent. Afr. Rep.	0.15
6	2007	Zambia	0.24	1987	Malawi	0.15
7	1984	Chad	0.26	1997	Zambia	0.15
8	2009	Botswana	0.31	1986	Nigeria	0.15
9	2003	Chad	0.38	1994	Sierra Leone	0.16
10	2001	Liberia	0.4	1989	Congo	0.17
11	1981	El Salvador	0.44	1984	Costa Rica	0.21
12	1987	Chad	0.5	1971	Niger	0.21
13	1976	Guatemala	0.5	1977	Zambia	0.21
14	1979	Nicaragua	0.5	1962	Norway	0.22
15	1963	Mauritania	0.53	1973	Rwanda	0.22

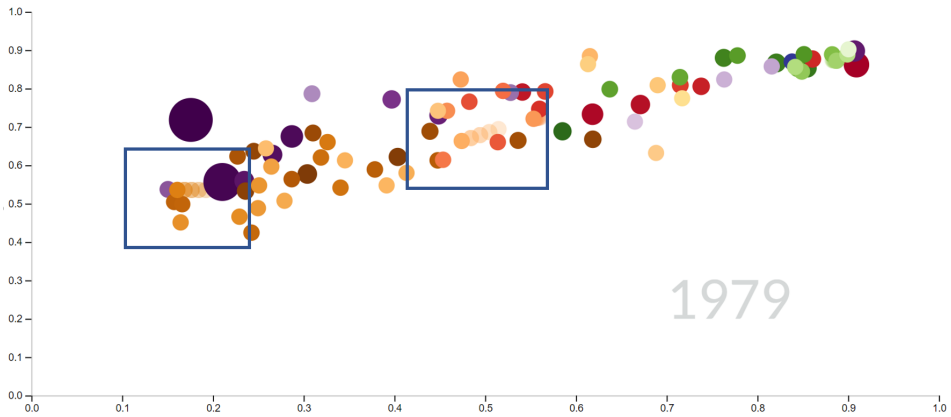


Figure 6.3: Based on predictions from an empirical model, motion traces are deployed when outliers are in saliency deficit. This visual intervention is intended to boost saliency, making outliers easier to detect.

can hide within a bubble cloud. In this example, El Salvador is by far the point that moves the most, but has rather average features otherwise, a fact that does not contribute to make it a global outlier. In this thesis, I offered evidence that suggests that only global outliers are likely to be properly identified.

In order to make a prediction with the logistic model, all we need is a measure of the saliency of the outlier in each considered visual channel; namely x, y, color, size, direction, and speed. Given these values, the model calculates a probability. We can then establish a threshold under which the probability is considered unsatisfactory, and a visual intervention is introduced to boost the saliency of the outlier. This saliency boosting strategy should increase the probability of an outlier being detected; moreover, it allows interventions to be deployed only when necessary. Using this strategy, I calculated the probability of correct outlier detection for every year transition in Gapminder between 1960 and 2011. The axes are life expectancy (y) and GDP/capita (x). From this calculation, I ranked the most saliency-deficited outliers (Table 6.1).

In the context of a large animated scene, the points in Table 6.1 can be interpreted as local outliers that are hard to detect (low probability of detection). The saliency of these motions can be improved in many ways, and ultimately, it is up to the designer to find a suitable visual accessory. To illustrate the potential of this approach, I have explored one option, the use of visual traces to boost low saliency motions. In Figure 6.3 visual traces are applied to outliers in saliency deficit during the animation.

Note that this approach is limited in that traces are not helpful when motion is an outlier because of low speed (little movement). Similarly, direction outliers which do not move very far between scenes will also not be enhanced much by the trace. In these cases, another type of highlight may be more appropriate, such as radiating rings.

This prototype illustrates a kind of dynamic interface that adjusts depending on predictions from an empirical model. This direction of research is interesting because it can improve the effectiveness of visualizations that are known to not support well certain tasks, but are very familiar to certain user groups. The animated scatterplot is serves as an example.

6.4 UNDERSTANDING SIMILARITY

In Chapter 5, I demonstrated how clusterings of scatterplots based on the Structural Similarity Index (SSIM) share information with human-made clusterings. In my preliminary tests of the SSIM, I discussed how the similarity of certain visualizations (e.g., graphs) tends to be judged at a

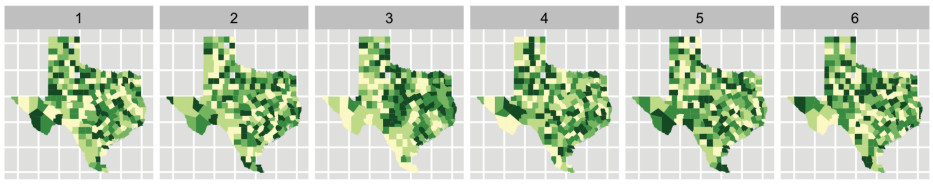


Figure 6.4: Plots lined up for visual inference. Five of these plots display simulated data under the null hypothesis. Only one of them displays real data (number 7*5 - 4*8). The visual inference tasks consists in finding the real data plot. The easiness of this task is indicative of confidence in rejecting the null-hypothesis. The visual inference framework relies heavily on similarity judgement. ©2010 IEEE. Reprinted, with permission, from H. Wickham, D. Cook, H. Hofmann, and A. Buja. Graphical inference for infovis, 2010.

different level of detail than other visualizations (e.g., scatterplots), even though they are based on the same visual marks. The multi-level SSIM scores that I computed support this hypothesis: the correct similarity ranking of graphs required heavier weights on coarser features. This brief analysis, however, was based on my own judgement of similarity. A better indication that SSIM can be used to study the granularity of plot similarity judgements is the fact that, in my experiments, the best match between SSIM and human clusterings of scatterplots was achieved with a coarser weighting, which matches the granularity of subjects' judgements. How can we know this ground truth granularity? Subjects were primed to make coarse assessments because the scatterplots were shown as thumbnails; moreover, the verbal descriptors elicited by Pandey et al. (2016) corroborate the high level of abstraction.

The hypothesis that human similarity judgements vary with visual encoding begs more investigation. A pair of visualizations could have very distinct values when examined closely, but be very similar when judged with "distant eyes". This is not necessarily a problem, but it can be one when it is assumed that only a single similarity judgement is possible. The visual inference framework seems particularly vulnerable to this problem. Visual inference is a method intended to be an alternative or at least a complement to null-hypothesis testing. It proposes the use of visual representations to gauge the plausibility of the null hypothesis as follows: a) plot the observed data; b) sample multiple datasets from a null model; c) plot the null datasets; d) rate the visual similarity between null and observed data 6.4. When observed data can easily be distinguished from null data, the null hypothesis can be rejected.

The success of visual inference depends, thus, on the appropriateness of the visual encoding. Among other things, it should not allow for ambiguous similarity judgements. The choice of visual encoding follows the

same basis upon which a statistical test is chosen: statistical power, or the probability of rejecting the null hypothesis when it is false. Hofmann et al. (2012) conducted a user study that compared the power of four statistical charts for depicting univariate distributions (boxplot, histogram, density, and dotplot). They concluded that dotplots were the best; however, the experiment did not shed light on the reason behind the differences in power. Can we understand the reason better by analysing which features are taken for similarity judgement in each chart? Could the reason be that some charts tend to be less discriminable at some level? Do people actually judge the features we assume they judge?

In summary, a deep understanding of how plot similarity is rated will inform appropriate visual methods for comparison of data, and free us from the need to run experiments to test specific visualizations.

6.5 EXPOSING AMBIGUITY

My research on measuring plot similarity with SSIM (Chapter 5) opens possibilities for the study of ambiguity in visualizations. Graph visualizations, for example, often employ edge bundling to reduce clutter. In doing so, ambiguity is traded for legibility, as changes in node connections can hide within bundles. The extent to which a visualization is ambiguous is unknown to the user. In order to make this information available, a tool would need to display all different datasets that yield a certain image, or some strategy to that effect. Given a visualization, the user asks how many distinct datasets could have generated this visualization?

The difficulty lies in how to discover these datasets. We have recently seen many generative neural networks that learned how to generate data, compose text, images, and videos. So it seems possible that, given a reference dataset, a neural network can be taught how to generate derived data that yields the same visualization. These models could be trained for each visualization type. However, a plot does not need to be identical pixel-by-pixel to be judged identical, so classic non-perceptual loss functions, such as l_1 and l_2 norms, are probably not up to the task. It is not practical either to collect human ratings.

Encouragingly, Zhao et al. (2017) recently demonstrated that SSIM and multi-scale SSIM are differentiable and suited to serve as loss functions for neural network training. Their results show that their SSIM-based loss function outperforms l_1 and l_2 norms in various image reconstruction problems. Furthermore, these loss functions were made readily available as plugins for the open source neural network framework Caffee.

Therefore, the foundations are laid to an exciting direction in visualization research. The value of this application lies in offering to the analyst

tools for inspection of the visual methods used, and a better understanding of their reliability, in the same vein as meta-analyses exist for statistical methods.

6.6 SUMMARY

In summary, this chapter discussed the following future research directions:

MODEL SELECTION Information-theoretical model selection is a promising framework for automatic parameter tuning of visualizations.

SOFT KNOWLEDGE ELICITATION Interfaces for eliciting soft knowledge are necessary to enable feature extraction and more informative overviews of large data.

ERROR PREDICTION Predictive empirical models can support dynamic interfaces that deploy visual accessories to avoid errors.

UNDERSTANDING SIMILARITY Multi-scale similarity measures can be employed to study how users read charts and make comparison judgements.

AMBIGUITY The ambiguity of visual encodings and its relation to scalability merits research. In particular, tools that can inform the level of ambiguity of a chart could help users adjust the confidence of decisions based on visual analysis.

7 CONCLUSION

This thesis presented three case studies that address quality and scalability problems of visual encodings in information visualization. These studies cover a wide range in the spectrum from application to foundational visualization research, which can be seen in their outcomes. I contributed an algorithm for tuning a specific visualization type, a user study that answers a question about a broad class of visualizations, and a method for evaluating a quality criterion that applies to all visualizations.

We see everywhere signs that information visualization is and will remain extremely important in the *communication* of data-driven insights. But it faces the challenge of remaining relevant in the exploratory phase of data analysis. The sheer scale and complexity of data that machine learning engineers deal with demands solutions that are designed and tested to be robust on the limit.

The cost of finding a good visual encoding and parameterizing it is high, and it discourages analysts to use visualization as a method to discover patterns in the data. In this thesis I proposed an automated approach for finding good views of the data that reduces this parameterization cost. The core of this approach consists in treating a data view as a *message* and scoring its information theoretic properties. As a method to find the a good view of a dataset, I consider this approach to be more promising than approaches that rank visual encodings based on coarse characteristics of the dataset.

However, I acknowledge the technical difficulty in designing these information criteria, especially with respect to modelling perceptual scalability. We are far from understanding well all perceptual phenomena that affect our ability to make sense of visually encoded data. This motivated me to break new ground in understanding how saliency influences our perception of data properties in multivariate visualizations. In my discussion of future work, I pointed to a use of the model resulting from this research not to find an optimal view, but to dynamically fix weaknesses according to effectiveness predictions.

Finally, I addressed the problem of evaluation. In order to invent scalable, robust visualizations, we need better evaluation methods. And to design better methods, we need a better understanding of the roots of effectiveness in visualization, and a way to verify them at a low cost. This points to automation of at least part of visualization evaluation. Empiri-

cal research on fundamental visualization questions enables the creation of models that synthesize knowledge of how humans interact with visualization. This knowledge should be put to use in a way that reduces our reliance on costly user studies. Here I proposed a general method for scoring discriminability, a basic quality property of visualizations that impacts their effectiveness. In the future, methods should be created to verify other properties. A stack of quality measures will help designers iterate faster and deliver custom solutions that are based on strong evidence.

7.1 SUMMARY OF CONTRIBUTIONS

The contributions from the three case studies discussed in this thesis are:

MDL TREECUTS A technique for summarization of hierarchies for visualization purposes. The treecuts are the result of a pruning strategy that balances information loss and clutter, and takes into account the specifics of the underlying data and the available display space.

SALIENCY DEFICIT An empirical study of the effect of saliency (and the lack thereof) on the effectiveness of animated scatterplots. The results indicate that accurate motion outlier detection in multivariate animated scatterplots depends on task-irrelevant features.

DISCRIMINABILITY TESTS A method based on a perception-motivated image similarity measure for rating the discriminability of a visual encoding given a collection of datasets. Results of the test on classic visual encodings are shown to correlate with empirical effectiveness rankings.

A SUPPLEMENTAL TABLES

Table A.1: Clustering quality measures used to compare label assignments of Pandey et al.'s data

Measure	Formula	Random ≈ 0	Range	Symmetric	Chance adjusted
Adjusted Mutual Information	$\frac{MI(U,V)}{\sqrt{H(U)H(V)}}$	✓	(0,1)	✓	✓
Normalized Mutual Information	$\frac{MI-E[MI]}{\max(H(U),H(V)-E[MI])}$	✓	(0,1)	✓	✗
Rand Index (RI)	$\frac{a+b}{V_2^n}$	✗	(0,1)	✓	✗
Adjusted Rand Index	$\frac{RI-E[RI]}{\max(RI)-E[RI]}$	✓	(-1,1)	✓	✓

U: clustering
V: ground truth label assignment
$H(X)$: entropy
$MI(X)$: mutual information
$E[X]$: expected value
a: the number of pairs of elements that are in the same set in U and in the same set in V
b: the number of pairs of elements that are in different sets in U and in different sets in V
V_2^n: the total number of possible pairs in the dataset (without ordering)

B CODE

Listing B.1: Numerical gradient descent algorithm

```
function(loss_funct, n_iter=20,
    init=c(.1, .1, .1, .3, .4),
    h=0.01, stepsize=0.01){

  f = loss_funct
  x = init

  while (n_iter > 0){
    fx = f(x)  # eval function w/ current weights

    grad = rep(0, length(x))  # store the gradient

    for (i in 1:length(x)) {
      # evaluate function at x+h
      xh = x
      xh[i] = x[i] + h
      fxh = f(xh)

      # compute the partial derivative
      grad[i] = (fxh - fx) / h
    }

    # follow the gradient
    x = params - stepsize * grad

    n_iter = n_iter - 1
  }

  return(x)
}
```

C | LICENSES

JOHN WILEY AND SONS LICENSE TERMS AND CONDITIONS

Feb 26, 2019

This Agreement between Dr. Rafael Veras Guimaraes ("You") and John Wiley and Sons ("John Wiley and Sons") consists of your license details and the terms and conditions provided by John Wiley and Sons and Copyright Clearance Center.

License Number	4536670634300
License date	Feb 26, 2019
Licensed Content Publisher	John Wiley and Sons
Licensed Content Publication	Computer Graphics Forum
Licensed Content Title	Assessing Effects of Task and Data Distribution on the Effectiveness of Visual Encodings
Licensed Content Author	Younghoon Kim, Jeffrey Heer
Licensed Content Date	Jul 10, 2018
Licensed Content Volume	37
Licensed Content Issue	3
Licensed Content Pages	11
Type of use	Dissertation/Thesis
Requestor type	University/Academic
Format	Print and electronic
Portion	Figure/table
Number of figures/tables	3
Original Wiley figure/table number(s)	Figure 7 Figure 8 Figure 3
Will you be translating?	No
Title of your thesis / dissertation	Visual Encoding Quality and Scalability for Information Visualization
Expected completion date	Feb 2019
Expected size (number of pages)	150
Requestor Location	Dr. Rafael Veras Guimaraes 33 Singer Crt Apt 1701 North York, ON M2K 0B4 Canada Attn: Dr. Rafael Veras Guimaraes
Publisher Tax ID	EU826007151
Total	0.00 USD
Terms and Conditions	

TERMS AND CONDITIONS

This copyrighted material is owned by or exclusively licensed to John Wiley & Sons, Inc. or one of its group companies (each a"Wiley Company") or handled on behalf of a society with which a Wiley Company has exclusive publishing rights in relation to a particular work (collectively "WILEY"). By clicking "accept" in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the billing and payment terms and conditions established by the Copyright Clearance Center Inc., ("CCC's Billing and Payment terms and conditions"), at the time that you opened your RightsLink account (these are available at any time at http://myaccount.copyright.com).

Terms and Conditions

- The materials you have requested permission to reproduce or reuse (the "Wiley Materials") are protected by copyright.

- You are hereby granted a personal, non-exclusive, non-sub licensable (on a stand-alone basis), non-transferable, worldwide, limited license to reproduce the Wiley Materials for the purpose specified in the licensing process. This license, **and any CONTENT (PDF or image file) purchased as part of your order,** is for a one-time use only and limited to any maximum distribution number specified in the license. The first instance of republication or reuse granted by this license must be completed within two years of the date of the grant of this license (although copies prepared before the end date may be distributed thereafter). The Wiley Materials shall not be used in any other manner or for any other purpose, beyond what is granted in the license. Permission is granted subject to an appropriate acknowledgement given to the author, title of the material/book/journal and the publisher. You shall also duplicate the copyright notice that appears in the Wiley publication in your use of the Wiley Material. Permission is also granted on the understanding that nowhere in the text is a previously published source acknowledged for all or part of this Wiley Material. Any third party content is expressly excluded from this permission.

- With respect to the Wiley Materials, all rights are reserved. Except as expressly granted by the terms of the license, no part of the Wiley Materials may be copied, modified, adapted (except for minor reformatting required by the new Publication), translated, reproduced, transferred or distributed, in any form or by any means, and no derivative works may be made based on the Wiley Materials without the prior permission of the respective copyright owner.**For STM Signatory Publishers clearing permission under the terms of the STM Permissions Guidelines only, the terms of the license are extended to include subsequent editions and for editions in other languages, provided such editions are for the work as a whole in situ and does not involve the separate exploitation of the permitted figures or extracts,** You may not alter, remove or suppress in any manner any copyright, trademark or other notices displayed by the Wiley Materials. You may not license, rent, sell, loan, lease, pledge, offer as security, transfer or assign the Wiley Materials on a stand-alone basis, or any of the rights granted to you hereunder to any other person.

- The Wiley Materials and all of the intellectual property rights therein shall at all times remain the exclusive property of John Wiley & Sons Inc, the Wiley Companies, or their respective licensors, and your interest therein is only that of having possession of and the right to reproduce the Wiley Materials pursuant to Section 2 herein during the

continuance of this Agreement. You agree that you own no right, title or interest in or to the Wiley Materials or any of the intellectual property rights therein. You shall have no rights hereunder other than the license as provided for above in Section 2. No right, license or interest to any trademark, trade name, service mark or other branding ("Marks") of WILEY or its licensors is granted hereunder, and you agree that you shall not assert any such right, license or interest with respect thereto

- NEITHER WILEY NOR ITS LICENSORS MAKES ANY WARRANTY OR REPRESENTATION OF ANY KIND TO YOU OR ANY THIRD PARTY, EXPRESS, IMPLIED OR STATUTORY, WITH RESPECT TO THE MATERIALS OR THE ACCURACY OF ANY INFORMATION CONTAINED IN THE MATERIALS, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTY OF MERCHANTABILITY, ACCURACY, SATISFACTORY QUALITY, FITNESS FOR A PARTICULAR PURPOSE, USABILITY, INTEGRATION OR NON-INFRINGEMENT AND ALL SUCH WARRANTIES ARE HEREBY EXCLUDED BY WILEY AND ITS LICENSORS AND WAIVED BY YOU.

- WILEY shall have the right to terminate this Agreement immediately upon breach of this Agreement by you.

- You shall indemnify, defend and hold harmless WILEY, its Licensors and their respective directors, officers, agents and employees, from and against any actual or threatened claims, demands, causes of action or proceedings arising from any breach of this Agreement by you.

- IN NO EVENT SHALL WILEY OR ITS LICENSORS BE LIABLE TO YOU OR ANY OTHER PARTY OR ANY OTHER PERSON OR ENTITY FOR ANY SPECIAL, CONSEQUENTIAL, INCIDENTAL, INDIRECT, EXEMPLARY OR PUNITIVE DAMAGES, HOWEVER CAUSED, ARISING OUT OF OR IN CONNECTION WITH THE DOWNLOADING, PROVISIONING, VIEWING OR USE OF THE MATERIALS REGARDLESS OF THE FORM OF ACTION, WHETHER FOR BREACH OF CONTRACT, BREACH OF WARRANTY, TORT, NEGLIGENCE, INFRINGEMENT OR OTHERWISE (INCLUDING, WITHOUT LIMITATION, DAMAGES BASED ON LOSS OF PROFITS, DATA, FILES, USE, BUSINESS OPPORTUNITY OR CLAIMS OF THIRD PARTIES), AND WHETHER OR NOT THE PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION SHALL APPLY NOTWITHSTANDING ANY FAILURE OF ESSENTIAL PURPOSE OF ANY LIMITED REMEDY PROVIDED HEREIN.

- Should any provision of this Agreement be held by a court of competent jurisdiction to be illegal, invalid, or unenforceable, that provision shall be deemed amended to achieve as nearly as possible the same economic effect as the original provision, and the legality, validity and enforceability of the remaining provisions of this Agreement shall not be affected or impaired thereby.

- The failure of either party to enforce any term or condition of this Agreement shall not constitute a waiver of either party's right to enforce each and every term and condition of this Agreement. No breach under this agreement shall be deemed waived or

excused by either party unless such waiver or consent is in writing signed by the party granting such waiver or consent. The waiver by or consent of a party to a breach of any provision of this Agreement shall not operate or be construed as a waiver of or consent to any other or subsequent breach by such other party.

- This Agreement may not be assigned (including by operation of law or otherwise) by you without WILEY's prior written consent.
- Any fee required for this permission shall be non-refundable after thirty (30) days from receipt by the CCC.
- These terms and conditions together with CCC's Billing and Payment terms and conditions (which are incorporated herein) form the entire agreement between you and WILEY concerning this licensing transaction and (in the absence of fraud) supersedes all prior agreements and representations of the parties, oral or written. This Agreement may not be amended except in writing signed by both parties. This Agreement shall be binding upon and inure to the benefit of the parties' successors, legal representatives, and authorized assigns.
- In the event of any conflict between your obligations established by these terms and conditions and those established by CCC's Billing and Payment terms and conditions, these terms and conditions shall prevail.
- WILEY expressly reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.
- This Agreement will be void if the Type of Use, Format, Circulation, or Requestor Type was misrepresented during the licensing process.
- This Agreement shall be governed by and construed in accordance with the laws of the State of New York, USA, without regards to such state's conflict of law rules. Any legal action, suit or proceeding arising out of or relating to these Terms and Conditions or the breach thereof shall be instituted in a court of competent jurisdiction in New York County in the State of New York in the United States of America and each party hereby consents and submits to the personal jurisdiction of such court, waives any objection to venue in such court and consents to service of process by registered or certified mail, return receipt requested, at the last known address of such party.

WILEY OPEN ACCESS TERMS AND CONDITIONS

Wiley Publishes Open Access Articles in fully Open Access Journals and in Subscription journals offering Online Open. Although most of the fully Open Access journals publish open access articles under the terms of the Creative Commons Attribution (CC BY) License only, the subscription journals and a few of the Open Access Journals offer a choice of Creative Commons Licenses. The license type is clearly identified on the article.

The Creative Commons Attribution License

The Creative Commons Attribution License (CC-BY) allows users to copy, distribute and transmit an article, adapt the article and make commercial use of the article. The CC-BY license permits commercial and non-

Creative Commons Attribution Non-Commercial License
The Creative Commons Attribution Non-Commercial (CC-BY-NC)License permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.(see below)

Creative Commons Attribution-Non-Commercial-NoDerivs License
The Creative Commons Attribution Non-Commercial-NoDerivs License (CC-BY-NC-ND) permits use, distribution and reproduction in any medium, provided the original work is properly cited, is not used for commercial purposes and no modifications or adaptations are made. (see below)
Use by commercial "for-profit" organizations
Use of Wiley Open Access articles for commercial, promotional, or marketing purposes requires further explicit permission from Wiley and will be subject to a fee.
Further details can be found on Wiley Online Library
http://olabout.wiley.com/WileyCDA/Section/id-410895.html

Other Terms and Conditions:

v1.10 Last updated September 2015

Questions? customercare@copyright.com or +1-855-239-3415 (toll free in the US) or +1-978-646-2777.

Title: Graphical inference for infovis
Author: Hadley Wickham
Publication: Visualization and Computer Graphics, IEEE Transactions on
Publisher: IEEE
Date: Nov.-Dec. 2010

Copyright © 2010, IEEE

LOGIN

If you're a copyright.com user, you can login to RightsLink using your copyright.com credentials.

Already **a RightsLink user** or want to learn more?

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW

Copyright © 2019 Copyright Clearance Center, Inc. All Rights Reserved. Privacy statement. Terms and Conditions.
Comments? We would like to hear from you. E-mail us at customercare@copyright.com

BIBLIOGRAPHY

Abello, James, Frank Van Ham, and Neeraj Krishnan

2006 "ASK-GraphView: A large scale graph visualization system", *IEEE Transactions on Visualization and Computer Graphics*, 12, 5, pp. 669-676. (Cited on p. 23.)

Albuquerque, Georgia, Martin Eisemann, Dirk J Lehmann, Holger Theisel, and Marcus Magnor

2010 "Improving the visual analysis of high-dimensional datasets using quality measures", in *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, IEEE, pp. 19-26. (Cited on p. 16.)

Alsallakh, Bilal, Wolfgang Aigner, Silvia Miksch, and M Eduard Gröller

2012 "Reinventing the contingency wheel: Scalable visual analytics of large categorical data", *IEEE Transactions on Visualization and Computer Graphics*, 18, 12, pp. 2849-2858. (Cited on p. 16.)

Andrienko, Natalia and Gennady Andrienko

2010 "Spatial generalisation and aggregation of massive movement data", *IEEE Transactions on Visualization and Computer Graphics*, 17, 2, pp. 205-19. (Cited on p. 22.)

Anscombe, Francis

1973 "Graphs in Statistical Analysis", *The American Statistician*, 27, 1, pp. 17-21. (Cited on p. 8.)

Archambault, Daniel and Derek Greene

2011 "ThemeCrowds: Multiresolution summaries of twitter usage", *International Workshop on Search and Mining User-generated Contents*, pp. 77-84. (Cited on p. 23.)

Archambault, Daniel, Tamara Munzner, and David Auber

2008 "GrouseFlocks: Steerable exploration of graph hierarchy space", *IEEE Transactions on Visualization and Computer Graphics*, 14, 4, pp. 900-913. (Cited on pp. 23, 46.)

Baldassi, Stefano, Nicola Megna, and David C. Burr

2006 "Visual Clutter Causes High-Magnitude Errors", *PLoS Biology*, 4, 3, e56. (Cited on p. 21.)

Bartram, Lyn, Colin Ware, and Tom Calvert

2003 "Moticons: detection, distraction and task", *Int. Journal of Human-Computer Studies*, 58, 5, Notification User Interfaces, pp. 515-545. (Cited on p. 55.)

Batch, Andrea and Niklas Elmqvist

2018 "The Interactive Visualization Gap in Initial Exploratory Data Analysis", *IEEE transactions on visualization and computer graphics*, 24, 1, pp. 278-287. (Cited on p. 2.)

Bates, Douglas, Martin Mächler, Ben Bolker, and Steve Walker

2015 "Fitting Linear Mixed-Effects Models Using lme4", *Journal of Statistical Software*, 67, 1, pp. 1-48. (Cited on p. 65.)

Behrisch, Michael, Michael Blumenschein, Nam Wook Kim, Lin Shao, Mennatallah El-Assady, Johannes Fuchs, Daniel Seebacher, Alexandra Diehl, Ulrik Brandes, Hanspeter Pfister, et al.

2018 "Quality metrics for information visualization", in *Computer Graphics Forum*, 3, Wiley Online Library, vol. 37, pp. 625-662. (Cited on p. 16.)

Bertini, Enrico and Giuseppe Santucci

2004 "Quality metrics for 2d scatterplot graphics: automatically reducing visual clutter", in *International Symposium on Smart Graphics*, Springer, pp. 77-89. (Cited on p. 18.)

Brew, Anthony, Derek Greene, Daniel Archambault, and Pádraig Cunningham

2011 "Deriving insights from national happiness indices", *Proceedings - IEEE International Conference on Data Mining, ICDM*, pp. 53-60. (Cited on p. 23.)

Bruckner, Stefan and Torsten Möller

2010 "Result-driven exploration of simulation parameter spaces for visual effects design", *IEEE Transactions on Visualization and Computer Graphics*, 6, pp. 1468-1476. (Cited on p. 13.)

Bylinskii, Zoya, Tilke Judd, Ali Borji, Laurent Itti, Frédo Durand, Aude Oliva, and Antonio Torralba

n.d. *MIT Saliency Benchmark*, http://saliency.mit.edu/. (Cited on p. 58.)

Bylinskii, Zoya, Nam Wook Kim, Peter O'Donovan, Sami Alsheikh, Spandan Madan, Hanspeter Pfister, Fredo Durand, Bryan Russell, and Aaron Hertzmann

2017 "Learning visual importance for graphic designs and data visualizations", in *Proc. 30th ACM Symp. on User Interface Software and Technology*, ACM, pp. 57-69. (Cited on p. 58.)

Cham, Jorge and Daniel Whiteson

2017 *We Have No Idea: A Guide to the Unknown Universe*, Penguin. (Cited on p. 104.)

Chen, Helen, Sophie Engle, Alark Joshi, Eric D Ragan, Beste F Yuksel, and Lane Harrison

2018 "Using Animation to Alleviate Overdraw in Multiclass Scatterplot Matrices", in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ACM, p. 417. (Cited on p. 56.)

Chen, Min and Amos Golan

2016 "What may visualization processes optimize?", *IEEE Transactions on Visualization and Computer Graphics*, 22, 12, pp. 2619-2632. (Cited on pp. 1, 3, 11, 18.)

Chen, Min and Heike Jänicke

2010 "An information-theoretic framework for visualization." *IEEE Transactions on Visualization and Computer Graphics*, 16, 6, pp. 1206-15. (Cited on pp. 11, 14, 18.)

Cheng, Eugenia

2008 "Categories", in *The Princeton Companion to Mathematics*, ed. by Timothy Gowers, Princeton University Press. (Cited on p. 80.)

Chuah, Mei C.

1998 "Dynamic aggregation with circular visual designs", in *Proceedings IEEE Symposium on Information Visualization*, pp. 1-9. (Cited on p. 22.)

Collins, Christopher, Sheelagh Carpendale, and Gerald Penn

2009a "DocuBurst: Visualizing document content using language structure", en, *Computer Graphics Forum*, 28, 3 (June 2009), pp. 1039-1046. (Cited on p. 32.)

Collins, Christopher, Gerald Penn, and Sheelagh Carpendale

2009b "Bubble Sets: Revealing Set Relations with Isocontours over Existing Visualizations", *IEEE Transactions on Visualization and Computer Graphics*, 15, 6, pp. 1009, 1016. (Cited on p. 117.)

Cook, Kristin, Georges Grinstein, and Mark Whiting

2014 *The VAST challenge: History, scope, and outcomes: An introduction to the special issue*. (Cited on p. 78.)

Correll, Michael and Jeffrey Heer

2017 "Surprise! Bayesian Weighting for De-Biasing Thematic Maps", *IEEE transactions on visualization and computer graphics*, 23, 1, pp. 651-660. (Cited on pp. 14, 115.)

Correll, Michael, Mingwei Li, Gordon Kindlmann, and Carlos Scheidegger

2018 "Looks Good To Me: Visualizations As Sanity Checks", *IEEE transactions on visualization and computer graphics*. (Cited on p. 1.)

Croner, Lisa J and Thomas D Albright

1997 "Image segmentation enhances discrimination of motion in visual noise", *Vision Research*, 37, 11, pp. 1415-1427. (Cited on pp. 54, 55.)

Cui, Qingguang, Matthew Ward, Elke Rundensteiner, and Jing Yang

2006 "Measuring Data Abstraction Quality in Multiresolution Visualizations", *IEEE Transactions on Visualization and Computer Graphics*, 12, 5, pp. 709-716. (Cited on p. 22.)

Dick, Miri, Shimon Ullman, and Dov Sagi

1987 "Parallel and serial processes in motion detection", *Science*, 237, 4813, pp. 400-402. (Cited on pp. 51, 57.)

Duncan, John and Glyn W Humphreys

1989 "Visual search and stimulus similarity." *Psychological review*, 96, 3, p. 433. (Cited on pp. 51, 56.)

Dunne, Cody, Steven I Ross, Ben Shneiderman, and Mauro Martino

2015 "Readability metric feedback for aiding node-link visualization designers", *IBM Journal of Research and Development*, 59, 2/3, pp. 14-1. (Cited on p. 18.)

Ellis, Geoffrey and Alan Dix

2006 "Enabling automatic clutter reduction in parallel coordinate plots", *IEEE Transactions on Visualization and Computer Graphics*, 12, 5, pp. 717-723. (Cited on p. 18.)

2007 "A taxonomy of clutter reduction for information visualisation", *IEEE Transactions on Visualization and Computer Graphics*, 13, 6, pp. 1216-1223. (Cited on p. 21.)

Elmqvist, Niklas and Jean-Daniel Fekete

2010 "Hierarchical aggregation for information visualization: overview, techniques, and design guidelines." *IEEE Transactions on Visualization and Computer Graphics*, 16, 3, pp. 439-54. (Cited on pp. 21, 49.)

Endert, Alex, Christopher Andrews, Yueh Hua Lee, and Chris North

2011 "Visual encodings that support physical navigation on large displays", *Proceedings of Graphics Interface 2011 (GI '11)*, pp. 103-110. (Cited on p. 116.)

Endert, Alex, Patrick Fiaux, and Chris North

2012 "Semantic interaction for visual text analytics", in *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, pp. 473-482. (Cited on p. 116.)

Etemadpour, Ronak and Angus Graeme Forbes

2017 "Density-based motion", *Information Visualization*, 16, 1, pp. 3-20. (Cited on p. 56.)

Etemadpour, Ronak, Paul Murray, and Angus Graeme Forbes

2014 "Evaluating density-based motion for big data visual analytics", in *Proc. of the IEEE Int. Conf. on Big Data*, IEEE, pp. 451-460. (Cited on p. 56.)

Faust, Rebecca, David Glickenstein, and Carlos Scheidegger

2017 "DimReader: Axis lines that explain non-linear projections", *arXiv preprint arXiv:1710.00992*. (Cited on p. 1.)

Fekete, Jean-Daniel and Catherine Plaisant

2002 "Interactive Information Visualization of a Million Items", in *Proceedings of the IEEE Symposium on Information Visualization*, INFOVIS '02, IEEE Computer Society, Washington, DC, USA, pp. 117-. (Cited on p. 15.)

Fink, Martin, Jan-Henrik Haunert, Joachim Spoerhase, and Alexander Wolff

2013 "Selecting the aspect ratio of a scatter plot based on its delaunay triangulation", *IEEE transactions on visualization and computer graphics*, 19, 12, pp. 2326-2335. (Cited on p. 16.)

Garner, Wendell R

2014 *The processing of Information and Structure*, Psychology Press. (Cited on p. 73.)

Gleicher, Michael, M. Correll, C. Nothelfer, and S. Franconeri

2013 "Perception of Average Value in Multiclass Scatterplots", *IEEE Transactions on Visualization and Computer Graphics*, 19, 12 (Dec. 2013), pp. 2316-2325. (Cited on p. 55.)

Haroz, Steve and David Whitney

2012 "How Capacity Limits of Attention Influence Information Visualization Effectiveness", *IEEE Transactions on Visualization and Computer Graphics*, 18, 12 (Dec. 2012), pp. 2402-2410. (Cited on pp. 4, 21, 40.)

Harper, Jonathan and Maneesh Agrawala

2014 "Deconstructing and restyling D3 visualizations", in *Proceedings of the 27th annual ACM Symposium on User interface Software and Technology*, ACM, pp. 253-262. (Cited on p. 85.)

Harrison, Lane, Fumeng Yang, Steven Franconeri, and Remco Chang

2014 "Ranking Visualizations of Correlation Using Weber's Law." *IEEE transactions on visualization and computer graphics*, 20, 12, pp. 1943-1952. (Cited on p. 4.)

Healey, Christopher and James Enns

2011 "Attention and Visual Memory in Visualization and Computer Graphics", *IEEE Transactions on Visualization and Computer Graphics* (July 2011), pp. 1-20. (Cited on p. 52.)

Heer, Jeffrey and Michael Bostock

2010 "Crowdsourcing Graphical Perception: Using Mechanical Turk to Assess Visualization Design", in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, ACM, Atlanta, Georgia, USA, pp. 203-212. (Cited on p. 55.)

Hofmann, Heike, Lendie Follett, Mahbubul Majumder, and Dianne Cook

2012 "Graphical tests for power comparison of competing designs", *IEEE Transactions on Visualization and Computer Graphics*, 18, 12, pp. 2441-2448. (Cited on p. 121.)

Holz, Christian and Steven Feiner

2009 "Relaxed selection techniques for querying time-series graphs", in *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, ACM, pp. 213-222. (Cited on p. 13.)

Hu, Yueqi, Tom Polk, Jing Yang, Ye Zhao, and Shixia Liu

2016 "Spot-tracking lens: A zoomable user interface for animated bubble charts", in *IEEE Pacific Visualization Symposium*, IEEE, pp. 16-23. (Cited on p. 117.)

Huber, Daniel E and Christopher G Healey

2005 "Visualizing data with motion", in *Visualization, 2005. VIS 05. IEEE*, IEEE, pp. 527-534. (Cited on p. 55.)

Interactive Data Lab

2018 *Vega Lite Example Gallery*, `https://vega.github.io/vega-lite/examples/` (visited on 09/30/2018). (Cited on p. 85.)

Itti, Laurent, Christof Koch, and Ernst Niebur

1998 "A model of saliency-based visual attention for rapid scene analysis", *IEEE Transactions on pattern analysis and machine intelligence*, 20, 11, pp. 1254-1259. (Cited on pp. 45, 57.)

Johansson, Jimmy, Camilla Forsell, Mats Lind, and Matthew Cooper

2008 "Perceiving patterns in parallel coordinates: determining thresholds for identification of relationships", *Information Visualization*, 7, 2, pp. 152-162. (Cited on pp. 12, 13.)

Judd, Tilke, Frédo Durand, and Antonio Torralba

2012 "A Benchmark of Computational Models of Saliency to Predict Human Fixations", in *MIT Technical Report*. (Cited on p. 58.)

Kaggle

2017 *The State of Data Science & Machine Learning*. (Cited on p. 2.)

Keim, D.A., F. Mansmann, J. Schneidewind, and H. Ziegler

2006 "Challenges in Visual Data Analysis", in *Proc. of the Int. Conv. on Information Visualisation*, IEEE, pp. 9-16. (Cited on p. 20.)

Kijmongkolchai, Natchaya, Alfie Abdul-Rahman, and Min Chen

2017 "Empirically measuring soft knowledge in visualization", in *Computer Graphics Forum*, 3, Wiley Online Library, vol. 36, pp. 73-85. (Cited on p. 115.)

Kim, Younghoon and Jeffrey Heer

2018 "Assessing effects of task and data distribution on the effectiveness of visual encodings", in *Computer Graphics Forum*, 3, Wiley Online Library, vol. 37, pp. 157-167. (Cited on pp. 1, 52, 101-103, 105, 106.)

Kindlmann, Gordon and Carlos Scheidegger

2014 "An Algebraic Process for Visualization Design", English, *IEEE Transactions on Visualization and Computer Graphics*, 20, 12 (Dec. 2014), pp. 2181-2190. (Cited on pp. 1, 73, 80.)

Kondo, Brittany and Christopher Collins

2014 "DimpVis: Exploring Time-varying Information Visualizations by Direct Manipulation", *IEEE Transactions on Visualization and Computer Graphics*, 20, 12 (Dec. 2014), pp. 2003-2012. (Cited on p. 117.)

Koutra, Danai, U Kang, Jilles Vreeken, and Christos Faloutsos

2015 "Summarizing and Understanding Large Graphs", *Statistical Analysis and Data Mining*, 8, 3, pp. 183-202, arXiv: 1206.3552. (Cited on p. 22.)

Krzywinski, Martin I, Jacqueline E Schein, Inanc Birol, Joseph Connors, Randy Gascoyne, Doug Horsman, Steven J Jones, and Marco A Marra

2009 "Circos: An information aesthetic for comparative genomics", *Genome Research*, eprint: `http://genome.cshlp.org/content/early/2009/06/15/gr.092759.109.full.pdf+html`. (Cited on p. 16.)

Lamarche-Perrin, Robin, Yves Demazeau, and Jean-marc Vincent

2014 "Building Optimal Macroscopic Representations of Complex Multi-agent Systems", in *Trans. on Computational Collective Intelligence XV*, ed. by N.-T. Nguyen, R. Kowalczyk, J.M. Corchado, and J. Bajo, Springer Berlin Heidelberg, vol. 8670, pp. 1-27. (Cited on p. 22.)

Lamarche-Perrin, Robin, Lucas Mello Shnorr, Jean-marc Vincent, and Yves Demazeau

2012 *Evaluating Trace Aggregation Through Entropy Measures for Optimal Performance Visualization of Large Distributed Systems*, tech. rep., INRIA, pp. 1-21. (Cited on p. 22.)

Lee, Bongshin, Rubaiat Habib Kazi, and Greg Smith

2013 "SketchStory: Telling more engaging stories with data through freeform sketching", *IEEE Transactions on Visualization and Computer Graphics*, 19, 12, pp. 2416-2425. (Cited on p. 13.)

Lee, Thomas C M

1999 "A Minimum Description Length Based Image Segmentation Procedure , and Its Comparison with a Cross-Validation-Based Segmentation Procedure", *Journal of the American Statistical Association*, 95, 1995, pp. 259-270. (Cited on p. 25.)

2001 "An Introduction to Coding Theory and the Two-Part Minimum Description Length Principle", *International Statistical Review*, 69, 2, pp. 169-183. (Cited on p. 25.)

Li, Hang and Naoki Abe

1998 "Generalizing Case Frames Using a Thesaurus and the MDL Principle", *Computational linguistics*, 24, 2, pp. 217-244, arXiv: `9507011 [cmp-lg]`. (Cited on pp. 25, 27, 29.)

Lins, Lauro, James T Klosowski, and Carlos Scheidegger

2013 "Nanocubes for real-time exploration of spatiotemporal datasets", *IEEE Transactions on Visualization and Computer Graphics*, 19, 12, pp. 2456-2465. (Cited on p. 15.)

Liu, Yang and Jeffrey Heer

2018 "Somewhere over the rainbow: An empirical assessment of quantitative colormaps", in *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*. (Cited on p. 62.)

Liu, Zhicheng and Jeffrey Heer

2014 "The effects of interactive latency on exploratory visual analysis", *IEEE Transactions on Visualization and Computer Graphics*, 1, pp. 1-1. (Cited on pp. 13, 15.)

Liu, Zhicheng, Biye Jiang, and Jeffrey Heer

2013 "imMens: Real-time Visual Querying of Big Data", in *Computer Graphics Forum*, 3pt4, Wiley Online Library, vol. 32, pp. 421-430. (Cited on p. 15.)

Loorak, Mona Hosseinkhani, Charles Perin, Christopher Collins, and Sheelagh Carpendale

2017 "Exploring the possibilities of embedding heterogeneous data attributes in familiar visualizations", *IEEE transactions on visualization and computer graphics*, 23, 1, pp. 581-590. (Cited on p. 16.)

Mackinlay, Jock

1986 "Automating the design of graphical presentations of relational information", *Acm Transactions On Graphics (Tog)*, 5, 2, pp. 110-141. (Cited on p. 80.)

Marquis, Jean-Pierre

2015 "Category Theory", in *The Stanford Encyclopedia of Philosophy*, ed. by Edward N. Zalta, Winter 2015, Metaphysics Research Lab, Stanford University. (Cited on p. 80.)

Matejka, Justin and George Fitzmaurice

2017 "Same stats, different graphs: generating datasets with varied appearance and identical statistics through simulated annealing", in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ACM, pp. 1290-1294. (Cited on pp. 9, 79.)

Matzen, Laura E, Michael J Haass, Kristin M Divis, Zhiyuan Wang, and Andrew T Wilson

2018 "Data Visualization Saliency Model: A Tool for Evaluating Abstract Data Visualizations", *IEEE Transactions on Visualization & Computer Graphics*, 1, pp. 563-573. (Cited on pp. 57, 58.)

Mayorga, Adrian and Michael Gleicher

2013 "Splatterplots: Overcoming overdraw in scatter plots", *IEEE Transactions on Visualization and Computer Graphics*, 19, 9, pp. 1526-1538. (Cited on pp. 10, 14.)

McInnes, Leland and John Healy

2018 "Umap: Uniform manifold approximation and projection for dimension reduction", *arXiv preprint arXiv:1802.03426*. (Cited on p. 15.)

McLeod, Peter, Jon Driver, and Jennie Crisp

1988 "Visual search for a conjunction of movement and form is parallel", *Nature*, 332, 6160, p. 154. (Cited on pp. 51, 54, 55.)

Méndez, Gonzalo Gabriel, Miguel A Nacenta, and Sebastien Vandenheste

2016 "iVoLVER: Interactive visual language for visualization extraction and reconstruction", in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ACM, pp. 4073-4085. (Cited on p. 85.)

Menne, Matthew J, Imke Durre, Russell S Vose, Byron E Gleason, and Tamara G Houston

2012 "An overview of the global historical climatology network-daily database", *Journal of Atmospheric and Oceanic Technology*, 29, 7, pp. 897-910. (Cited on p. 101.)

Miller, G a

1956 "The magical number seven, plus or minus two: some limits on our capacity for processing information." *Psychological review*, 101, 2, pp. 343-352. (Cited on p. 21.)

Moorthy, Anush Krishna and Alan Conrad Bovik

2009 "Visual importance pooling for image quality assessment", *IEEE journal of selected topics in signal processing*, 3, 2, pp. 193-201. (Cited on p. 87.)

Moritz, Dominik, Chenglong Wang, Greg L Nelson, Halden Lin, Adam M Smith, Bill Howe, and Jeffrey Heer

2018 "Formalizing visualization design knowledge as constraints: actionable and extensible models in Draco", *IEEE transactions on visualization and computer graphics*. (Cited on pp. 52, 114.)

Munzner, Tamara

2009 "A Nested Model for Visualization Design and Validation", *IEEE Transactions on Visualization and Computer Graphics*, 15, 6 (Nov. 2009), pp. 921-928. (Cited on p. 76.)

2014 *Visualization Analysis & Design*, CRC Press. (Cited on p. 40.)

Ninassi, Alexandre, Olivier Le Meur, Patrick Le Callet, and Dominique Barba

2007 "Does where you gaze on an image affect your perception of quality? Applying visual attention to image quality metric", in *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, IEEE, vol. 2, pp. II-169. (Cited on p. 87.)

Olsson, Donald M and Lloyd S Nelson

1975 "The Nelder-Mead simplex procedure for function minimization", *Technometrics*, 17, 1, pp. 45-51. (Cited on p. 33.)

Pahins, Cicero AL, Sean A Stephens, Carlos Scheidegger, and Joao LD Comba

2017 "Hashedcubes: Simple, low memory, real-time visual exploration of big data", *IEEE transactions on visualization and computer graphics*, 23, 1, pp. 671-680. (Cited on p. 15.)

Pandey, Anshul Vikram, Josua Krause, Cristian Felix, Jeremy Boy, and Enrico Bertini

2016 "Towards Understanding Human Similarity Perception in the Analysis of Large Sets of Scatter Plots", in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, ACM, San Jose, California, USA, pp. 3659-3669. (Cited on pp. 93-96, 120.)

Papathomas, Thomas V, Andrei Gorea, and Bela Julesz

1991 "Two carriers for motion perception: Color and luminance", *Vision Research*, 31, 11, pp. 1883-1892. (Cited on pp. 54, 55.)

Pirolli, Peter and Stuart Card

1999 "Information foraging." *Psychological review*, 106, 4, p. 643. (Cited on p. 3.)

Popper, Karl

2005 *The logic of scientific discovery*, Routledge. (Cited on p. 77.)

Purchase, Helen C

2002 "Metrics for graph drawing aesthetics", *Journal of Visual Languages & Computing*, 13, 5, pp. 501-516. (Cited on p. 18.)

Quinlan, J Ross and R Rivest

1989 "Inferring Decision Trees Using the Minimum Description Length Principle", *Information and Computation*, 80, 1989, pp. 227-248. (Cited on p. 25.)

Rigau, Jaume, Miquel Feixas, and Mateu Sbert

2008 "Informational aesthetics measures", *IEEE Computer Graphics and Applications*, 28, 2, pp. 24-34. (Cited on p. 81.)

Rissanen, Jorma

1983 "A universal prior for integers and estimation by minimum description length", *Annals of Statistics*, 11, 2, pp. 416-431. (Cited on pp. 25, 26.)

1989 *Stochastic Complexity in Statistical Inquiry*, World Scientific Publishing Co., Inc., River Edge, NJ, USA, p. 188. (Cited on p. 26.)

Robertson, George, Roland Fernandez, Danyel Fisher, Bongshin Lee, and John Stasko

2008 "Effectiveness of animation in trend visualization", *IEEE Transactions on Visualization and Computer Graphics*, 14, 6, pp. 1325-1332. (Cited on pp. 55, 117.)

Rodrigues, Nils and Daniel Weiskopf

2018 "Nonlinear Dot Plots", *IEEE Transactions on Visualization and Computer Graphics*, 24, 1, pp. 616-625. (Cited on pp. 77, 79.)

Rosenholtz, Ruth

1999 "A Simple Saliency Model Predicts a Number of Motion Popout Phenomena", *Vision Research*, 39, pp. 3157-3163. (Cited on pp. 51, 54, 56, 57.)

Rosenholtz, Ruth, Yuanzhen Li, Zhenlan Jin, and Jonathan Mansfield

2010 "Feature congestion: A measure of visual clutter", *Journal of Vision*, 6, 6, pp. 827-827. (Cited on pp. 18, 21, 22, 40, 44, 73.)

Rosenholtz, Ruth, Yuanzhen Li, and Lisa Nakano

2007 "Measuring visual clutter", *Journal of Vision*, 7, 2, pp. 17.1-22. (Cited on pp. 21, 57.)

Rosenholtz, Ruth, Allen L. Nagy, and Nicole R. Bell

2004 "The effect of background color on asymmetries in color search", *Journal of Vision*, 4, 3 (Mar. 2004), pp. 224-240. (Cited on p. 57.)

Saket, Bahador, Hannah Kim, Eli T Brown, and Alex Endert

2017 "Visualization by demonstration: An interaction paradigm for visual data exploration", *IEEE Transactions on Visualization and Computer Graphics*, 23, 1, pp. 331-340. (Cited on p. 13.)

Saket, Bahador, Arjun Srinivasan, Eric D Ragan, and Alex Endert

2018 "Evaluating interactive graphical encodings for data visualization", *IEEE Transactions on Visualization and Computer Graphics*, 24, 3, pp. 1316-1330. (Cited on p. 13.)

Saw, Jimmy HW, Michael Schatz, Mark V Brown, Dennis D Kunkel, Jamie S Foster, Harry Shick, Stephanie Christensen, Shaobin Hou, Xuehua Wan, and Stuart P Donachie

2013 "Cultivation and complete genome sequencing of Gloeobacter kilaueensis sp. nov., from a lava cave in Kilauea Caldera, Hawaii", *PloS one*, 8, 10, e76376. (Cited on p. 17.)

Shneiderman, Ben

1983 "Direct manipulation: A step beyond programming languages", *Computer*, 8, pp. 57-69. (Cited on p. 13.)

1996 "The eyes have it: A task by data type taxonomy for information visualizations", in *Visual Languages, 1996. Proceedings., IEEE Symposium on*, IEEE, pp. 336-343. (Cited on p. 20.)

2008 "Extreme Visualization: Squeezing a Billion Records into a Million Pixels", in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, ACM, Vancouver, Canada, pp. 3-12. (Cited on p. 15.)

Simons, Daniel J., Steven L. Franconeri, and Rebecca L. Reimer

2000 "Change blindness in the absence of a visual disruption", *Perception*, 29, 10, pp. 1143-1154. (Cited on p. 74.)

Smith, Nathaniel and Stefan van der Walt

2015 *A Better Default Colormap for Matplotlib*. (Cited on p. 62.)

Szafir, Danielle Albers, Steve Haroz, Michael Gleicher, and Steven Franconeri

2016 "Four types of ensemble coding in data visualizations", *Journal of vision*, 16, 5, pp. 11-11. (Cited on pp. 11, 55, 57.)

Tatu, Andrada, Georgia Albuquerque, Martin Eisemann, Peter Bak, Holger Theisel, Marcus Magnor, and Daniel Keim

2011 "Automated analytical methods to support visual exploration of high-dimensional data", *IEEE Transactions on Visualization and Computer Graphics*, 17, 5, pp. 584-597. (Cited on p. 16.)

Töpfer, F. and W. Pillewizer

1966 "The Principles of Selection", *The Cartographic Journal*, 3, 1, pp. 10-16. (Cited on p. 22.)

Treisman, Anne M and Garry Gelade

1980 "A feature-integration theory of attention", *Cognitive Psychology*, 12, 1, pp. 97-136. (Cited on p. 52.)

Van der Maaten, Laurens and Geoffrey Hinton

2008 "Visualizing data using t-SNE", *Journal of machine learning research*, 9, Nov, pp. 2579-2605. (Cited on pp. 2, 15.)

Van Goethem, Arthur, Frank Staals, Maarten Löffler, Jason Dykes, and Bettina Speckmann

2017 "Multi-granular trend detection for time-series analysis", *IEEE transactions on visualization and computer graphics*, 23, 1, pp. 661-670. (Cited on p. 14.)

Van Wijk, Jarke J. and Huub van de Wetering

1999 "Cushion Treemaps: Visualization of Hierarchical Information", in *Proceedings of the IEEE Symposium on Information Visualization*, INFOVIS '99, IEEE Computer Society, Washington, DC, USA, pp. 73-. (Cited on p. 16.)

Von Mühlenen, Adrian and Hermann J Müller

2000 "Perceptual integration of motion and form information: Evidence of parallel-continuous processing", *Perception & Psychophysics*, 62, 3, pp. 517-531. (Cited on pp. 51, 54, 55.)

Wagner, Andreas

2000 "Enriching a lexical semantic net with selectional preferences by means of statistical corpus analysis", in *Proceedings of the European Conference on Artificial Intelligence - ECAI*. (Cited on p. 32.)

Wall, Emily, Leslie M Blaha, Lyndsey Franklin, and Alex Endert

2017 "Warning, bias may occur: A proposed approach to detecting cognitive bias in interactive visual analytics", in *IEEE Conference on Visual Analytics Science and Technology (VAST)*. (Cited on p. 14.)

Wang, Zhe, Nivan Ferreira, Youhao Wei, Aarthy Sankari Bhaskar, and Carlos Scheidegger

2017 "Gaussian cubes: Real-time modeling for visual exploration of large multidimensional datasets", *IEEE transactions on visualization and computer graphics*, 23, 1, pp. 681-690. (Cited on p. 15.)

Wang, Zhou, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al.

2004 "Image quality assessment: from error visibility to structural similarity", *IEEE transactions on image processing*, 13, 4, pp. 600-612. (Cited on p. 81.)

Wang, Zhou and Qiang Li

2011 "Information content weighting for perceptual image quality assessment", *IEEE Transactions on Image Processing*, 20, 5, pp. 1185-1198. (Cited on p. 87.)

Wang, Zhou, Eero Simoncelli, Alan Bovik, et al.

2003 "Multi-scale structural similarity for image quality assessment", in *ASILOMAR CONFERENCE ON SIGNALS SYSTEMS AND COMPUTERS*, IEEE; 1998, vol. 2, pp. 1398-1402. (Cited on pp. 84, 98, 115.)

Ware, Colin

2004 *Information Visualization: Perception for Design*, 2nd, Morgan Kaufmann Publishers Inc., San Francisco. (Cited on p. 51.)

Wilkinson, Leland

2006 *The grammar of graphics*, Springer Science & Business Media. (Cited on p. 13.)

Wilkinson, Leland and Graham Wills

2008 "Scagnostics distributions", *Journal of Computational and Graphical Statistics*, 17, 2, pp. 473-491. (Cited on pp. 13, 93.)

Wolfe, Jeremy M

1998a "Visual Search", *Attention*, pp. 1-41. (Cited on pp. 21, 40.)

1998b "What can 1 million trials tell us about visual search?", *Psychological Science*, 9, 1, pp. 33-39. (Cited on pp. 51, 54.)

Woodruff, Allison, James Landay, and Michael Stonebraker

1998 "Constant density visualizations of non-uniform distributions of data", in *Proceedings of the 11th annual ACM symposium on User interface software and technology - UIST '98*, pp. 19-28. (Cited on pp. 22, 46.)

Wu, Yingcai, Xiaotong Liu, Shixia Liu, and Kwan Liu Ma

2013 "ViSizer: A visualization resizing framework", *IEEE Transactions on Visualization and Computer Graphics*, 19, 2, pp. 278-290. (Cited on p. 22.)

Yoghourdjian, Vahan, Tim Dwyer, Karsten Klein, Kimbal Marriott, and Michael Wybrow

2018 "Graph Thumbnails: Identifying and Comparing Multiple Graphs at a Glance", *IEEE Transactions on Visualization and Computer Graphics*, 1, pp. 1-1. (Cited on p. 15.)

Yost, Beth and Chris North

2006 "The perceptual scalability of visualization", *IEEE Transactions on Visualization and Computer Graphics*, 12, 5, pp. 837-844. (Cited on p. 20.)

Zgraggen, Emanuel, Zheguang Zhao, Robert Zeleznik, and Tim Kraska

2018 "Investigating the Effect of the Multiple Comparisons Problem in Visual Analysis", in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ACM, p. 479. (Cited on pp. 1, 14.)

Zhao, Hang, Orazio Gallo, Iuri Frosio, and Jan Kautz

2017 "Loss functions for image restoration with neural networks", *IEEE Transactions on Computational Imaging*, 3, 1, pp. 47-57. (Cited on p. 121.)

Zheng, Zhixiong, Haibo Cheng, Zijian Zhang, Yiming Zhao, and Ping Wang

2018 "An Alternative Method for Understanding User-Chosen Passwords", *Security and Communication Networks*, 2018. (Cited on pp. 85, 86.)