Universität Münster

Department of Geoscience

Modeling the performance of interaction techniques for the comparison of spatial entities in the context of geo-dashboards

Bachelorthesis

Institute for Geoinformatics

Principal supervisor:	Ph.D Auriol Degbelo
Supervisor:	Dr. Thomas Bartoschek
Author:	Simon Meißner
Matriculation number:	462170
Field of Study:	Geoinformatics B.Sc.
Contact Details:	s_meis06@uni-muenster.de
Submission date:	November 9, 2023

Abstract

Contents

1 Introduction

The growing usage of dashboards to represent data across a range of different fields suggests a need for research on layout and design features of dashboards and their influence on the user experience. Previous research has shown that there is no one-fits-all solution for the design of dashboards [24, 20].

This existing agreement that general design recommendations for dashboards are not sufficient and/or possible ask for a breakdown of components that constitute dashboards. One approach is to identify user interactions that are possible when visualized data is explored or analysed. The field of geovisualizations and geodashboards is enriched with many different perspectives that all try to define taxonomies and or classifications models for possible user interactions [1, 7]. Many have their reasonable own application for spatial-temporal information visualization. But the minority of these classifications and taxonomies are empirically-derived. The proposed framework of Roth represents an exception. He has shown that a functional taxonomy of interaction primitives can be empirically derived. He identified general tasks users want to accomplish (objective primitives) [19]. Besides narrowing down the scope to one of Roths derived objective primitives this work will also look at this topic from the perspective of different interaction techniques.

An interaction technique as broadly defined in the Computer Science Handbook from 2004 is "the fusion of input and output, consisting of all hardware and software elements, that provides a way for the user to accomplish a task." [11]. Input describes all sensed information about the physical environment to the computer. Output from computers on the other hand include any emission or modification to the physical environment. In the context of geovisualizations and geodashboards, interaction techniques have been researched [12, 14, 21]. Roth also describes an interaction technique in the context of geovisualizations as the functionality of an given interface and the procedures of manipulating its visualizations [19].

This work will deal with the derived objective primitive of *comparison* from Roth's work. But not only Roth writes about comparison. Wehrend describes *compare* as a seperate operation class in visualization problem [23] and Brehmer et al. speak of comparison as a low level visualization task [3]. Also Buja et al. propose *comparison* to be one of the three fundamental plot manipulations in data visualization [4]. In the scope of geovisualizations Crampton identified *compare* as an interactivity task [7] and Gorte and Degbelo argue that *comparison* is a basic task that is relevant in exploratoy and confirmatory analysis [10]. Buja distingiushes between two dimensions of comparison. The first describes the goal of comparing different variables or projections of the whoel dataset. The second describes the goal of comparing subsets of the whole dataset against each other [4]. This work will only focus on the latter.

We will examine two broadly used interaction techniques: *filtering* and *highlighting* [12, 19]. Keim et al. describe *filtering* as a combination of selection and view enhancement and Roth attributes filtering to identifying matches from user-defined conditions. The literature often use the term *brushing* to describe *highlighting*. They can be considered synonymous in the scope of interaction techniques as both describe the process of visualy emphasizing a subset from the whole dataset. Historically to define the subset the process started by drawing a rectangle directly in the view with the mouse which was called *brush*. Which explains the term *brushing*. For the rest of this work we will use the term *highlighting*. Keim et al. state that *highlighting* is often combined with linking which describes the process of selected data beeing communicated to other views of the data. They follow one of the proposed user strategies *Select Subset* from Gleicher [9].

In this work we will investigate how these interaction techniques influence user performance in the context of comparing subsets in geodashboards. Because the interaction technique is by far not the only variable that can be changed, we also want to observe the influence of different variables on the user performance. To provide a starting point backed with emperical data we want to derive a mathematical model that should display user performance in dependence from different variables which are described later. Therefore we can infer two research questions for this work:

1. Which mathematical models best describe user performance during the comparison of spatial entities in the context of geodashboards?

2. Which interaction technique best supports the task of comparison in the context of geodashboards?

To answer these questions we conducted a user study in which participants try to answer questions with the goal of finding differences and/or similarities of subsets of spatial-/temporal datasets. To answer the questions they are using a specially builded digital web-prototype with six different dashboard variants. The dashboards vary in their interaction technique and some render additional views utilizing *explicit encoding* as it is defined as one of the basic designs for visual comparison [9]. The goal of the experiment is to collect data about the user performance. We have defined user performance to be two dimensional. First want to know about the time it takes to answer questions. It is important to note that it does not matter wether a correct answer was given. The second dimension is about accuracy. This seperatly tracks wether an answer was correct or not. After collecting the data we want to use that data to derive mathematical models that best approximate answer time and accuracy during the comparison of features in geodashboards. With special interest for the differences between the selected interaction techniques. We want to learn about the different factors we have included and how they influence answer time and accuracy in this setting.

To get an overview of the current state of research, chapter 2 will provide information about scientific contributions on interaction techniques in geo-dashboards and comparison. We will also give an short introduction to multiple coordinated because we are utilizing such a system in our prototype. Section 3 will describe key concepts of the mathematical models and what constitues them. In Section 4 details about the digital web-prototype and how it was built are presented. How the experiment was designed and what factors that possibly influence comparison performance were considered are covered in section 5. Section 6 will present the results of the experiment and propose our found mathematical models that showed the best testing results. As this experiment only covers a selection of possible factors that possible influence difficulty and accuracy during the comparison of features in geodashboards, this work should be a starting point for further research. Because comparison can be of many different kinds and cover different scopes this work also opens the door for more reseach in different comparison settings. Section 7 discusses such limitations in depth, how our research questions can be answered and how our findings can be transferred to other domains. Lastly we will sumarize our key-learnings and propose future work that has to be done in section 8.

2 Related Work

In this chapter we want to summarize existing research with regard to our two interaction techniques: *highlighting* and *filtering*. First we will look at how the use of *highlighting* and *filtering* in the context of geodashboards is analysed. Second, we will also examine the interaction techniques in the context of multiple coordinated views because of its central role in our web-prototype.

2.1 Geo-dashboards

2.2 Multiple coordinated views

Multiple Coordinated Views (MCV) is a specific exploratory visualization technique that allow users to explore data. It consists of multiple views that all encode the same data in different representations. Interactions and operations of the user are managed and synchronised between views [18]. Buja et al. argue that beeing able to pose queries graphically and viewing the response in the same visual field is a fundamental component [4]. The principle is extensivly accepted and researched and proven to increase user performance, discovery of unforeseen relationships and unification of the desktop [16]. Buja et al. even include it in their taxonomy for data visualization. They even mention *highlighting*, one of our two researched interaction techniques, to often be a substantial part of MCV as it is one way pose queries about subsets graphically [4].

Because of its popularity 20-30 years ago much of previous research in the field of MCV focused on scatterplot matrices [5, 2]. Lawrence et al. used a specfic tool for the exploratory analysis of systems biology data and showed how *highlighting* is an effective technique for discovering outliers [13]. Carr et al. argue that *highlighting* is the most common interaction technique in scatterplot matrices when working with subsets [5, 2]. On the other hand they argue that if the subsets becomes larger another approach may be more suitable. They called it *specify, then compute*. The idea is that a selection region is defined, similar to the highlighting procedure, then the subset is computed and finally a new display is rendered that only contains the subset. This describes our more modern idea of *filtering*.

Some contributions on interaction techniques in MCV systems define *filtering* as a type of *highlighting* (*brushing*) [22].

2.3 Comparison

Gleicher writes about four considerations when visualizing comparison [9]. At first we have to identify our comparison elements. Because every comparison task in our study focuses on comparing two or three features of the dataset we can descibe our targets as 'explicit targets' as every item is known and already available. From Gleichers proposed actions on relationships between targets our comparison task fall into the *Identify* and *Measure/Quantify/Summarize* categories. Second we have to identify comparitive challenges. The number of targets should not add much complexity as we are only using two or three targets. Because we are using timeseries with only one observed variable the complexity of each individual item is also fairly low. As we are only identifying and measuring direct differences or differences between differences of targets the complexity of the relationships is low to moderate. Gleicher next proposes to deal with a scalability strategie. To reduce scale challenges the strategies of *Select Subset* and *Summarize Somehow* are utilized. In all dashboard variants either *filtering* or *highlighting* as an interaction technique is used which help with scale because subsets are created. In some variants additional views are rendered that already encode the difference of two targets which summarize a relation. Lastly we have to consider design visualizations. Across all dashboard variants all three basic visual designs for comparison are utilized as each have their benefits and drawbacks. Because we deal with temporal data all graph views are utilizing *superposition*. To reduce scalability problems either *filtering* or *highlighting* are utilized as already mentioned. Our table views use a combination of *super* - and *juxtapostion* as showing each datapoint in the same place would hinder readability. Finally because our actions on the targets include comparing differences between targets we included *explicit encoding* in some dashboard variants.

3 Digital Web-prototype

The prototype was build using the open-source web development framework next.js which provides react-based javascript or typescript applications with server-side rendering and static website generation. It was mostly selected because of its developer-friendly ecosystem especially regarding deployement. Another reason for this stack was the existing experience with react-based typescript and the light-weight and modern setup. The whole application makes use of a few react libraries. The most important beeing the popular "react-leaflet" library which drastically simplifies using leaflet maps in react applications. Other important libraries are: "zustand" to help with coordination, "recharts" to help building time lines and "tailwindcss" to ease building the css of the application. Because the experiment was designed to be held online the application needed to be deployed. For that we used the free deployment plan of vercel.com. Because the application only used non-personal static data there was no need for implementing a backend or an authentication system. Although the prototype took responsive behaviour into account the screen dimension of the user's device should not fall below 1280x720px.

3.1 Data

The data used in the web-prototype consists of two components. First we needed geodata of the states of germany. In terms of performance, low-resolution geojson data with a reference scale of 1:5000000 was used. The data is open and freely downloadable from the german federal agency for cartography and geodesy [8]. The questions in the study were about static spatio-temporal datasets which were about changes of thematic properties expressed through values of attributes. Meaning qualitative changes with numeric characteristics. According to Andrienko et al. this is one of three types of spatio-temporal data [1]. On the other hand spatio-temporal datasets could have existential changes, features appear and disappear over time, or they could have changes in their spatial properties. Because we wanted to minimize learning effects when using the data and because every participant was going to answer eight questions, we decided to implement four different datasets. The dataset was switched after every question using the "dataset control" which will be explained later. All four datasets had the exact same structure. They were about qualitative numeric changes of one marker of all 16 states of germany and covered the years from 2008 to 2022.

3.2 Interaction techniques

In addition to the methods already presented: *filtering* and *highlighting*, we have decided to investigate a third method. It is a modification of the highlighting method. When features are selected for comparison, they are visually highlighted using *different* colors. In classic highlighting, all selected features are highlighted in the *same* color. For the rest of this work we will distinguish highlighting using the same color and highlighting using different colors by stating the first *highlighting_1* and the second *highlighting_2*.

3.3 Layout & Function

In addition to the static data, the entire application consisted of a simple frontend that served six different non-scrollable geodashboard variants. Always two of the six variants implemented the same interaction technique. Variant 1 & 4 used *filtering*, variant 2 & 5 used *highlighting_1* and variant 3 & 6 used *highlighting_2* (see Table 3.1). On entering the page the first variant was rendered. The other variants were reachable over a tab-based navigationbar at the top of the screen. A seventh tab was implemented to learn about the application and its use. Each variant was covered with an openstreet map that was centered on Germany covering the rest of the page. All states of Germany were rendered using the borders as polygon representations, building a cloropleth-map. Every polygon used the same blue fill color. All other elements in the dashboards were placed on top of this map. On every variant, two additional views were also rendered on top of each other, taking up the left third of the screen. The graph view in the top-left corner of the screen visualized the spatial temporal data by rendering 16 lines, each representing one of the 16 states of germany. The bottom-left view visualized a two-dimensional table connecting each state with each year containing a numerical value. The "dataset control" and another button, which will be termed "comparison control" were positioned in the middle third of the screen close to the navigationbar at the top. The comparison control offered the ability to select two or three of the 16 states for comparison. After selecting the states and starting the comparison process on three of the six variants an additional graph view and an additional table view were rendered. In both views calculated differences between two states of the values of the selected states were displayed. In the graph view again as lines representing the temporal evolution of the differences and in the table view as simple numerical values for every year. Those are the already mentioned views that use *explicit encoding* as defined by Gleicher [9]. After starting the comparison process both views on the left are replaced which other views utilizing the different interaction techniques. *Replace* is one of the three common operational models visual interfaces use on parameter change [6, 17]. At any time the user could use the comparison control again to terminate the comparison process and all previously described changes are reverted. The "dataset control" on the other hand allowed the user to change the currently selected dataset via button click. After switching the dataset all views were rerendered with new labels and values.

Variant	**Interaction technique**	**Number of views**
1	filtering	2
2	highlighting_1	2
3	highlighting_2	2
4	filtering	4
5	highlighting_1	4
6	highlighting_2	4

Table 3.1: This table describes all dashboard variants visualized in the web-prototype. The number of views it rendered in "comparison mode" and the interaction technique is displayed

3.4 Multiple Coordinated Views

As already stated earlier MCV systems are highly popular and can increase user performance. Baldonado et al. argue that they can provide utility by minimizing cognitive overhead created from a single complex view. On the other hand multiple view can increase cognitive overhead (e.g. context switching) and can raise system requirements. They propose guidelines to find out when to use MCV and how to use MCV with special focus on information visualization. One use of MCV in the web-prototype was the additional rendering of two views when entering the "comparison mode". This mechanism represents the stated example from Baldonado et al. of using MCV to display aggregates of the data.

The whole prototype was designed with a focus on coordination and synchronization.

3.5 Non-/Functionl requirements

This section describes all functional and non-functional requirements on the web-prototype. They are derived from all other things described in this chapter.

Nr.	Requirement	Type
01	The app should visualize one spatial-temporal dataset with polygons on a map. Each polygon should represent a spatial entity	Functional
02	The app should allow the user to read the exact attribute value for every spatial entity over the whole time period in a separate view	Functional
03	The app should visualize the temporal evolution for the attribute values of all spatial entities at a glance in a separate view	Functional
04	The app eases the process of comparing two (or three) spatial entities through a comparison mode that can be switched on/off	Functional
05	The app should provide six different dashboard versions that differ in their interaction technique and number of rendered views for comparing spatial entities	Functional
06	The app consists of version 1 where in comparison mode only the selected entities are filtered and shown in all views. It represents a 'filtering' interaction technique	Functional
07	The app consists of version 2 where in comparison mode the selected entities are visually emphasized using one color. It represents a 'highlighting' interaction technique	Functional
08	The app consists of version 3 where in comparison mode the selected entities are visually emphasized using multiple colors. One for each entity. It represents a 'highlighting' interaction technique	Functional

Continued on next page

Continued from previous page

Nr.	**Requirement**	**Type**
09	The app consists of version 4 which is a copy of version 1 described in requirement 06. In addition, one more view that meets requirement 02 and one more view that meets requirement 03 are displayed in comparison mode. These additional views encode the subtracted values of the selected entities forming views that encode 'differences'. For every combination of the selected entities one additional data series is displayed.	Functional
10	The app consists of version 5 which is a copy of version 2 described in requirement 07. In addition, one more view that meets requirement 02 and one more view that meets requirement 03 are displayed in comparison mode. These additional views encode the subtracted values of the selected entities forming views that encode 'differences'. For every combination of the selected entities one additional data series is displayed.	Functional
11	The app consists of version 6 which is a copy of version 3 described in requirement 06. In addition, one more view that meets requirement 02 and one more view that meets requirement 03 are displayed in comparison mode. These additional views encode the subtracted values of the selected entities forming views that encode 'differences'. For every combination of the selected entities one additional data series is displayed. In line with version 6, which represents a highlighting interaction technique with individual colors, the rendered differences are also visually highlighted in different colors.	Functional
12	In all six versions in comparison mode all selected spatial entities are highlighted on the map.	Functional
13	The app enables the user to switch between four selected datasets which all have the same spatial-temporal dimensions	Functional
14	The app should be available over a website	Non-Functional
15	The app should be user-friendly and have fast loading times	Non-Functional

Continued on next page

Continued from previous page

Nr.	Requirement	Type
16	The app should use multi-coordinated views appropriately by paying attention to common guidelines to reduce cognitive overhead.	Non-Functional

Table 3.2: This table describes all functional and non-functional requirements of the web-prototype

4 User Study

4.1 Goal

The goal of the study was to find mathematical models that approximate user performance best. In our definition user performance is two-folded: answer time and accuracy. We hope to learn something about the impact the interaction technique has on these two parameters. The findings refer to the comparison of features in geo-dashboards.

4.2 Variables

We wanted to include several independent variables to get a rough idea of which factors would affect user performance and which would not. We also had to make sure that we gather enough observations per independent variable. That is the second reason we included multiple independent variables. We included the number of views the dashboard renders, the number of spatial entities that are compared simultaneously, the interaction technique the dashboard used and the type of comparison question that was asked.

4.3 Participants

4.4 Design

4.5 Procedure & Apparatus

5 Mathematical models

Because our dataset is limited, we have decided to use linear regression to try to derive a function that explains how a possible connection between user performance and our independent variables looks. Machine learning requires much more data to be effective.

First we want to look at the user performance in terms of their time used to answer the question (in the following abbreviated with 'answer time'). Because we have such limited data available and because it simplifies the analization process we decided to include all datapoints, even if the answer was not correct. Inspired by Fitts' Law, which is often used in HCI (human computer interaction) we focus on one possible aspect of answer time. An index that describes the level of difficulty of the task to which the user has given an answer. We now want to look how all our independent variables influenced the DifficultyIndex.

5.1 Assumptions

One of the rationales of MCV is that more views on the same data, help with data exploration (if applied correctly). Because comparison is a type of data exploration we can assume that more views also help increase user performance in tasks that involve comparison. We can derive:

1. A higher number of views on the same data will reduce the DifficultyIndex.

The suitability of views change across different contexts. One context for example is what task is tried to be solved using the view. In our experiment, we used a total of four different task types, which are represented by the four question types. Half of the questions ask for comparison of numerical(*attribute in space*) information and the other half asks for comparison of temporal(*space in time*) information. Lohse et al. classify visual representations(views) depending on the type of information conveyed. They classified 11 different view types and empircally derived likert scores(1-9) for every type of information [15]. Among other things they classified *time charts* and *tables* when dealing with *temporal* or *numerical* information. We use these empircal likert scores to quantify the quality of the views used when answering a specific question. Since each interaction technique uses at least one table and one time chart and we cannot predict which view the user will use, we need to calculate the mean likert score of both views for a given question type. In the experiment the question types have a second dimension. Questions could either be of type *identify* or *measure*. In our mathematical model we do not distinguish between these two types because we lack research that would help quantify view quality in those regards. We can conclude with the following assumption:

2. A higher combined encoding quality of the views in relation to the type of question will reduce the DifficultyIndex

3. A higher number of distractors will increase the DifficultyIndex. Distractors are elements in the user interface that impede the user to find the answer

$F_{difficulty} = \frac{1}{n_{views}} + \frac{n_{targets}}{1} + Difficulty_{question-type} + Difficulty_{interaction-technique}$

6 Evaluation

7 Discussion

8 Conclusion

Bibliography

[1] Natalia Andrienko, Gennady Andrienko, and Peter Gatalsky. Exploratory spatio-temporal visualization: an analytical review. *Journal of Visual Languages & Computing*, 14(6):503–541, 2003.

[2] Richard A. Becker and William S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[3] Matthew Brehmer and Tamara Munzner. A multi-level typology of abstract visualization tasks. *IEEE transactions on visualization and computer graphics*, 19(12):2376–2385, 2013.

[4] Andreas Buja, Dianne Cook, and Deborah F. Swayne. Interactive high-dimensional data visualization. *Journal of Computational and Graphical Statistics*, 5(1):78–99, 1996.

[5] D. B. Carr, R. J. Littlefield, W. L. Nicholson, and J. S. Littlefield. Scatterplot matrix techniques for large n. *Journal of the American Statistical Association*, 82(398):424–436, 1987.

[6] Maria Francesca Costabile, editor. *AVI04: International Conference on Advanced Visual Interfaces*. Association for Computing Machinery, New York, United States, 2004.

[7] Jeremy W. Crampton. Interactivity types in geographic visualization. *Cartography and Geographic Information Science*, 29(2):85–89, 2002.

[8] Bundesamt für Kartographie und Geodäsie. Bundesamt für kartographie und geodäsie. `https://gdz.bkg.bund.de/index.php/default/digitale-geodaten/verwaltungsgebiete.html`, 2023. [Online; accessed 15-10-2023].

[9] Michael Gleicher. Considerations for visualizing comparison. *IEEE transactions on visualization and computer graphics*, 24(1):413–423, 2018.

[10] Viktor Gorte and Auriol Degbelo. Choriented maps: Visualizing sdg data on mobile devices. *The Cartographic Journal*, 59(1):35–54, 2022.

[11] Ken Hinckley, Robert J. K. Jacob, and Colin Ware. Input/output devices and interaction techniques. In Allen B. Tucker, editor, *Computer Science Handbook*. Chapman and Hall/CRC, New York, 2004.

[12] Daniel A. Keim. Chapter 2 - information visualization: Scope, techniques and opportunities for geovisualization. In Jason Dykes, Menno-Jan Kraak, and Alan M. MacEachren, editors, *Exploring Geovisualization*, pages 21–52. 2005.

[13] Michael Lawrence, Eun Kyung Lee, Dianne Cook, Heike Hofmann, and Eve Syrkin Wurtele. explorase: Exploratory data analysis of systems biology data. In *Fourth International Conference on Coordinated & Multiple Views in Exploratory Visualization (CMV'06)*, pages 14–20, 2006.

[14] María-Jesús Lobo, Emmanuel Pietriga, and Caroline Appert. An evaluation of interactive map comparison techniques. In Bo Begole and Jinwoo Kim, editors, *CHI15: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3573–3582, Seoul Republic of Korea, 2015. Association for Computing Machinery, New York, United States.

[15] Gerald L. Lohse, Kevin Biolsi, Neff Walker, and Henry H. Rueter. A classification of visual representations. *Communications of the ACM*, 37(12):36–49, 1994.

[16] Chris North and B Schneiderman. A taxonomy of multiple window coordination: Dept. technical report #cs-tr-3854. `https://api.semanticscholar.org/CorpusID:17397701`, 1998. [Online; accessed 15-10-2023].

[17] Jonathan C. Roberts. Coordinated multiple views for exploratory geovisualization. In Martin Dodge, Mary McDerby, and Martin Turner, editors, *Geographic Visualization*, pages 25–48. Wiley, 2008.

[18] Jonathan C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV 2007)*, pages 61–71. IEEE, 7/2/2007 - 7/2/2007.

[19] Robert E. Roth. An empirically-derived taxonomy of interaction primitives for interactive cartography and geovisualization. *IEEE transactions on visualization and computer graphics*, 19(12):2356–2365, 2013.

[20] Alper Sarikaya, Michael Correll, Lyn Bartram, Melanie Tory, and Danyel Fisher. What do we talk about when we talk about dashboards? *IEEE transactions on visualization and computer graphics*, 2018.

[21] Bradley van Tonder and Janet Wesson. Intellitilt: An enhanced tilt interaction technique for mobile map-based applications. In Pedro Campos, Nicholas Graham, Joaquim Jorge, Philippe Palanque, and Marco Winckler, editors, *Human-Computer Interaction – INTERACT 2011*, volume 6947, pages 505–523. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[22] M. O. Ward. Xmdvtool: integrating multiple methods for visualizing multivariate data. pages 326–333.

[23] S. Wehrend and C. Lewis. A problem-oriented classification of visualization techniques. In Arie Kaufman, editor, *Proceedings of the First IEEE Conference on Visualization: Visualization '90*, pages 139–143, 1990.

[24] Ogan M. Yigitbasioglu and Oana Velcu. A review of dashboards in performance management: Implications for design and research. *International Journal of Accounting Information Systems*, 13(1):41–59, 2012.