Lin Shao, B.Sc, M.Sc

Interactive Visual Analysis and Guidance Methods for Discovering Patterns in High-Dimensional Data

Doctoral Thesis

to achieve the university degree of

Doctor of Technical Sciences

submitted to

Graz University of Technology

Supervisor

Univ.-Prof. Dr. Tobias Schreck
Institute of Computer Graphics and Knowledge Visualisation

Prof. Dr. Daniel A. Keim
University of Konstanz

Graz, December 2020

ACKNOWLEDGMENTS

I would like to thank my primary supervisor and mentor Tobias Schreck for making this work possible. He encouraged me to start my PhD and guided me through all phases of my time as a doctoral student. He always provided me with valuable feedback and inspired me with his professional and creative abilities. It was a pleasure to be a PhD student under his guidance and I am thankful for his support in my research.

I am also grateful to my second supervisor Daniel Keim for reviewing this work and supporting me during my time at the Data Analysis and Visualization Group (DBVIS) at the University of Konstanz. I want to particularly thank Michael Behrisch who supported me during my studies and my first years as a junior PhD student.

Special thanks go to all my former colleagues at the DBVIS group and recent colleagues at the institute of Computer Graphics and Knowledge Visualisation (CGV) at Graz University of Technology, who supported my research. I would further like to thank my external collaborators for their contributions to the research that has become part of this thesis: Prof. Keith Andrews, Dirk J. Lehmann, and Peter Bak.

Finally, I want to thank the persons that contributed to this thesis in a very different manner. My warmest thanks to my parents Wan-Tung and Chao-Ti, my sisters Litz and Axiao, and my girlfriend and muse Nina for their never-ending support during the past years.

–Thank you.

Abstract

In this day and age, data is increasingly important and fast-growing. It will be collected in all different areas, whether in research, industry or social life with the intent to gain knowledge and insights from it. To explore these large collections of data, information visualization techniques are often used, which map the data into a visual representation for an easier detection of patterns and hidden insights. However, there still exist well-known problems, like the *curse of dimensionality* for high-dimensional data, where merely visualizing the data is not sufficient enough. Another challenge is to find the right means, i.e., visualization type, parameter settings, exploration tools, to find the valuable information in the data. To this end, the integration of visual analytics approaches can be used to support the user in finding interesting patterns or to guide them through the analytical process.

In this doctoral thesis, I will discuss important aspects for searching and analyzing patterns in high-dimensional data, and present novel approaches that focus on current limitations. This especially includes the development, implementation and use of visual search techniques, which help the user to discover interesting patterns in complex data. For example, I will investigate sketch-based search methods to find patterns in several data types, e.g., bivariate data and trajectory data, and that allow a more intuitive way to express complex search queries. One central research objective, amongst others, is to investigate approaches for analyzing local patterns and create interactive exploration tools to make comparisons of local patterns more efficient. Interactive lens techniques are efficient exploration tools for analyzing local areas of interest, which can be used to interactively select a portion of data on which the analysis is performed. I will discuss the use of interactive lens techniques for local patterns in various data structures and show how they open up new opportunities for local pattern analysis. Moreover, I present visual guidance concepts that support the user in various analytical tasks, such as query formulation, data selection and pattern exploration. I will consider novel sensor technologies like eye tracking for the guidance process and utilize gaze information for pattern recommendation in large data sets. Finally, this thesis discusses the benefits and challenges of the proposed methods and outlines future directions.

AUTHOR'S DECLARATION

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

____________________ Date

____________________ Signature

"Everything has beauty, but not everyone sees it." (Confucius)

Table of Contents

Chapter 1

Introduction

Contents

In this doctoral thesis, I will present novel visual search modalities, interactive exploration techniques and guidance concepts for visual analytics systems to support users in discovering interesting patterns. Most metrics for discovering patterns are global in nature, and thus not applicable for the analysis of *local patterns*. One primary objective of this work is to adapt existing methods for the analysis of local patterns in order to detect important patterns for different subsets of data points. This chapter introduces my research interests in relation to local pattern analysis for point-based visualization techniques. I will point out the primary research questions, challenges and goals that will be addressed in this Ph.D. thesis. Furthermore, I will present a structured approach to tackle these issues and summarize my scientific contributions.

1.1 Motivation

In the era of Big Data, we collect and process vast amounts of data in our everyday life. The development of novel information technologies, such as wearable devices, mobile sensors or traditional measurement instruments, allow us to record a mass collection of data. These huge amounts of data are often stored and archived in so called data repositories, which make the data accessible to everyone. Among others, the research community contributes a lot from these open data sources, since it opens up application areas and new perspectives for collaborating with other researchers. Thus, researchers over the whole world can work individually on the same data basis and publish their results for future work. However, the problem of finding valuable information –the beauty– in these large collections of data is as difficult as finding a needle in the haystack. At this point, information visualization may help representing data in a visual and meaningful way so that users can better understand it. Due to rapid growth of open data repositories, the content of data sets also became step by step more complex and heterogeneous. Data sets may get expanded by other researchers, using additional sensor devices or by merging external data sources. It is common practice that data sets comprise hundreds of dimensions in a mix of different data types, e.g., numerical, categorical, hierarchical or textual data types. For instance, a scatter plot is a simple and well-known example for bivariate data visualization that provides an overview of the data and shows relevant information, such as trends, correlation and clusters, from abstract data. Besides scatter plots there are plenty of other visualization techniques, which are created to reveal the hidden information in particular data types. There exist time series visualizations, text document visualizations, network representations or high-dimensional visualization techniques, such as parallel coordinates, glyph- and matrix-representations, to name only a few. Consequently, data analysts have to choose the best suitable visualization for the given data set.

Further steps to improve the analysis process could be the integration of *Visual Analytics* approaches and interactive interfaces in information visualization systems. Visual analytics combines interactive visual representations with underlying analytical processes (e.g., statistical procedures, data mining techniques) such that high-level tasks can be effectively performed (e.g., decision-making processes, similarity searches of patterns, or visual guidance for data exploration). In addition, interactive exploration methods, such as focus+context or brushing and linking functions, can be integrated into visual analytics systems, which allow users to draw insights even more efficient and effective. The goal of visual analytics is to visually represent the data, allowing the human to directly interact with the data, to gain insight, to draw conclusions, and finally make better decisions [107].

However, there still exists a number of challenges to find interesting patterns by only exploring large collections of data or by using search methods. A good example for this is the analysis of *local patterns*. Prior research has shown that multivariate data sets may contain locally valuable information that require specialized tools to discover them [72, 73, 222]. In scatter plots, a pattern is generally defined as a set of points (i.e., from two of the n dimensions) that contain a meaningful structure or appears repetitively in the data. In my view, a global pattern comprises all objects of a particular view, while a local pattern only considers a subset of the data of a view. For instance, in a scatter plot, a global pattern may be a correlation or a mathematical function that describes the binary relation of the two dimensions, and a local pattern may be a cluster that represents the relation for a portion of the data. This distinction of patterns can be

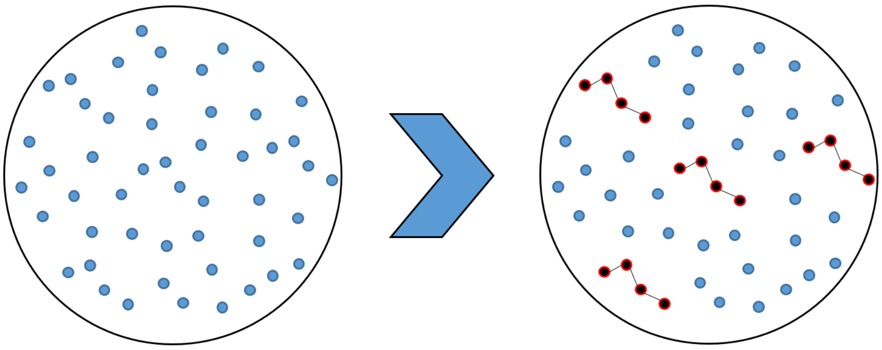


Figure 1.1: Illustration of local pattern analysis on a point based visualization. On the left side a view is shown that visualizes a data set by using a standard 2D mapping. Local patterns are difficult to spot. By using local pattern analysis tools (right side), similar patterns can be found within one view.

observed in various visualization techniques, such as time series, networks, trajectories and many other high-dimensional data visualizations. All these visualization techniques can be considered as a whole or in parts when it comes to pattern analysis. Figure 1.1 exemplifies the notion of local patterns by using a scatter plot like representation. The goal is to discover locally similar structures within one view or across multiple views by applying local pattern analysis techniques. This example shows how local patterns may get lost in larger overview visualization and how important visual encodings are to reveal them. In time series, local patterns would be periods that contain similar peaks and valleys, networks may consist of subnetworks with similar linked nodes, and trajectory data sets may contain sections with similar structures. However, many state-of-the-art techniques focus on analyzing data patterns at a global level and cannot be used for analyzing local structures in the data. Many methods do not consider additional preprocessing steps like data segmentation and feature comparison of multiple combination of smaller units, and thus, fail to automatically detect such local patterns.

In this sense, my focus of interest is to conduct further research in this field, and investigate new processes to detect, extract and analyze local properties in visualizations. In particular, this thesis focuses on visual search, interactive exploration, analytical guidance and discuss different application domains.

1.2 Goal of this Thesis

The general objective of visual analytics is the creation of tools and techniques that help people to gain knowledge from large and complex data sets. A visual analytics process is described as a tight coupling of automated and visual analysis through interaction between data, visualization, models and the user. The model by Keim et al. [108] in Figure 1.2 illustrates this process very well by showing the different stages and the respective user interactions. They describe the first

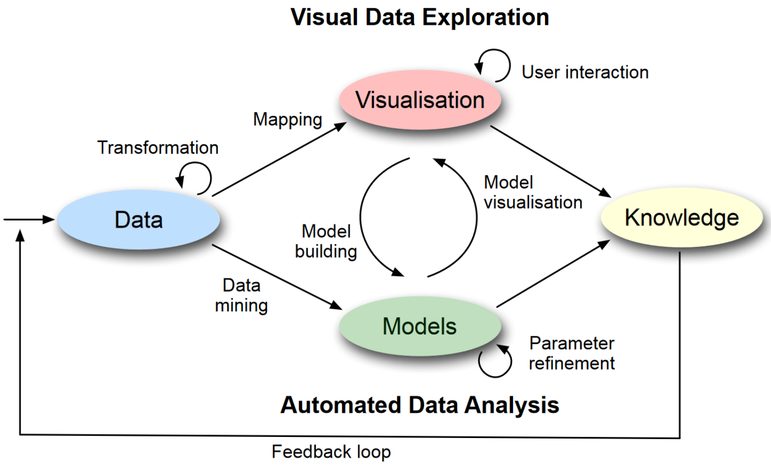


Figure 1.2: Visual analytics pipeline by Keim et al. [108]. In this model knowledge generation is described as a process through user interaction between data, visualizations and models.

step (*Data*) as a data preprocessing and transformation step to derive different representations of the data, which includes typical tasks like data cleaning, normalization, grouping and integration of heterogeneous data sources. After the transformation, analysts can start the coupled analysis by using automated data analysis (*Models*) or visual data exploration (*Visualisation*). For an automated analysis, data mining methods, such as classification, clustering or association rule mining, can be applied to generate models that are then used for visualization. Conversely, analysis may start exploring the data by using visual mappings to represent the data by a visualization, e.g., scatter plot or parallel coordinate plot, and then interact with the automatic methods by modifying parameters or selections. Key approach of visual analytics process is alternating between visual and automatic methods to gain knowledge.

In recent years, a lot of research has been done to support visual data exploration as well as automated data analysis. What is lacking, however, are tailored approaches for analyzing *local patterns*. One of my areas of interest is to foster analysis of local patterns in visual analytics processes. According to the visual analytics model, local patterns can be analyzed on the *Visualisation* and *Model* stage. Basically, it is also possible to extract local patterns from raw data (*Data* stage) by using automatic data segmentation approaches. The segmentation approaches on the raw data typically use low-level segmentation features like time stamps or other meta data information, and thus will not be considered in this work. On the visualization side, common interaction techniques like brushing and linking, color coding, details on demand, zooming and panning can be used to discover local patterns in data visualizations. However, these techniques are not specially designed for comparing local patterns and are limited when it comes to complex and large data sets. On the model side, there exist pattern recognition and feature extraction techniques, which can be used on the model side to discover interesting patterns in the data. Usually, these tools are designed for global patterns and require custom adjustment for the application of local structures. In particular, we need visual analytics tools that incorporate both visual data exploration as well as automated data analysis, like, for example, practical interfaces to visually define patterns or areas of interest that interact with analytical models for evaluation.

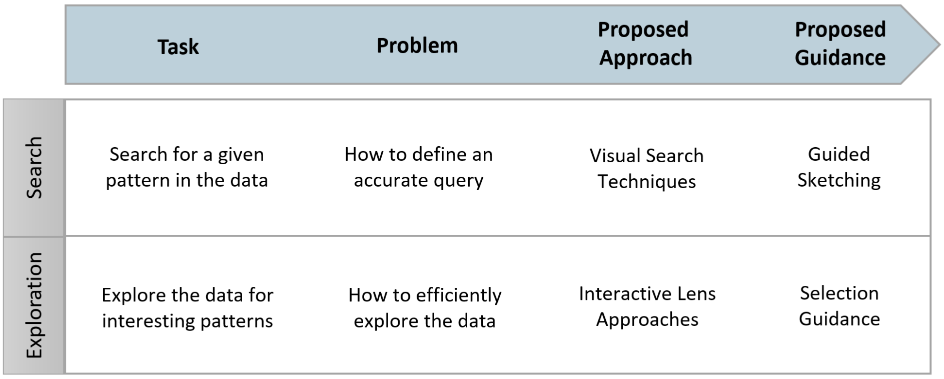
It is also important to note that the analysis methods vary based on the initial situation of the user. In general, there are two different approaches for local pattern analysis:

(i) Search for specific patterns in the data by using pattern retrieval algorithms;
(ii) Explore the data for interesting patterns by using interactive exploration tools.

If analysts are aware of specific phenomena in the data and want to find resulting patterns, a search interface may be an efficient tool for this purpose. Hence, analysts can specify the requirements as query and retrieve all relevant views that match to the query. On the other hand, if analysts are not searching for a specific pattern and the task is to find phenomena in the data, search interfaces may not be the best tool since query formulation might be challenging. In this case, interactive exploration tools can help to explore the entire data set and find insights. One goal of this work is to research the two approaches and provide various tools that enhance both analysis processes.

Furthermore, I want to investigate guidance concepts that support local pattern analysis in both directions, for the automated analysis and visual exploration. Guidance is very important field in visual analytics as it directly contributes to the tight integration between automated analysis and visual exploration. In many applications, guidance approaches are used to automatically communicate between models and visualization. For instance, it can be used to monitor user inputs on the exploration side and control parameter refinement on the model side to suggest potentially interesting views. Another goal of this thesis is to integrate novel guidance functions into visual search and interactive exploration systems that especially focus on local pattern analysis. In visual search applications, a guidance function may assist users to formulate more precise queries or recommend queries that lead to new insights in the data. To support data exploration tasks, it could be used to improve user selections, recommend interesting views to the user and adjust general settings in order to maximize the outcome. Therefore, the guidance functions must be integrated as intermediate layer between *Visualisation* and *Models* of the visual analytics pipeline (Figure 1.2) to learn the users' interest and create recommendations. In this context, I want to experiment with novel sensoring devices, such as eye tracking devices, to improve the process of finding user interests.

Figure 1.3 outlines the different aspects for searching and exploring local patterns, as well as my proposed solutions. As solution I propose visual search techniques, interactive lens approaches and visual guidance features. If analysts are looking for a given pattern, e.g., local trend or cluster, a textual query might be difficult to specify. Therefore, I want to investigate visual search techniques where analysts can specify a query in a visual way, like sketching a pattern. Sketching trends, correlations or a shape of a pattern can be much easier than describing it in textual form. If one wants to explore the data on a local level interactive lens techniques, which integrate



	Task	Problem	Proposed Approach	Proposed Guidance
Search	Search for a given pattern in the data	How to define an accurate query	Visual Search Techniques	Guided Sketching
Exploration	Explore the data for interesting patterns	How to efficiently explore the data	Interactive Lens Approaches	Selection Guidance

Figure 1.3: Detailed comparison between pattern exploration and search. The pipeline summarizes major differences in tasks, problems and my proposed approaches to support the analysis of local patterns.

additional model based information can help to gain insights. Lens selections can be positioned on local areas of interest and allow a fast and interactive way to analyze the data. Difficulties may arise in practice, hence, guidance functions can help to sketch the query or suggest more suitable selections for analysis.

1.3 Research Questions and Challenges

Based on the defined goals in the previous section, I elaborated several research questions, which will be described in detail in the following.

Research Questions

1. How can we find local patterns in high-dimensional data sets?
    (a) How can we automatically extract local patterns in data sets?
    (b) Which high-level features can be used to characterize local patterns?
2. How can we help the user to efficiently explore and search for local patterns?
    (a) How can we measure the similarity of local patterns?
    (b) How can we abstract high-level information from local patterns?
3. How can we support the user in finding interesting local patterns?
    (a) How can we identify the users interests?
    (b) How can we derive interestingness for local patterns?
    (c) How can we make user recommendations?

The first research question also refers to the overall question for the definition of a local pattern.

What is actually a local pattern?

To find local patterns in high-dimensional data it is important to clarify what patterns are in general and what the difference between a global and local pattern is. Currently, many visual analytics techniques focus on exploring patterns within the data at a global level, for example by considering the distribution of all data points in a scatter plot. Bertini et al. defined a (*global*) pattern as *"recurring events or objects that repeat in a predictable manner. The most basic patterns are based on repetition and periodicity"* [22]. In scatter plot analysis cluster and correlation patterns are often recurring events in many data sets. Visual analytics systems are widely used in the literature to automatically detect such patterns or help users to identify them based on visual perception and cognition models. However, it has emerged that patterns may also occur in local data properties, e.g., a pattern that comprises a subset of points in a scatter plot, which are not particularly considered in the analysis [171]. In market analysis local patterns may emerge from different target groups that differ in their characteristics like age, sex, location, income, and household type. A pattern of one target group may also repeat in views of different dimensions in a predictable manner. To this end, one could say that a local pattern is a recurring event by subset of objects that repeat in a predictable manner.

To perform automated data analysis on local patterns, first the data have to be partitioned into several unconnected subsets. This can be done by clustering algorithms or machine learning models that split the data set based on their local properties. The main factors of these methods that influence the output are parameter settings for clustering and a good test data set to train machine learning models. Furthermore, users often require domain knowledge to label the test data sets or to configure the clustering approach. More flexible approaches are needed that do not rely on user inputs. I therefore investigate the performance of data segmentation approaches for point-based visualization techniques and conducted an experimental study to evaluate cluster results for segmentation (c.f. Section 3.3).

The second set of research questions addresses the problem of retrieving and exploring potentially interesting patterns. As discussed earlier, there are two different ways how users can efficiently find local patterns in the data –either by applying specific search query or by exploring the data interactively.

In the first case, users must be aware of available patterns and be able to define a concise query for search. Depending on the underlying data structure, e.g., hierarchical, geospatial or numerical-data, a query definition for local properties could already be challenging. Visual search techniques like sketch-based or example-based search, may help to formulate queries simpler and more efficiently. For example, sketch-based search approaches allow users to directly sketch patterns of interest, e.g., by defining the shape, size or orientation, for search. The challenge here is to measure the similarity between the user sketch and the data pattern in the view space. This could either be done by transforming the user sketch into a normalized representation or by using feature extraction techniques that enable a comparison of high-level information on local properties. Further challenges include consideration of small deviations, e.g., scaling, rotation and translation, between user sketch and target pattern. Both approaches have their advantages and disadvantages. More research is needed to clarify these questions and to provide efficient visual search techniques for local pattern retrieval.

In the latter case, advanced visual analytics system including interactive exploration possibilities are needed to reveal locally interesting patterns, which could be hidden in overview visualizations. Therefore, flexible and responsive exploration tools are required, which abstract local features in real-time and display additional information for given data subsets. Figure 1.1 illustrates such an example where highlighting is used to show similar local patterns for a selected region. One main challenge is to identify a good partitioning for a variable number of data points since, the data subset can be changed multiple times during an exploration process. It also requires good data abstractions and feature extraction methods to make patterns comparable and to identify similar patterns in the data. Then, these abstracted information of local patterns can be visually encoded and shown in detail-on-demand views. Mathematical models, skeleton-based representation or many other models can be considered as data abstraction to compare local properties. Furthermore, when considering the similarity of local patterns in a data set recurring patterns (motifs) can be taken into account, too. In this thesis, I will investigate several exploration methods and develop visual analytics systems, which allow detail-on-demand information based on local properties.

The last set of research questions focuses more on guidance functions for visual analytics systems that actively assist users during search and exploration tasks. The exploration for relevant and meaningful information in large data sets is a challenging task, since the number of possible interesting views grows with the amount of data. Usually, not all views from a possibly large view space, are potentially relevant to a given analysis task or user. The same applies for retrieving specific patterns in the data. When the view space increases, the number of matching results may also increase for a given query. Thus, a precise query is needed in oder to ensure the quality of retrieval results and receive good precision and recall scores.

To foster exploration, quality metrics and recommender systems are often used to help analysts during exploration processes to find interesting views. However, current quality metrics are based on heuristics and do not take the users' interest into account, whereas recommender systems need specific user interactions to express their feedback. What we need in addition are visual analytics systems, which derive individual user interests by tracking interactions to support the user in their particular task. Here, the question arises of which user interactions should be focused on and how can we train a system to learn the users' interest. Assuming that users' interest is known then another second question arises of how to derive interestingness scores and rank relevant views for recommendations. There are many possibilities, e.g., by using relevance feedback, annotations or monitoring systems for user interactions, to archive this goal. To answer these questions, I will investigate several approaches including interest measures based on local motifs and eye-tracking based information to recommend interesting views to the

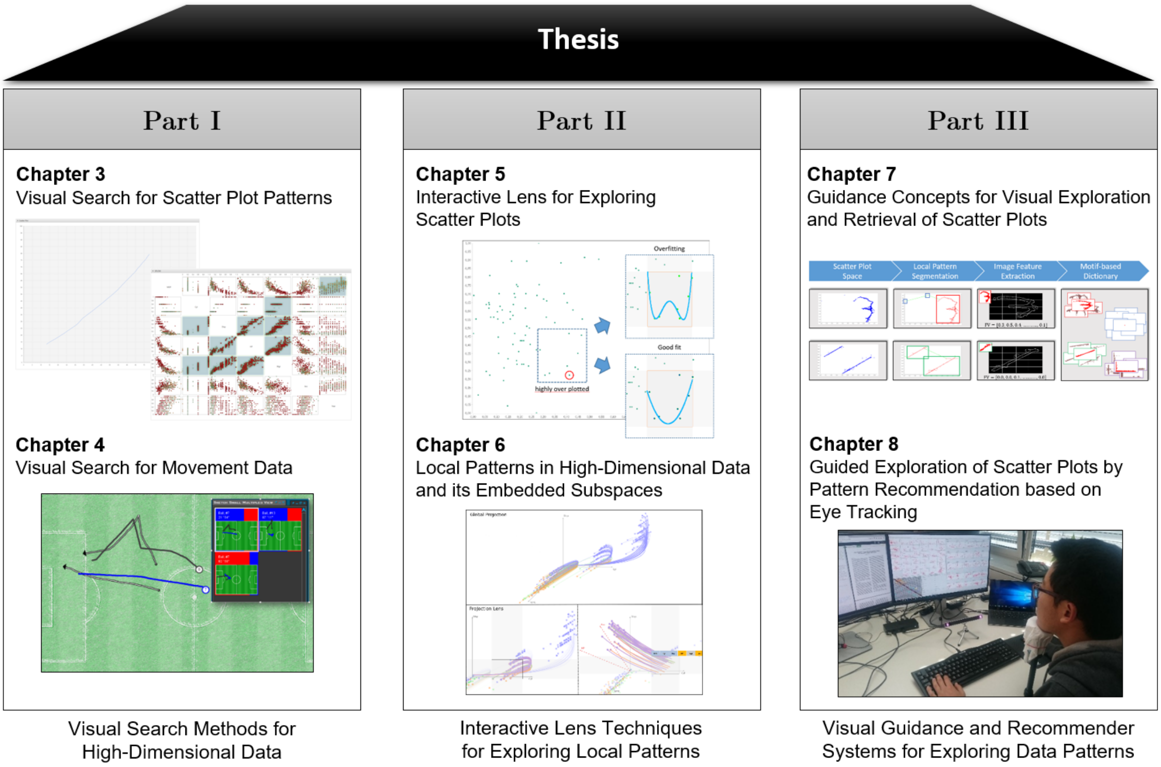

Thesis
Part I
Chapter 3
Visual Search for Scatter Plot Patterns
Chapter 4
Visual Search for Movement Data
Visual Search Methods for High-Dimensional Data
Part II
Chapter 5
Interactive Lens for Exploring Scatter Plots
Overfitting
Good fit
highly over plotted
Chapter 6
Local Patterns in High-Dimensional Data and its Embedded Subspaces
Interactive Lens Techniques for Exploring Local Patterns
Part III
Chapter 7
Guidance Concepts for Visual Exploration and Retrieval of Scatter Plots
Scatter Plot Space
Local Pattern Segmentation
Image Feature Extraction
Motif-based Dictionary
Chapter 8
Guided Exploration of Scatter Plots by Pattern Recommendation based on Eye Tracking
Visual Guidance and Recommender Systems for Exploring Data Patterns

Figure 1.4: This thesis is structured into three main parts that contain summarized results of my works. Each part is further subdivided into chapters that discuss different application areas. The first part includes works in the field of visual search for high-dimensional data, in particular for the retrieval of local patterns in scatter plot and movement data. The following part considers interactive lens techniques to support the analysis of local patterns. Finally, Part III covers visual guidance concepts and recommender systems for exploring local patterns.

user. Finally, the last important question is to identify the best way to present the suggestions to the analyst. For instance, analysts could be automatically navigated through the most interesting views or can be notified about patterns/views which are not recognized. To support the visual search process, I will research on guidance functions to support users in drawing scatter plot patterns.

1.4 Structure of the Thesis

The thesis is organized into nine chapters including introduction, background, conclusion and six main chapters. The main contribution of this thesis are incorporated into the six main chapters, which is further divided into three parts based on the research domains presented in the previous section. Figure 1.4 illustrates the content of the six main chapters (Chapter 3–8).

The following Chapter 2 discusses necessary background information for the thesis and points out state-of-the-art techniques for similarity search, interactive exploration and visual guidance.

It also clarifies the notion of local and global patterns, and indicates the importance of local pattern analysis.

Chapter 3 presents visual search approaches to identify local patterns in high-dimensional data by using scatter plot representations. First, an automated segmentation approach is presented to disassemble scatter plots into their individual components –local patterns. On this basis, feature extraction approaches are presented that can be used for comparison and retrieval. To perform advanced searches, two visual search interfaces are introduced that allow analysts to visually define queries of interest. Therefore, analysts may sketch a pattern of interest (Query-by-Sketch) or assemble a query by using the local patterns (Query-by-Example).

Chapter 4 considers visual search approaches for other application domains and utilizes soccer analytics as alternative application. Instead of scatter plot patterns, trajectory data of player movements and ball positions are used as retrieval criteria to find interesting game situations. It presents two visual search approaches where analysts can sketch movements directly on the soccer pitch as query or define start and end areas to retrieval all trajectories that run through the selected sections.

Chapter 5 focuses on interactive exploration tools for discovering and comparing local patterns in large scatter plot data. It introduces a system for visual-interactive regression analysis supporting both global and local regression modeling. Therefore, an interactive lens approach is used, allowing analysts to interactively select a portion of data, on which regression analysis is run in interactive time.

Chapter 6 extends the interactive lens approach from the previous chapter for a projection-based data exploration. By this extension the interactive lens can be used for analyzing local patterns that result from subspace changes in a 2D projection view. Therefore, an analyst can explore data subspaces by iteratively adding dimensions to be considered to a portion of data. Rather than creating a single static projection, the presented technique creates a continuum of projections for each added dimension. The space of projections is visualized by displaying trajectories, which can reveal local structures (projection paths) in high-dimensional data spaces.

Chapter 7 presents visual guidance approaches that support visual search and exploration processes for scatter plot data. The guidance methods are based on the approaches and techniques presented in the previous sections. To support the sketch based search process a shadow-drawing approach provides suggestions for possibly relevant patterns, while query drawing takes place. It also introduces an interest measure inspired by the well-known tf×idf-approach from information retrieval to suggest interesting views for exploration. Both local and global quality measures are computed based on certain frequency properties of local motifs in one view.

Chapter 8 introduces a recommender system that supports the exploration of scatter plot matrices in real time by using eye tracking information. The system is based on a set of precomputed pattern classes and utilizes the information of seen patterns together with plot similarity to suggest unexplored scatter plots. A classifier learns the visual characteristics from the previous seen plots and uses visual search methods to recommend the most visually dissimilar scatter plots for further exploration.

Chapter 9 provides a summary of the described approaches and technical contributions with a discussion of limitations and open research questions.

1.5 Scientific Contributions and Publications

During my doctoral studies, I collaborated with different research groups and worked on several publications that connect to my research topic. Most of these projects have been performed in close collaboration with the University of Konstanz (former department) and Graz University of Technology (current department). The publications cover, amongst others, applications in the area of visual analytics, sports analytics and human sensing. The following list outlines all publications that contributed to this thesis as well as the contributions of all authors with these publications.

- **Guided Sketching for Visual Search and Exploration in Large Scatter Plot Spaces**, L. Shao, M. Behrisch, T. Schreck, T. von Landesberger, M. Scherer, S. Bremm and D. A. Keim. *In EuroVis Workshop on Visual Analytics (M. Pohl and J. Roberts, eds.), The Eurographics Association, 2014.*

  This paper grew out of my Master's thesis on visual search for scatter plots at the University of Konstanz and was further developed during my PhD. The initial idea of this work was elaborated by T. Schreck, M. Behrisch and myself. After the initial discussion, I have taken the leader of this work and took primary responsibility for this publication. I implemented the system, prepared use cases and wrote the major parts of the text. M. Behrisch supported me in writing the text. T. Schreck and T. von Landesberger revised the paper draft and helped me to improve the text. M. Scherer, S. Bremm and D. A. Keim reviewed the paper and gave inspiring comments.
  Contributions: Visual Search (C1); Guidance (C3)

- **Guiding the exploration of scatter plot data using motif-based interest measures**, L. Shao, T. Schleicher, M. Behrisch, T. Schreck, I. Sipiran and D. A. Keim. *Journal of Visual Languages & Computing, vol. 36, pp. 1 – 12, 2016.*
  Best Paper Award IEEE International Symposium on Big Data Visual Analytics (BDVA), 2015.

  This work is a result of a close collaboration between T. Schleicher and myself. I supervised T. Schleicher's Master project and thesis during my time at the University of Konstanz. In this collaboration we worked together on scatter plot segmentation approaches and cluster quality assessments, which was reused for this publication. For this paper, I developed the major ideas, implemented the local interest measure and was responsible for leading the

project. T. Schleicher implemented the note taking interface for pattern exploration and helped me in writing the technical section. T. Schreck and M. Behrisch guided the project and supported me in writing the paper. I. Sipiran and D. A. Keim reviewed and revised the paper and commented on paper drafts.
Contributions: Visual Search (C1); Interactive Exploration (C2); Guidance (C3)

- **Visual-Interactive Search for Soccer Trajectories to Identify Interesting Game Situations**, L. Shao, D. Sacha, B. Neldner, M. Stein and T. Schreck. *Conference on Visualization and Data Analysis, Electronic Imaging, vol. 2016, no. 1, 2016.*
  Honorable Mention »Impact on Business« Award of Fraunhofer IGD and TU Darmstadt GRIS, 2017.

  This research has been initiated in the context of the soccer analysis team within the research group at the University of Konstanz. The paper is an outcome of a close collaboration between B. Neldner and myself (I supervised his Bachelor project and thesis). B. Neldner was responsible for the major implementation efforts and I was responsible for leading the project, the major ideas, and writing the paper. The soccer analysis team including D. Sacha supported us to integrate the visual search tool into the soccer analytics system [97]. I evaluated the system by use cases and wrote the major parts of the text. D. Sacha supported me in writing Section 3. M. Stein was helping me with the related work and provided feedback during the discussions. T. Schreck guided the project and commented on paper drafts.
  Contributions: Visual Search (C1); Interactive Exploration (C2)

- **Interactive Regression Lens for Exploring Scatter Plots**, L. Shao, A. Mahajan, T. Schreck and D. J. Lehmann. *In Eurographics Conference on Visualization (EuroVis), vol. 36, pp. 157–166, Computer Graphics Forum, The Eurographics Association and John Wiley & Sons Ltd., 6 2017.*

  Together with T. Schreck, we developed the concept for exploring scatter plot patterns by using local regression models. I was leading the project and responsible for organizing meetings, discussions and structuring the work. A. Mahajan implemented the first prototype of the system, which I further developed and finalized for this publication. I prepared the use cases and wrote the major parts of the text. T. Schreck and D. J. Lehmann guided the project and provided continuous feedback during the project. T. Schreck supported me in writing the discussion section and D. J. Lehmann was responsible for the mathematical background in Section 3. All authors commented on paper drafts and gave valuable feedback to improve the text.
  Contributions: Interactive Exploration (C2); Guidance (C3)

- **Discovering Local Patterns in High-Dimensional Data and its Embedded Subspaces**, L. Shao, S. Kloiber, M. Chegini, K. Andrews, T. Schreck. and D. J. Lehmann.

Submitted to Special Issue on Neural Computing & Applications Journal, 2020

This paper is a follow up our paper on Interactive Regression Lens for Exploring Scatter Plots [171]. Together with T. Schreck. and D. J. Lehmann, we developed the approach for integrated projection paths as extension for the current system. I implemented the system, prepared use cases and was primary responsible for this publication. S. Kloiber supported me in writing Section 3 and M. Chegini contributed to the discussion section. All parts of the paper were revised several times by me and K. Andrews. T. Schreck and D. J. Lehmann guided the project and commented on paper drafts.
Contributions: Interactive Exploration (C2)

- **Visual Exploration of Large Scatter Plot Matrices by Pattern Recommendation Based on Eye Tracking**, L. Shao, N. Silva, E. Eggeling and T. Schreck. *In Proceedings of the 2017 ACM Workshop on Exploratory Search and Interactive Data Analytics, ESIDA '17, (New York, NY, USA), pp. 9–16, ACM, 2017.*

  This publication was the outcome of an collaboration with N. Silva from Fraunhofer Austria at Graz University of Technology. N. Silva is a researcher in the field of human sensing and data science, and was involved in the implementation of the recommendation system and eye-tracking integration. Also, N. Silva helped to conduct a qualitative user study and provided related work. I was leading the project, developed the major ideas and was responsible for most of the sections in the paper. I did the implementation for the similarity search and ranking approach. All authors were involved in discussing interesting extensions and commented on paper drafts. E. Eggeling and T. Schreck actively reviewed and revised the paper.
  Contributions: Visual Search (C1); Guidance (C3)

- **Query by Visual Words: Visual Search for Scatter Plot Visualizations**, L. Shao, T. Schleicher and T. Schreck. *In Proceedings of the Eurographics / IEEE VGTC Conference on Visualization: Posters, EuroVis '16, (Goslar Germany, Germany), pp. 41–43, Eurographics Association, 2016.*

  This poster paper provides an extension to a previous paper [173, 178] on local scatter plot exploration and retrieval. We reused the pattern segmentation approach of [178] and combined it with the visual search approach of [173] to develop a scatter plot retrieval system based on visual motifs. I was leading the project, writing all sections and incorporating valuable feedback of all co-authors. T. Schleicher implemented the interface and conducted the user experiment under my supervision. T. Schreck reviewed and revised the paper and commented on paper drafts.
  Contributions: Visual Search (C1)

I wrote the major parts of these publications and revised all the sections several times. Thus, I will reuse parts of the text in this dissertation without citation marks.

Papers/Contribution	C1	C2	C3
Guided Sketching for Visual Search and Exploration in Large Scatter Plot Spaces	•	-	•
Guiding the exploration of scatter plot data using motif-based interest measures	•	•	•
Visual-Interactive Search for Soccer Trajectories to Identify Interesting Game Situations	•	•	-
Query by Visual Words: Visual Search for Scatter Plot Visualizations	•	-	-
Interactive Regression Lens for Exploring Scatter Plots	-	•	•
Integrated Projection Paths for the Discovery of Data Patterns in Subspaces	-	•	-
Visual Exploration of Large Scatter Plot Matrices by Pattern Recommendation Based on Eye Tracking	•	-	•

Table 1.1: Overview of publications that are part of this thesis. The contributions of the publications are marked with (•) according to the defined research domain.

Table 1.1 summarizes all publications that contribute to this thesis. The main contributions relate to visual search techniques (C1), interactive exploration approaches (C2) and visual guidance & recommendations (C3), and are incorporated in Part I, Part II and Part III. Hereinafter, *"we"* refers to me and the coauthors that are introduced in the beginning of each chapter.

In addition, there are a number of related publications that I was involved in, but that only indirectly contributed to the content of this thesis:

- **mVis in the Wild: Pre-Study of an Interactive Visual Machine Learning System for Labelling**, M. Chegini, J. Bernard, L. Shao, A. Sourin, K. Andrews and T. Schreck. *In Proceeding of IEEE VIS 2019 Workshop on Evaluation of Interactive Visual Machine Learning Systems, 2019.*
- **Extending Document Exploration with Image Retrieval: Concept and First Results**, L. Shao, M. Glatz, E. Gergely, M. Müller, D. Munter, S. Papst, and T. Schreck. *In Eurographics Conference on Visualization - Poster Paper, EuroVis '18, 2018.*
- **Quality Metrics for Information Visualization**, M. Behrisch, M. Blumenschein, N. W. Kim, L. Shao, M. El-Assady, J. Fuchs, D. Seebacher, A. Diehl, U. Brandes, H. Pfister, T. Schreck, D. Weiskopf, and D. A. Keim. *Computer Graphics Forum, 37: 625-662, 2018.*
- **Interactive Visual Exploration of Local Patterns in Large Scatterplot Spaces**, M. Chegini, L. Shao, R. Gregor, D. J. Lehmann, K. Andrews and T. Schreck. Computer Graphics Forum, 37: 99-109, 2018.
- **Toward Multimodal Interaction of Scatterplot Spaces Exploration**, M. Chegini, L. Shao, K. Andrews and T. Schreck. *AVI Workshop on Multimodal Interaction for Data Visualization, 2018.*
- **Analysis and Comparison of Feature-Based Patterns in Urban Street Networks**, L. Shao, S. Mittelstädt, R. Goldblatt, I. Omer, P. Bak and T. Schreck. *In Proceedings of*

the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISIGRAPP 2016, pp. 84-95. Revised Selected Papers (1 ed., Vol. 693). Springer International, 2017.

- **Interaction Concepts for Collaborative Visual Analysis of Scatterplots on Large Vertically-Mounted High-Resolution Multi-Touch Displays**, M. Chegini, L. Shao, D. J. Lehmann, K. Andrews and T. Schreck. *In Proceedings of the 10th Forum Media Technology, 2017.*
- **Visual Exploration of Hierarchical Data Using Degree-of-Interest Controlled by Eye-Tracking**, N. Silva, E. Eggeling, T. Schreck, L. Shao, D. W. Fellner. *In Proceedings of the 9th Forum Media Technology 2016 and 2nd All Around Audio Symposium, 2016.*
- **Sense.me - Open Source Framework for the Exploration and Visualization of Eye Tracking Data**, N. Silva, L. Shao, T. Schreck, E. Eggeling and D. W. Fellner. *In Proceedings of IEEE Conference on Information Visualization: Posters, 2016.*
- **Quality Metrics Driven Approach to Visualize Multidimensional Data in Scatterplot Matrix**, M. Behrisch, L. Shao, J. Buchmüller and T. Schreck. *Eurographics Conference on Visualization (EuroVis) - Poster Paper, 2016.*
- **Feedback-Driven Interactive Exploration of Large Multidimensional Data Supported by Visual Classifier**, M. Behrisch, F. Korkmaz, L. Shao and T. Schreck. *In Proceedings of IEEE Conference on Visual Analytics Science and Technology (VAST), pp. 43-52, 2014.*

CHAPTER 2

BACKGROUND

Contents

The focus of this chapter is to give a short summary of necessary background information on the three major parts of this thesis: *Visual Search*, *Interactive Exploration* and *Visual Guidance*. The first part of this chapter starts with the basic principles of visual analytics including a brief insight on information visualization and the techniques that are applied in this work. In addition, I will give definitions of fundamental terms used throughout this thesis.

Parts of this chapter are adapted and/or taken from the text I have written and coauthored in the paper *Quality Metrics for Information Visualization* [15].

2.1 Basic Principles of Visual Analytics

Before discussing the details and definitions of my work, I want to start this chapter with a discussion about the fundamentals of Visual Analytics (VA). First of all, what is visual analytics? A general definition of visual analytics is not that easy to determine, since it has evolved from multiple processes and application areas and thus there exist multiple definitions and pipelines with slight variations. From the historical perspective, visual analytics has evolved from the information and scientific visualization communities. One of the first articles in the scientific literature –IEEE Xplore digital library– that uses the term "Visual Analytics" was published in the year 2004 by Wong and Thomas [144]. Here, Wong and Thomas described visual analytics as follows:

> *"Visual analytics is the science of analytical reasoning facilitated by interactive visual interfaces".*

For decision makers and data analysts, it is an essential task to rapidly extract relevant information from huge data collections, which may contain massive, inconsistent and messy volumes of data. This information overload problem is a well-known phenomenon of the information age, due to the progress in computational power and storage capacity that produce data at an incredible rate. The basic idea of VA is to turn the information overload into an opportunity and gain valuable information from the flood of data [109]. VA approaches visually represent the information, allowing the human to directly interact with the information in order to gain insight, to draw conclusions, and finally, make better decisions. Furthermore, people may use VA tools and techniques to synthesize information and derive insight from massive, dynamic, and often conflicting data, such that they can detect the *expected* but also discover the *unexpected* in the data [44].

Over the years, it has been evolved into a multidisciplinary field that includes several focus areas, such as analytical reasoning, visual representation and interaction, and data representation and transformation. Therefore, Keim et al. [107] refined the definition of visual analytics and included the scope of automated analysis with interactive visualization together with the visual analytics pipeline (Figure 1.2) that I described in Section 1.2. According to Keim et al.:

> *"Visual analytics combines automated analysis techniques with interactive visualisations for an effective understanding reasoning and decision making on the basis of very large and complex datasets".*

Generally speaking, visual analytics approaches combine the strengths of human and automatic data processing to generate knowledge from data. To facilitate smooth cooperation between humans and machines, visualization techniques have become the medium for analytical processes, where humans can steer the analysis according to a specific task and machines provide effective means to support the task. For instance, a suitable visualization technique can be used to display a given data set, e.g., scatter plots for bivariate data, networks and graphs for complex relationships, maps representations for geospatial information, etc., and enhanced by adequate user interactions, which allow the user to effect on visualizations (i.e., selecting or zooming) or effect on hypotheses by revealing insights. Since most data sets are complex as well as too large to be visualized straightforward, a visual analytics process comprises a continual utilization of automated analysis models. This procedure is described by the Visual Analytics Mantra by Keim et al. [109] as: "Analyze first - show the important - zoom, filter and analyze further - details on

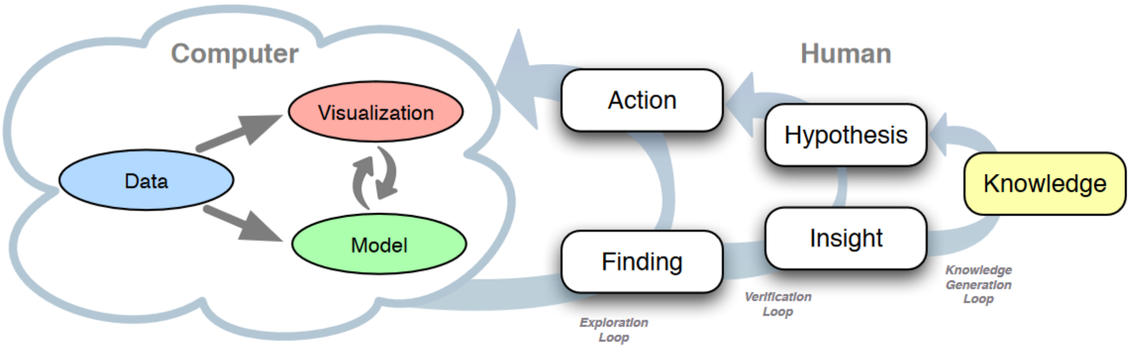


Figure 2.1: Knowledge generation model for visual analytics by Sacha et al. [158]. It is an extension of the well-known visual analytics process model of Keim et al. and integrates the human interaction processes within visual analytics systems.

demand". Further information about the wide spectrum of visual analytics including its scope, applications and challenges can be found in [17, 44, 105, 107, 109, 133, 191].

More recently, the VA pipeline by Keim et al. was then extended by Sacha et al. [158] to provide high-level descriptions of the human and computer processes within visual analytics systems (see Figure 2.1). They indicate that there is no clear separation between computer and human, and that both are required to understand the functionality and interaction between the components. Essential elements of the knowledge generation model are the exploration loop, verification loop and knowledge generation loop. The exploration loop describes how analysts interact with visual analytics systems to generate visualizations, models and analyze the data. Whereas the verification loop and knowledge generation loop are two subsequent loops that build on the exploration loop to confirm hypotheses and gain new knowledge. The actions described in the exploration loop refer to the three base modules *Data*, *Model*, *Visualization*, and consist of actions for data preparation, model building, model usage, visual mapping, visualization manipulation and model-vis mapping. Findings in this model lead to further interaction with the system or to new insights, e.g., missing values from data inspection or patterns in the visualization or model.

Furthermore, they indicate that in case of missing analysis goals, the exploration loop becomes a search for findings, which may lead to new analytical goals. Considering the search for findings, one can use various visual analytics tools to achieve this. However, they did not specify in detail how and which visual analytics tools can be used to gain insights. In my research work, I concentrate on the model-vis mappings that support the analysis of global and local patterns for high-dimensional visualization techniques and show that the *exploration loop* can be performed in several ways.

2.2 Data Visualization Techniques

As already mentioned in the previous chapter, there exists a great variety of visualization techniques to display high-dimensional data sets. Usually, analytical approaches in VA are organized according to a visualization technique (e.g., scatter plot, parallel coordinate plot, matrix, graph) or a data type (e.g., nominal, ordinal, numeric). For instance, scatter plots are particularly suitable for visualizing clusters and distributions for typically two dimensions, while parallel coordinates are better suited to show cluster relations and flows across multiple dimensions. A good overview of existing visualization techniques and its applications are given in [83, 124, 214]. Popular techniques to display high-dimensional data include point-based visualization techniques, tabular displays, parallel and radial axes plots, and pixel displays. The challenge here is to identify the best automated algorithms for a given analysis task and develop a tightly integrated solution including user interactions for an appropriate visualization. The most VA systems rely on interaction concepts such as brushing and linking, zooming, panning, and filtering. However, there exist many different categories of interaction concepts and design choices with regard to visualization technique and analysis task. For instance, Sarikaya and Gleicher [161] characterized scatter plot-specific analysis tasks and presented a survey on different design choices and interactions. A more extensive overview of existing methods for visualization and interactive visual analysis of multi-faceted data is given in the survey by Kehrer and Hauser [105].

Both, automated analysis approaches and interactive exploration techniques often use the corresponding characteristics of visual encoding, like distances between points, clutter or edge crossings, for analytical assessments. For instance, there exist a number of quality metrics that automatically judge the quality of views in matrix visualizations [14, 18, 149], parallel coordinates [10, 99] or scatter plots [131, 194, 216]. For more details, an extensive overview of quality metrics for various visualization techniques is given by Behrisch et al. [15].

The proposed approaches in this thesis focus on analysis methods for point-based visualization techniques. Point-based visualizations are a fundamental representation technique in VA and have a very wide application area. It projects records from an n-dimensional data space to an arbitrary k-dimensional display space. For instance, scatter plot visualizations are one of the most widely used point-based visualization technique and contain well-understood visual representations. Usually, it is used to display bivariate data, but can also applied for high-dimensional data via dimensionality reduction or the scatter plot matrix representation. I mainly deal with automated approaches and interaction techniques for scatter plot visualizations but also consider trajectory visualizations and high-dimensional projection views, since they have a strong relation to scatter plots. Next, I will briefly outline the visualization techniques that are considered in this thesis.

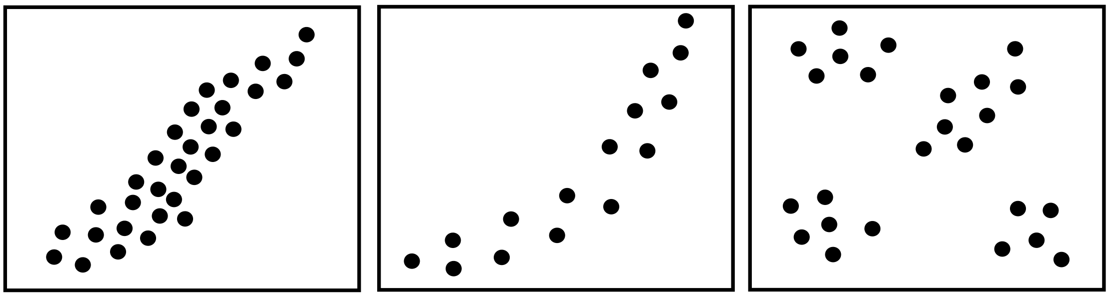


Figure 2.2: Examples of data patterns in scatter plot visualizations. Scatter plots are suitable to reveal trends, correlations and clusters for bivariate data.

Scatter Plot

One opportunity to visualize high-dimensional data is to use scatter plots and scatter plot matrices respectively. A scatter plot is a common and well-known technique, which presents the data distribution of typically two variables as x and y axis in a Cartesian coordinate view [41, 214]. The main advantage of this visual representation is that the readability of single data instances, as well as data patterns, are straightforward and easy to understand. Figure 2.2 illustrates three examples of data distributions using scatter plots. One can clearly see if two variables correlate, reveal clusters or patterns by using a scatter plot. In order to represent high-dimensional data set, this type of visualization can be applied to all pairwise combinations of dimension views in a tabular form ordered by dimensions.

Although the visualization schema is simple, there still exist challenges to improve the visual representation. To visualize clusters, patterns, and trends properly, the scaling of the two variables needs to be chosen carefully. Another well-known problem of scatter plot visualizations is to visualize large numbers of items, which often results in visual clutter. Visual clutter may obscure patterns in the data and makes it difficult for the user to find relationships among the dimensions. A challenge is to reduce the number of displayed elements but maintain the overall information at the same time. In recent years, several clutter reduction techniques have been developed to reduce the number of elements in a plot, which include sampling, filtering, clustering, and distortion techniques. In the following chapters, I will present sketch-based search techniques, interactive exploration approaches and guidance concepts for analyzing global and local patterns in scatter plot visualizations.

Scatter Plot Matrix

A scatter plot only visualizes the relationship between two dimensions (bivariate data). To investigate the whole data space of a high-dimensional data set a scatter plot matrix (SPLOM) can be used, which shows all pairwise scatter plots of n different variables in a matrix [41, 214]. A SPLOM consists of n^2 cells, where each column and row reflects one data dimension. Thus, data analysts can inspect the changes of independent variables according to a dependent variable by scanning the rows and columns respectively of the matrix. Hence, we obtain a SPLOM including $n^2 - n$ single scatter plots.

Exploratory data analysis in large scatter plot matrices is a challenging task, since the number of projection views grows quadratically with the number of dimensions. Furthermore, the goal of exploration is based on a given analysis task or user, and typically not all scatter plot views are potentially relevant. Thus, often a manual exploration for finding interesting patterns, trends or clusters becomes exhausting and ineffective. To improve the exploration in large SPLOMs, I will investigate visual guidance and recommender approaches based on eye-tracking data that take already seen plots into account.

Multidimensional Projection View

Instead of using multiple views, like in a SPLOM, multidimensional data sets can also be visualized in one single view by means of multidimensional projection techniques or dimension reduction techniques. Multidimensional projections map data points form a higher dimensional data space into a lower dimensional data space, typically into a 1D, 2D or 3D representation space [139, 147]. Such a mapping can be achieved by dimensionality reduction techniques and displayed by means of visual representations, which may vary from points on a plane to graphs, surfaces or volumes. Dimensionality reduction techniques such as principal component analysis (PCA) [100] and multidimensional scaling (MDS) [45] reduce to a number of dimensions in the data, while preserving most of the relevant structures in the visualization. Thus, for instance, a n-dimensional data set can be projected to a 2D domain and visualized by a standard scatter plot.

Another method to visualize multidimensional data is to use radial axes plots, such as RadViz [141] and Star Coordinates [104]. In radial axes plots, n-dimensional data is mapped to a two-dimensional plane by arranging the coordinate axes in a circular manner. Basically, there are two classes of radial coordinate visualizations: non-linear projections, such as RadViz, and linear projections, such as Star Coordinates. Both techniques use a point-based representation and have a similar visual encoding as scatter plots. To explore trends, clusters and outliers in n-dimensional projection spaces, interactive exploration tools can be used to modify the radial axes arrangement and thus discover how axes manipulation will influence the underlying data points. My research interests include the analysis in multidimensional projection spaces and the exploration of local patterns in various subspaces.

Trajectory Visualization

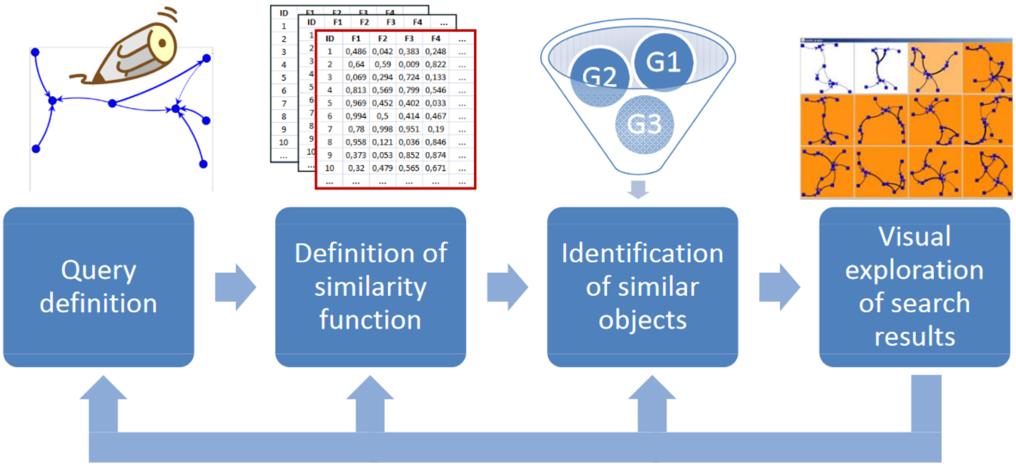
Nowadays, huge amounts of movement data are collected by mobile tracking devices and location-based services. As a consequence of these large accumulations of data, visual analytics has become an important field of application for the analysis of movement data. By considering time and spatial information as dimensions a trajectory visualization can be created to reveal patterns of moving objects. In such trajectory visualizations, data objects are often represented as discrete objects whose spatial position can be visualized as points. For example, in map visualizations, the spatial position of individuals can be represented as points by using latitude and longitude coordinates. By connecting data points according to a temporal event, individual patterns in the form of paths could be revealed. In this context, trajectories can also be considered as a set of connected points on a two-dimensional plane.

In recent years, many visual analytics systems for the analysis of movement data have been developed. Examples include the analysis of vessel movements [162, 218], traffic jams [5, 143, 211], or sports data [115, 223, 224]. The focus of many visual analytics techniques is on moving objects and the exploration of spatial- and temporal properties of individual trajectories as well as the comparison of multiple trajectories. Concerning the previously defined goals, my works focus on visual search and exploration techniques for analyzing local structures in trajectory data. Andrienko and Andrienko named this analysis task *Looking inside trajectories* [8], *"...Trajectories are considered at the level of segments and points. The methods support detecting and locating segments with particular movement characteristics and sequences of segments representing particular local patterns of individual movement"*.

2.3 Visual Search in High-Dimensional Data Visualization

The terminology visual search is not be equated with visual analysis although there are similarities in the process. The essential difference between the two methods lies in the analysis task –searching vs. analyzing. Tasks in visual analysis relate to the identification of global structures, clusters and patterns in the data, and how they relate to each other. Classic data mining methods like cluster analysis and association rule mining are two good examples for this purpose. Also, interactive exploration tools that allow data overview and details-on-demand are important features in respective systems. In contrast, visual search relates to finding relevant objects and views on a more local level, based on specific query formulation. Common search tasks are for example searching for the similar visual appearance of data objects among a set of objects and detecting recurring motifs in one view or collection of views. Both methods require some notion of similarity, for example, based on descriptors to find similar patterns, to group data points or to visually highlight related objects in the data. Therefore, a common problem is the processing of complex data types, e.g., image data, video data, movement data or data sets with mixed types, which require more expensive algorithms to measure similarities between data instances. For more information, I refer to the work of v. Landesberger et al. [205], who provided a detailed comparison of both methods.

High-dimensional data sets can be presented by various visualization techniques, which all have different visual encodings. To perform a visual search for one visualization type user queries need to be compared to all views in the search space and matched to the visual characteristics,



ID	F1	F2	F3	F4	...
1	0,486	0,042	0,383	0,248	...
2	0,64	0,59	0,009	0,822	...
3	0,069	0,294	0,724	0,133	...
4	0,813	0,569	0,799	0,546	...
5	0,969	0,452	0,402	0,033	...
6	0,994	0,5	0,414	0,467	...
7	0,78	0,998	0,951	0,19	...
8	0,958	0,121	0,036	0,846	...
9	0,373	0,053	0,852	0,874	...
10	0,32	0,479	0,565	0,671	...
...	...	...	...	...	...

G2
G1
G3
Query definition
Definition of similarity function
Identification of similar objects
Visual exploration of search results

Figure 2.3: Visual search pipeline by v. Landesberger et al. [205]. They summarized the visual search process in four phases: 1. Query Definition; 2. Similarity Function; 3. Object Identification and 4. Visual Exploration.

e.g., shape, direction, size, of the patterns. Depending on the visual representation abstraction levels need to be chosen by which the query is specified. For example, in scatter plot search, a query by sketch is possible where users can sketch a draft shape of the desired pattern for search. Therefore, appropriate user interfaces are needed that enable users to specify the query visually and interactively. The similarity of user sketch and data patterns can be evaluated by extracting distinctive features, e.g., density, connectivity or edges, of draft shape and data patterns. Following, distance metrics can be used to rank to most similar patterns and present the sequence of found results according to the sketch. In document retrieval list representations are commonly used to present the most relevant results. For visual search, other result representations, like thumbnail views or clustered views, may be more suitable to show an overview of visually similar results. Furthermore, detailed views for feature comparison can be included in the result representation that helps to interpret the similarity of found results in relation to the user sketch. In [205] the general process of visual search is summarized in four phases and illustrated as a pipeline, as shown in Figure 2.3.

This search process has been applied to various application domains. A very popular area where visual search techniques or more specifically sketch-based searches are used is in 3D object retrieval. Usually, the data structure of 3D objects is encoded by polygonal meshes and is not directly suitable for similarity comparisons. To compare the similarity of two 3D objects mesh descriptors are needed that abstracts the data by their general shape, e.g., feature-based, graph-based or geometry-based. These extracted features are then stored in a vector, so-called feature vector, and can be compared based on their distances. To name a few examples, in [51, 63, 113, 127], 2D query interfaces are used to sketch shapes of 3D objects and compute the similarity based on their view similarity. Similar visual search techniques are also used in the image retrieval domain. In this area outline sketches are typically used as input queries and Histogram of Oriented Gradients (HOG) descriptors for the comparison [50, 87].

Further research areas that consider visual search techniques include analysis of graph data, time series data and scatter plot data. Finding relevant patterns is a well-known problem in various application areas of visual analytics. In [206], a system for graph query building is proposed that supports the process with three ways of defining the query graph, i.e., query-by-example, query-by-sketch and a combination of both. As in the case of 3D objects and images a descriptor is needed that stores the visual characteristics of data patterns in a feature vector. In this work, the similarity of graphs is measured by using graph topological features. One well-known visual search system for time series data is the TimeSearcher by Hochheiser et al.[80, 81]. TimeSearcher is a dynamic query tool that allows direct manipulation of query constraints by using timeboxes –rectangular query regions that can be directly drawn on a graph. Furthermore, a sketch-based search for time series data has been proposed by Bernard et al. [19], which uses curve descriptors to find the most similar patterns in earth observation data. In [163], a content-based search system including query-by-sketch options is introduced for scatter plot data. The system uses regressional features to describe the functional form of data patterns and computes the similarity between user sketch and data patterns by the relative goodness-of-fit to each model.

2.4 Interactive Exploration Techniques in Visual Analytic

As mentioned in the previous Section 2.3, are interactive exploration tools major components of today's visual analysis systems. These tools allow users to directly interact with the visualizations and offer the possibility to manipulate data objects in the visualization dynamically. In recent years, a large number of interaction techniques have been developed for various data types, exploration tasks and user requirements.

In the history of visual data analysis, John Tukey has introduced in 1977 Exploratory Data Analysis (EDA) as an approach to assist hypothesis generation and data understanding through graphical representations [201]. Early on he recognized the importance of interactive exploration and promoted the use of interaction techniques with graphical representations. Other pioneering works that contributed to the area of EDA are the work of Tufte [200] and Bertin [20] that focused, among other things, on inherent 2D-/3D-semantics, general rules for layout and color composition. Later, Shneiderman described in his work [183] the *Visual Information Seeking Mantra* that has established as today's standard guideline for visual data exploration. The mantra follows a three-step process: Overview first, zoom and filter, and then details-on-demand. The basic idea is that users start with an overview visualization to gain insight into the entire view space. Afterward, users may explore the view space for interesting patterns or groups and focus on individual data items –subsets. This can be archived by the second step *zoom and filter*.

Both zooming and filtering help to reduce the complexity of the data by removing extraneous information from the view. Depending on the visualization, the number of represented data items can vary considerably and impair visibility. The details-on-demand technique allows a more compact visualization that provides detailed information, e.g., metadata, on a point-by-point basis without changing the representational context.

Keim distinguishes between two categories of techniques to interactively change visualizations, namely *interaction techniques* and *distortion techniques* [106]. Interaction techniques can be used to dynamically change the visualization according to the exploration items and make it possible to relate and combine multiple independent visualizations. While distortion techniques show portions of the data with a high level of detail and simultaneously preserve an overview of the remaining data with a lower level of detail. Furthermore, Keim classified interaction techniques into five categories:

1. **Dynamic Projections:** allows to dynamically change the projections to explore multidimensional data sets. A good example therefore is the Grand Tour system [12], which shows a series of interesting 2D projection views of a multidimensional data set.
2. **Interactive Filtering:** helps to explore large data sets by filtering individual data items directly in the visualization. This is especially important for focusing on interesting subsets in large data sets. Magic Lenses [25] and Dynamic Queries [81] are example tools that provide such filtering possibilities.
3. **Interactive Zooming:** means to interactively change the level of detail when zooming-in or zooming-out in a visualization. The TableLens approach by Rao and Card [155] follows this concept for tabular data. It presents numerical values as small bars to keep the table compact and allows users to interactively zoom-in for more detail –also known as focus+context technique.
4. **Interactive Distortion:** supports data exploration processes by preserving an overview of the data while showing other potions of data with a higher level of detail. In [110], a spherical distortion is used to enlarge a focus area while preserving context information. One current approach by Lehmann and Theisel [118] combines the magic lens concept with interactive distortion techniques based on projection coefficients to reduce the distortion on local areas in multivariate projection spaces.
5. **Interactive Linking and Brushing:** the basic idea of linking and brushing is to combine different visualization techniques to overcome the shortcomings of one single technique. Selected data items (brushed elements) will be highlighted in all visualizations and allows to discover information that might be hidden in one view. Well-known systems that support linking and brushing include Polaris [190], XGobi [192] and Xmdv [212].

One major advantage of these techniques is that users can explore the data on a local level, focus on individual areas and do not have to consider the whole data set at once. Thus, new findings like sub-clusters, local data structures and other interesting patterns can be detected that may be hidden in a static visualization.

However, since these taxonomies have been published a lot of novel interaction techniques were developed for different applications. Recently, interaction techniques have been presented that focus on novel technologies, such as touch displays, head-mounted displays, speech recognition and sensor devices, for visual analytics systems. Isenberg et al. [140, 204, 219] worked on multi-touch gestures to interact with visualization techniques on touch-sensitive displays. For instance, they investigated interaction approaches for data selection and manipulation based on touch inputs, and proposed design considerations for focus+context lenses on tabletop displays. Reiterer et al. [34, 88] introduced interaction techniques for visualizations in immersive environments (augmented/virtual reality). They developed a collaborative 3D parallel coordinates visualization anchored to a touch-sensitive tabletop that can be controlled by touch inputs and mobile devices. Another interesting area that found its ways into visual analytics is natural language processing. In [188], a multimodal interaction system for visual exploration of network data was presented that supports touch- and speech-based interactions. The system allows users to navigate through the visualization by touch interactions, and filter and highlighting data points by natural language inputs. Furthermore, there exist interaction techniques that utilize eye-gaze information to manipulate objects [150] and to help exploring large hierarchical data sets [185]. In [185], an eye tracker is used for interactive zooming in large graph visualizations. It captures the user's focus points to expand or compress parts of the visualization during explorations tasks.

2.5 Visual Guidance and Recommender Systems In Visual Analytics Processes

The basic idea of visual analytics is to support the user in complex analytical tasks by providing automated analysis methods and visual interactive means. In general, the interplay between visualization and models plays an important role for knowledge generation in VA processes (c.f. Section 1.2 and Section 2.1). Interactive exploration tools can be used to directly select data objects in the visualization, which serves as input for analytical models. On the other hand, changing parameter settings may influence the visual encoding of the underlying model and create new output visualizations.

In practice, however, there is the problem that these methods are often not that simple to use. Visual analytics systems are often built for expert users in a particular domain. These users are usually able to identify and interpret phenomena in data visualizations but are inexperienced with VA methods. Often parameter settings, e.g., clustering settings or model selection, may be challenging for novice users in VA. Furthermore, there exist several methods to explore large data sets or to search for specific patterns (c.f. Section 2.3 and Section 2.4). To make proper progress in the analysis task it also requires some knowledge to choose the most appropriate tools for a given task and know how to get from one view to another. What is needed are guidance approaches that support the user during their analysis tasks by recommending parameter configurations, data selections and explorations hints. The general importance of user guidance is also discussed in the work of Ram and Hunter [154].

In general, there are several dictionary definitions about guidance that are quite suitable for VA applications. For example, the Oxford dictionary defines guidance as:

> *"Advice or information aimed at resolving a problem or difficulty, especially as given by someone in authority.".*

In the context of VA systems, the phrase *resolving a problem or difficulty* may refer to a challenge that emerges from data exploration or search tasks, and *someone in authority* could be the system including analytical algorithms. Further perspectives regarding user guidance can also be found in the scientific literature. For example, in human-computer interaction, Engels [56] presented a system that breaks down the complexity of a typical KDD-task and supports users in selecting and using machine learning techniques. Therefore, he defined a problem as an artifact that contains three elements, 1. an initial state, 2. a goal state and 3. a discrepancy between those, and decomposed the main problem in a sequence of sub-tasks that can be solved easier. In [134], an automated knowledge model selection is presented that supports parameter settings based on local optimization. This approach can be of great help for novice users, especially if the user has no background in machine learning.

In the visualization community, guidance is also a hot research topic and has several related notions, such as recommendations, assistance and visual guidance. In this research field, users are often faced with additional challenges besides parameter settings for algorithmic models. Users have to decide which visualization technique should be used for a given data set or which parts of the data are relevant for a given task. To overcome the question of visualization type, the commercial data visualization tool Tableau [130] provides the *show me* feature that suggests suitable visualizations as a starting point based on selected dimensions of a data set. In [16], Behrisch et al. introduced a recommender system that suggests interesting views in large scatter plot collections based on explicit user relevance feedback. Based on Scagnostics features of relevant examples, a decision tree is trained to identify additional views for further exploration. A different guidance approach by Lehmann et al. [116], called Visualnostics, uses pictograms to visually communicate certain data properties, such as correlations and distributions, in high-dimensional projection views. Furthermore, Silva et al. [184] proposed a recommendation model based on eye-gaze to guide the user in exploration tasks for time series data. The model utilizes time series similarity functions together with user interests captured by an eye-tracker to recommend potentially relevant patterns.

As one can see, there exists a great diversity of guidance approaches in data visualization and visual analytics. For a better understanding of guidance, Schulz et al. [166] characterized the different types of guidance in visualization and defined key aspects for guidance. They consider guidance by four aspects which include *guidance context*, *guidance domain*, *guidance target* and *guidance degree*. According to their scheme, one should consider (1) the expertise of the user to determine to which extent guidance is needed, (2) where guidance can be applied, e.g., data space or view space; (3) how guidance should be presented, e.g. direct or indirect; and (4) how much guidance should be provided. A more detailed version of this characterization model is given by Ceneda et al. [38]. They extended the scheme by Schulz et al. [166] with respect to the knowledge gap of users, the input and output of the guidance generation process, and the degree of guidance that is actually provided to the user. In the context of the three main characteristics, the authors elaborated mayor questions that guidance approaches should consider. The questions are:

1. **Knowledge Gap**: What does the user need to know to make progress?
2. **Input and Output**: What is the basis for the guidance generation, and what is the answer to the user's problem and how is the answer presented?
3. **Guidance Degree**: How much guidance is provided?

Moreover, they provided an extensive overview of state of the art examples with guidance connections to the model and discovered that none of these approaches cover the whole guidance spectrum.

Part I

Visual Search Methods for High-Dimensional Data

CHAPTER 3

VISUAL SEARCH FOR SCATTER PLOT PATTERNS

Contents

The analysis of high-dimensional data sets may involve several problems due to the large number of dimensions, which is also known as the *curse of dimensionality* [91]. Although there exist different visualization techniques for exploring particular data types, it is still challenging to find interesting data patterns in large exploration spaces. For instance, the exploration space of a Scatter Plot Matrix (SPLOM) grows quadratically by the number of dimensions n. Consequently, a n-dimensional SPLOM will contain $n^2 - n$ single scatter plots, and thus makes manual exploration for data sets with hundreds of dimensions practically impossible. One popular way to narrow down the search space is to use search and filter methods that eliminate irrelevant content from the search space. Such search tools are commonly used for document retrieval, which allows user to precisely search for textual content and meta-data. Usually, it is also be used as exploration procedure if the first results are not satisfying. For example, users may start an exploration by an initial search, browse through the results, refine the search query and repeats this process until desired documents are found.

However, defining efficient queries for data patterns in high-dimensional data sets is challenging, particularly because the visual representation of a pattern depends on the visualization technique, e.g., scatter plot, parallel coordinate plot, glyphs. To perform a textual search to find data patterns is often not feasible, since interesting data patterns must be evaluated and annotated in advance. Furthermore, annotations must be consistent for all kinds of patterns and match the users' descriptions. Since there is no standardized definition of data patterns, it is very hard for users to specify a precise textual query to find a particular pattern in the data set. A more suitable approach would be visual search techniques that focus on matching actual data patterns with a visual query of the user. In this way, users may visually define a representation of the data according to their interests.

In this chapter, I investigate two visual search approaches to support retrieval and exploration of patterns in scatter plot data. The main idea is to provide a better and simplified procedure to find interesting patterns in large data sets that operate without the use of textual data. Therefore, I introduce sketch-based search and example-based search, also known as Query-by-Sketch and Query-by-Example technique, for scatter plot retrieval, and compare the impact of those approaches. I demonstrate the usefulness of those approaches by applying real-world data sets and show how data patterns can be formed and retrieved by similarity search functions.

This chapter is based on:

[173] **Guided Sketching for Visual Search and Exploration in Large Scatter Plot Spaces,** L. Shao, M. Behrisch, T. Schreck, T. von Landesberger, M. Scherer, S. Bremm and D. A. Keim. *EuroVis Workshop on Visual Analytics (EuroVA). The Eurographics Association, 2014.*

[178] **Guiding the exploration of scatter plot data using motif-based interest measures,** L. Shao, T. Schleicher, M. Behrisch, T. Schreck, I. Sipiran and D. A. Keim. *Journal of Visual Languages & Computing, 2016*

[179] **Query by Visual Words: Visual Search for Scatter Plot Visualizations,** L. Shao, T. Schleicher and T. Schreck. *Eurographics Conference on Visualization (EuroVis) - Poster Paper, 2016.*

3.1 Introduction

Nowadays, vast amounts of data are rapidly created in many application domains and thus the problem of effective and efficient access to large multivariate and high-dimensional data arises. While in the past, the storage capacity was the primary problem, today the challenges comprise tasks like detecting interesting patterns or correlations in large data sets. One solution is to apply suitable visualization techniques and search for hidden information within the data. Scatter plot visualizations are one of the most widely used and well-understood visual representations for bivariate data. They can also be applied for high-dimensional data via dimensionality reduction or the scatter plot matrix representation [214]. Nonetheless, perceiving and finding interesting patterns in large scatter plot collections is a challenging task, especially when working with large scatter plot matrices.

Manually searching through large amounts of data views is exhaustive and may become infeasible for high-dimensional data sets due to the curse of dimensionality. Recent work in Visual Analytics has focused on computing interestingness measures, which can be used to filter and rank large data spaces to present the user a good starting point for exploration. Specifically, several previous approaches, such as [186, 194, 216], have focused on interestingness measures based on *global* properties of scatter plots for ranking and filtering. However, global interesting scores do not consider the impact of local patterns, which add to the overall interestingness of a scatter plot. Often, it is a combination of several different local scatter plot patterns which by their composition constitute interesting data views. Furthermore, these quality measures may fail to reflect given user interest, since interestingness is strongly dependent on the application domain and user context.

As an alternative, visual search techniques can be used that allow users to express their interest in view patterns and foster the exploration of large data sets. For instance, a sketch-based search enables an easy and intuitive way to define data patterns, especially with novel devices that support freehand drawing like touch screens, drawing tablets. Recently, several visual analytics systems based on interactive whiteboards have been presented for data exploration [33, 208]. Another approach for exploring large visualization spaces could be an example-based search, which allows users to interactively build a query by using template patterns.

In this chapter, we investigate the impact of these two visual search approaches to discover scatter plot similarities derived from *local* data properties. In order to compare local properties, a non-parametric segmentation approach based on an adapted minimum spanning tree clustering is used to extract local patterns from scatter plots. Thus, users may search for interesting scatter plot views by either sketching rough patterns or by building a query by using the extracted patterns as templates. The general search approach is inspired by methods from the image analysis domain. More specifically, we extract image features to compare the user sketch with all target patterns to find views containing matching local patterns. Since the features characterize local properties it enables a search for data patterns at different scales and positions. Furthermore, are the visual features used as a basis to cluster scatter plot segments into groups of similar patterns, called motifs. The motifs can either be used to represent retrieval results (cluster view) or for interactive construction of scatter plot queries, i.e., by using available motifs as building bricks as query (query by example). We demonstrate the effectiveness of both approaches by a case study on large real-world data sets and discuss the benefits of both approaches.

The remainder of this chapter is structured as follows: In Section 3.2, we discuss related work and show commonalities and highlight differences. Section 3.3 gives an overview of our general search approach by using a motif-based dictionary and detail how the dictionary is generated. In Section 3.4, we present our search system including different user interfaces for sketch-based and example-based queries. Next, in Section 3.5, we apply our implementation to different data sets and showcase retrieval examples. Our approach is a first step to visual search based on local patterns, and we discuss limitations and a range of possible extensions in Section 3.6. Finally, Section 3.7 concludes this chapter.

3.2 Related Work

Several works support the exploration of large scatter plot data by means of ranking, filtering and searching functionalities. Typically, these approaches focus on global features and do not explicitly consider local properties of interest in scatter plots. When it comes to exploring and searching patterns in scatter plots it is of utmost importance to include local patterns, which may be unrevealed in traditional global approaches. Next, we review a selection of works and point out the current gaps in this research field.

Visualization of Scatter Plot Patterns

Visualizations of scatter plots need to have an appropriate aspect ratio and scale to reveal correlations, patterns, trends and clusters. This is challenging since the identification of patterns in scatter plots, and the notion of interestingness, are subjective in nature and depend on scale and proportions. Most existing aspect ratio optimization methods rely on properties of line segments displayed in a plot. In [42], it is suggested to use segments of a virtual polyline that connects all existing data points of a scatter plot, or the segments of a regression line through the plot. Talbot et al. [193] showed that this approach is suitable for data containing trends, but can be less appropriate for data, which do not have this kind of functional relationship. Hence, they proposed a method based on contour lines resulting from a kernel density estimation, which is able to deal with pairs of variables without functional relationships. In a recent approach, Fink et al. [59] presented a scatter plot aspect ratio calculation that is based on the Delaunay triangulation of the data points. The authors claimed that the aspect ratio is appropriate if the edges of the Delaunay triangulation have certain geometric properties. In [135] a visual separation measure based on an extended minimum spanning tree was presented to derive local patterns in projection mappings.

Another well-known problem of scatter plots is the degree of overlapping and overdrawing data points, which makes the identification of subgroups more difficult. In [132], an abstraction approach was introduced to group dense data points and to reveal relationships between subgroups by using smooth contour lines in combination with different color codings. Another recent work on visual abstraction has been presented by Chen at al. [40], where a multi-class sampling technique is presented that reduces the overdraw and preserves the point distributions for quantitative analysis. More generally, a study on perceptional factors, which links scatter plot properties with perceived interestingness and interpretability is given in [168].

Sketch-based Search Techniques

In the image and video retrieval domain, sketch-based search has developed into an important retrieval technique over the past few years. It allows the user to search for specific content of a video or image, and deals with the basic problem of the textual search that merely searches for tagged annotations, which are related to the video or image. Recently, a lot of research has been done in this area. In [35, 50], novel approaches for indexing and evaluating sketch-based systems were proposed. Recent work in sports analytics [115] uses sketch-based search techniques to analyze player movements in rugby matches. It uses multiple distance-based similarity measures, which compares the user sketch against video scenes. Besides the traditional image and video search, sketch-based search techniques can also be applied to other complex data sets. In [51, 113], 2D sketches are used to search for 3D objects. Lee et al. [114] make use of synthesized background shadows to help users in sketching objects. Further applications of sketch-based search include retrieval in graphs [206], or for bivariate data patterns in scatter plots [163, 173]. Regarding data generation, in [209] 2D shape sketches are used to create and manipulate high-dimensional data spaces.

Navigation in Scatter Plot Space

The effectiveness of analyzing large scatter plot data also depends on appropriate navigation facilities. Animated navigation and extrusion-based transitions between views were proposed in [54] to navigate in scatter plot matrix spaces. Scherer et al. [163] introduced a search and navigation interface that is based on the scatter plots global regression features. Another possibility to explore and navigate through scatter plot spaces is the usage of projection visualizations in connection with extracted features. For instance, radial projection visualizations like Star Coordinates [103, 104] or RadViz [84] can be utilized to show scatter plot clusters, trends or outliers by using the features as projection dimensions. Lehmann et al. [116, 118] introduced a visual guidance approach for those projection visualizations and proposed a generalization of both visualizations to achieve a higher degree of freedom for finding suitable projections. Furthermore, in [164] an experimental study compared the effectiveness of global features for ranking scatter plots by similarity.

3.3 Pattern Segmentation and Feature Extraction for Visual Search

The main goal of the two approaches is to foster the exploration and retrieval of interesting patterns when facing a data set with a large number of scatter plot views. In contrast to previous approaches that use global features, these approaches will include *local* properties of interest in scatter plots. It is therefore possible to search for global and local patterns – meaning that it is possible to find matching patterns regardless of size and position in a scatter plot view. In order to achieve this, the proposed approaches are based on a *motif dictionary* that involves local scatter plot segments from the set of all scatter plots. The dictionary can be computed in a preprocessing step and will contain prototype scatter plot segments, called motifs, that represent the different local scatter plot shapes occurring in a given data set. In this work the motif dictionary is used as indexing for similarity search but also serves as a basis for guidance functions, e.g., measuring interestingness in views (see Chapter 7). When searching for a specific pattern, whether sketch or example-based, the dictionary is used to find the best matching candidates in the overall search space.

The motif dictionary is generated as follows: First, all scatter plots need to be partitioned into a set of local scatter plot segments. We employ the idea of a minimum spanning tree-based segmentation, as introduced by Jana and Naik [96], to split scatter plots into several local patterns. After that, image descriptors are used to extract grid-based features for each pattern and stored within a vector. The feature vector of a pattern characterizes shapes on a local level and allow subsequent algorithms to compare structures with other patterns. For instance, one can compare the feature vector of a user sketch with all entries in the dictionary. To facilitate similarity search, similar patterns in the dictionary can be grouped in advance by a clustering approach. Thus, it will produce a number of clusters (i.e., dictionary entries), which represent the local motifs occurring in the overall data set. This clustering step relies in turn on visual features extracted from the individual local scatter plot segments. This approach is inspired by techniques from image processing and in particular the segmentation of local areas-of-interest in images and feature extraction. Figure 3.1 shows the workflow to create the motif dictionary, which is detailed in the following.

1. *Segmentation of Local Scatter Plot Patterns.* We perform a segmentation of each scatter plot into regions of interest. To this end, a Minimum Spanning Tree is constructed and the longest links are removed, hence segmenting the data into a number of dense clusters.
2. *Visual Feature Extraction.* For each scatter plot segment, a feature vector based on gradient orientation and density histogram is computed [146].
3. *Dictionary Generation.* The dictionary is formed by clustering the set of scatter plot segments, using the visual feature vector as a basis. The dictionary consists of scatter plot segment clusters together with the size of the cluster members and quality statistics.

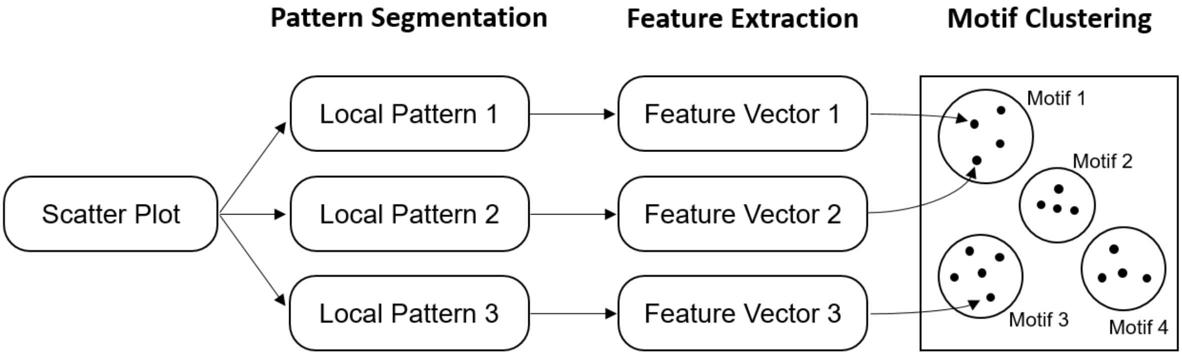
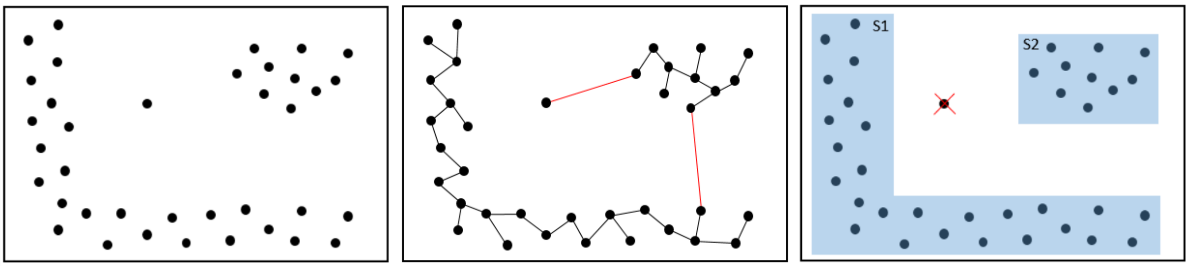


Figure 3.1: First, scatter plots are segmented by an adapted minimum spanning tree clustering approach to extract all local patterns from the view collection. Then, visual features of each scatter plot segment needs to be computed, which are used as input for the subsequent clustering step. The final motif dictionary is generated by applying k-means to the set of all segments. This leads to a number of k clusters and k dictionary entries respectively. In this example a scatter plot is subdivided into three local patterns, which in turn belongs to two motifs.

Segmentation of Local Patterns

The automatic segmentation of scatter plots is the basis of the local pattern search and hence requires special attention. Since each scatter plot may contain a distinct set of characteristics regarding its motifs (e.g., number, shape), its points (e.g., density) and the input scale of the dimensions, a flexible segmentation method is needed. A manual adjustment of segmentation parameters or the incorporation of domain knowledge in the segmentation process is practically infeasible because many data sets under consideration contain possibly thousands or more plots. To accomplish this challenges a technique is needed, which is independent of user parameters and at the same time able to find segments regardless of their shape. Since basic potential segmentation techniques like k-means or DBSCAN would not satisfy these requirements, we decided to use a minimum spanning tree (MST) based clustering technique [96]. Another important prerequisite for the segmentation technique is to extract meaningful motifs that have a strong connection to human perception. Experiments in [49] have shown that the MST method produces similar structures in the constellation of connection pairs of points (stars) as humans. The idea of the segmentation technique is to represent the data by means of a minimum spanning tree and iteratively remove the longest edge to derive an appropriate amount of local scatter plot segments. Recent research on MST clustering has been conducted by Jana et al. in [96]. While their MST approach assesses the clustering quality in each iteration by an internal validity criterion [126], we follow-up on their research by introducing an *outlier-insensitive technique* that focuses on larger clusters containing more than one point.

Figure 3.2 demonstrates this segmentation approach by splitting a scatter plot into two segments. The first step is to represent the data by a MST and iteratively remove its longest edge, see Figure 3.2 (b). In this example, the longest edges are colored in red and will create new



(a) Original scatter plot.

(b) Scatter plot represented as MST.

(c) Final segmentation after applying the segmentation technique.

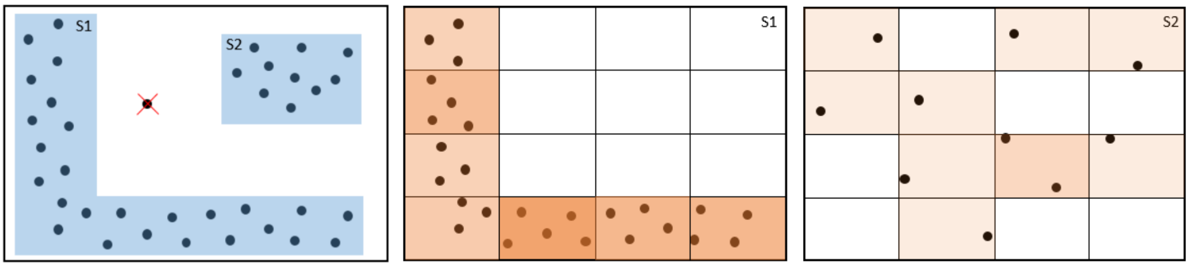
Figure 3.2: Demonstration of the MST-based segmentation approach.

segments after removing them. In order to exclude undesired extreme cases, i.e. when single points are considered as an individual cluster, a distance based outlier detection technique is used. The outlier detection technique compares the length of the last removed edge with the nearest neighbor distances and considers a point as an outlier if the removed edge is exceptionally long. The nearest neighbor distances are computed by taking the mean edge distances of k connected vertices. A neighborhood size of one indicates a potential outlier and induces the examination. Consequently, detected outliers are ignored in subsequent iterations of the segmentation algorithm, as shown in Figure 3.2 (c). This procedure is continued until the cluster quality achieves a high *F-Ratio* [64] value, which has been proven to be a good statistical measure for detecting clusters.

A crucial parameter of the outlier detection is the neighborhood size k, which can negatively influence the result if it is chosen too small or too big. If k is chosen too small, a reliable average distance value cannot be reproduced and if it is chosen too big, points from other clusters could possibly be considered as well. Although k is heavily dependent on the individual application, a value between 4 and 8 mostly showed satisfying results. Note that we will discuss the impact of different quality criteria and their influence on the exploration process and the final result set in the following sections. The final scatter plot segmentation is achieved by considering the clustering with the overall best assessment.

Feature Extraction

Since it has shown that image-based descriptors perform quite well for scatter plot retrieval [164], we adopt this approach to match user queries, i.e. sketches or query-by-example, with patterns from the dictionary. The scatter plot segments are first converted into images and then further separated into equally sized subcells, e.g., 4×4 or 8×8 grid, on which the image features are extracted. By first partitioning images into subcells, important spatial characteristics can be maintained in the feature extraction process and individually stored in a feature vector. Since the extracted features describe the visual properties of a local scatter plot segment one can



(a) Segmented patterns from Figure 3.2. (b) Density-based feature extraction of pattern S1. (c) Density-based feature extraction of pattern S2.

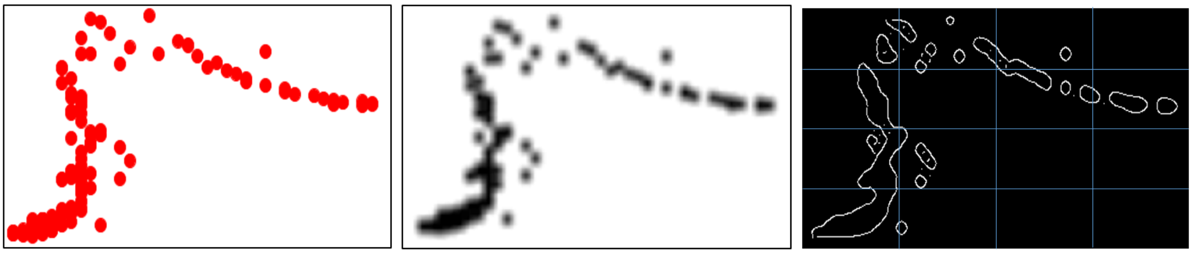
Figure 3.3: Demonstration of the density-based feature extraction approach.

search for patterns regardless of size and position. Figure 3.3 demonstrates the feature extraction process of two local patterns S1 and S2. In our approach we included two different types of image descriptors, one is based on density features and the other is based on edge features.

Density Descriptor. The density-based approach is a simple but very efficient method for searching larger areas that are covered by points, e.g., dense patterns and larger clusters. It computes the percentage of pixel distribution (coverage of points) in each subcell of an image and stores the values in a uniform vector. In Figure 3.3 the feature extraction approach is visually demonstrated on the two separated pattern from Figure 3.2 (c). In order to guarantee a scale-invariant search, extracted patterns are first scaled to a uniform size, c.f. Figure 3.3 (b) - (c)). This is followed by dividing the image into an equally sized grid and features are extracted for the corresponding cells. Figure 3.3 showcase this example by using a 4×4 grid, however also finer grids can be used. The grid-based approach enables a more accurate comparison of the user query and target patterns, whereby individual grid cells can be compared including proportional resizing. To find the most similar pattern to the user query, the Euclidean distance to all target patterns are computed and then sorted by distance.

Since the comparison for a single image is concerned with a rather small feature vector (16 dimensions / 64 dimensions) the comparison process also can be executed very quickly for large amounts of scatter plots.

Edge Histogram Descriptor. As second descriptor we propose the optimized Edge Histogram Descriptor by Park et al. [146], which stores the frequency of five different edge types in a histogram. Similar to the density-based approach this descriptor divides images into 4×4 grid and computes the frequency bins for each subcell. Consequently, it will return a feature vector containing 80 dimensions –5 values for each of the 16 subcells– that is also included in the MPEG-7 standard. Figure 3.4 shows the features extraction process from a scatter plot. First, a burring filter is applied to the scatter plot images to improve the shape description of near located points. The blurring process creates a soft-focus effect of the scatter plot patterns and



(a) Segmented pattern from extracting shape features.

(b) Outcome after the blurring process.

(c) Edge detection and subdivision of the segmented pattern.

Figure 3.4: Feature extraction via edge histogram descriptor.

helps the subsequent process to extract by actual shape, c.f. Figure 3.4 (a) – (b). After that an edge detection algorithm is used, e.g. Canny edge detection, to create an image with contours of the actual shape, see Figure 3.4 (c). Finally, the image is divided into 16 sub-cells on which the frequency of the five edge types are computed. The descriptor distinguishes horizontal, vertical, 45 degree, 135 degree and non-directional edges.

Besides the traditional features from the MPEG-7, the approach by Park et al. [146] extends the feature vector with further global and semi-global features. The global edge histogram contains frequency bins for the whole image without subdivision, whereas the semi-global histogram constitutes further cell groupings to describe the frequency of edge types for particular regions. The additional regions describe edge feature for horizontal, vertical and rectangular areas –2×2 blocks– within an image. Altogether, there are four horizontal and four vertical histograms that are formed by aggregating row-wise and column-wise cells. Furthermore, five rectangular areas characterize the corners and the center of the image. Thus, the basic feature vector is expanded by 13 semi-global features, i.e. four horizontal, four vertical and five rectangular areas, that improve the description of a pattern.

The two additional histogram bins of this optimized approach are used to increase the matching performance and can be generated directly from the previous feature vector.

Motif Dictionary

After the feature extraction of local scatter plot segments, the motif dictionary can be computed by clustering the set of local segments. By using a good clustering technique, the dictionary will include several entries –clusters– that represent the motifs of the data set. To this end, we decided to use a k-means clustering to create k entries in the dictionary. This clustering requires an appropriate vector-based description of each segment and, of course, a good selection of k. Choosing the right parameter k is a well-known issue and highly depends on the data set. Therefore, we provide a simple user interface to modify the dictionary (see Section 3.5.2) and allow users to steer parameter k and dictionary size respectively. A recent study has shown that edge orientation and density features are effective to distinguish scatter plot shapes [164].

We therefore compute these features and feed them into a k-means clustering to produce the dictionary.

To ensure proper and quick processing of search requests, all previously mentioned steps –image generation, feature extraction and clustering– are computed in advance and stored in a database. That means that after a search request only the distance of query to dictionary entries has to be computed in runtime. For sketch-based queries, the features of the user sketch need to be computed once to perform the search.

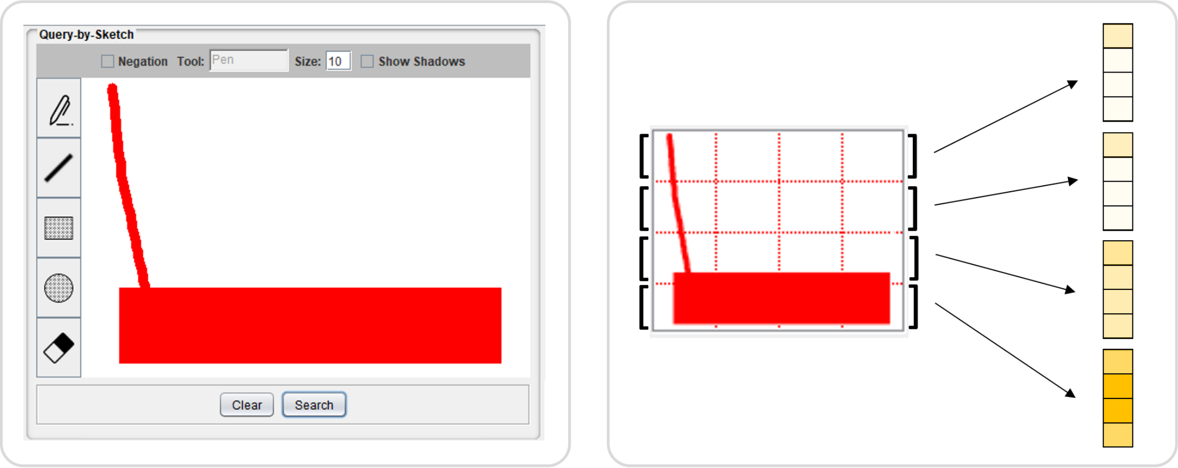
3.4 Query Formulation & Result Representation

To retrieve the extracted local patterns from the previous section, an appropriate user interface is needed where non-textual queries can be designed. Thus, we investigated different approaches for visual query formulation that ensure a user-friendly operation for the retrieval of scatter plot patterns. One of the most intuitive ways to sketch a query is by using well-known drawing tools, which are familiar to everyone, e.g., from drawing and painting applications. Another interesting approach that we want to introduce is the utilization of motif representations as building bricks to design a query based on the query by example principle. Furthermore, we discuss representations of retrieval results that incorporate visual comparisons of matching patterns.

3.4.1 Query by Sketch for Scatter Plot Patterns

A sketch-based search can be performed in several ways. One possibility would be to convert a user sketch into an actual data set of x and y coordinates, and compare the values of user sketch and scatter plot by using numerical measures. Since we decided to use image-based descriptors, it would be appropriate to store the user sketch as an image and apply the proposed descriptors. Therefore, we developed a simple sketching interface that provides beside freehand drawing also several drawing elements such as line, circle and rectangle tools to draw a query, as shown in Figure 3.5 (a). For instance, one could use the circle and rectangle tool to quickly draw filled surfaces in order to find dense point clouds or use line and freehand drawing tools to sketch more skinny patterns. For minor corrections, an eraser tool is included too. These tools can be easily steered by mouse operations or, when available, by touch operations –finger or pen. This allows intuitive control of the drawing process and lets users precisely sketch a query of interest.

To compute the similarity between user sketch and scatter plot patterns in the dictionary, comparable features of the sketch needs to be computed for each search. Figure 3.5 demonstrates how a sketch will be processed into a feature vector. This example showcases a search based on the density descriptor. It must be taken into account that a descriptor is used that has the same properties, e.g., gird size, as for the precomputed plots, and thus, the feature vector of query and scatter plots have the same length. Finally, the nearest neighbors of the query will be computed by using the Euclidean distance and ranked by distance.



□ Negation Tool: Pen Size: 10 □ Show Shadows

Figure 3.5: A sketch interface for drawing scatter plot patterns as search query. To compute the feature vector, user sketches will be first converted into an image file and divided into a grid. Then for each grid cell features will be computed according to the selected descriptor and combined into a vector. In this illustration, a darker color corresponds to a higher density value.

Search results can either be presented as a ranked list or in a dictionary view (clustered view). The ranked list shows the result set with additional metadata information, such as axis dimensions, similarity distances and motif affiliations. The dictionary view represents the result set in a more compact view by aggregating the matches according to the dictionary.

3.4.2 Query Formulation based on Local Motifs

The basis of the second approach is the motif dictionary (c.f. 3.3) that comprises a collection of available local patterns in the data set. By using this dictionary users can quickly explore the local pattern space, reuse interesting motifs as query templates and find scatter plots that involve novel compositions with the searched motif. A user query Q consists of a number of selected patterns from the dictionary, $Q = \{q_1, q_2, ..., q_n\}$, as well as approximate spatial positions of each of the local patterns. The goal is to find those scatter plots S with the local patterns $S = \{p_1, p_2, ..., p_n\}$ which show the highest similarity to Q. Therefore, the similarity of scatter plot patterns, as well as their spatial positions, are taken into account. A major advantage of using the motif dictionary is to initially eliminate candidate plots from the result set, which do not include matching motifs. This allows a faster computation of the result set and ensures that all matching plots contain at least one pattern of each motif defined in Q. Figure 3.6 demonstrates a user interface for designing queries where users can freely position a pattern and thereby actively influence the search on spatial positions of patterns. Furthermore, users can choose between either motif classes or particular patterns as building brinks for their queries. A motif class is a representative of a cluster –dictionary entry– and can be used if one is more interested in the position of a motif than the precise pattern matching. This search focuses on the spatial position

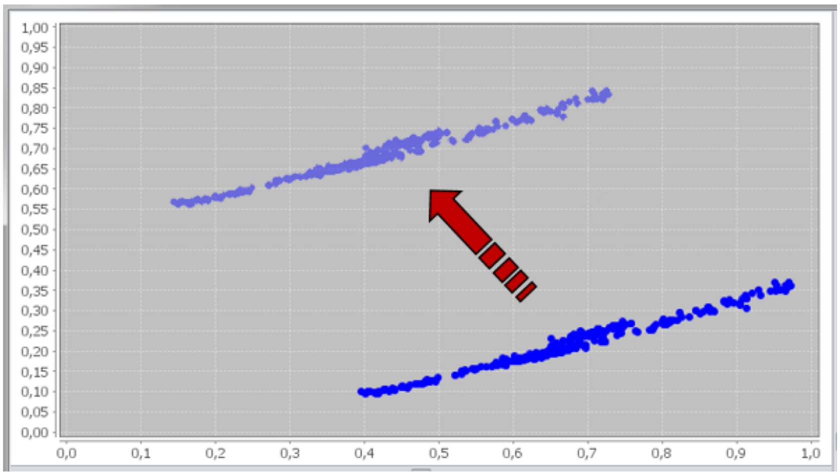


Figure 3.6: Illustration of creating a motif-based query. Users can freely position selected motif prototypes on a template coordinate system to find similar views.

of similar motifs. It considers all patterns in a cluster as similar and uses the spatial distance as a ranking criterion. In contrast, a search for particular patterns will additionally consider the similarity of pattern S to a motif of Q. As distance measure, we apply the Euclidean distance since it is the most prominent distance metric used in the area of information retrieval.

However, it has to be taken into account that a user query Q may consist of several patterns that have a non-uniform matching relationship to existing patterns of a scatter plot S. Therefore, we decided to consider three types of matching criteria, a one-to-one (1:1), one-to-many (1:N) and many-to-many (N:M) relationship, which is illustrated in Figure 3.7. The simplest comparison is the 1:1 alignment, where the user query Q has one motif q_1 and the comparing scatter plot S has one pattern p_1 as well. Since a motif may contain several scatter plot patterns, we compute the similarity distance of a pattern p_1 with respect to their dictionary entry (i.e., cluster centroid) as a constraint. To determine the spatial distance between a pattern of S to a motif of Q, we compute the center of mass on the normalized scatter plot axes of the patterns and motifs respectively. A 1:N alignment may result in scatter plots that contain one query motif several times whereas other query motifs may not be contained at all, although they have a potential matching partner. In this case, we compute the nearest neighbor pattern on the 2D plane that can be matched in a 1:1 fashion. This is especially important if the set of scatter plot patterns and the set of query motifs have a varying size. If for instance a scatter plot consists of two patterns but the user specified more query motifs, then only the two best matching patterns are taken into account. A second important search constraint is the number of motifs. Since the user also defines the number of desired motifs, this needs to be considered in addition. For this reason, we use an optional weighting factor depending on the number of patterns that have not been matched from the query due to a too small number of motifs in a respective scatter plot. Another matching case that needs a specific treatment is the N:M alignment for identical motifs. For this scenario, we

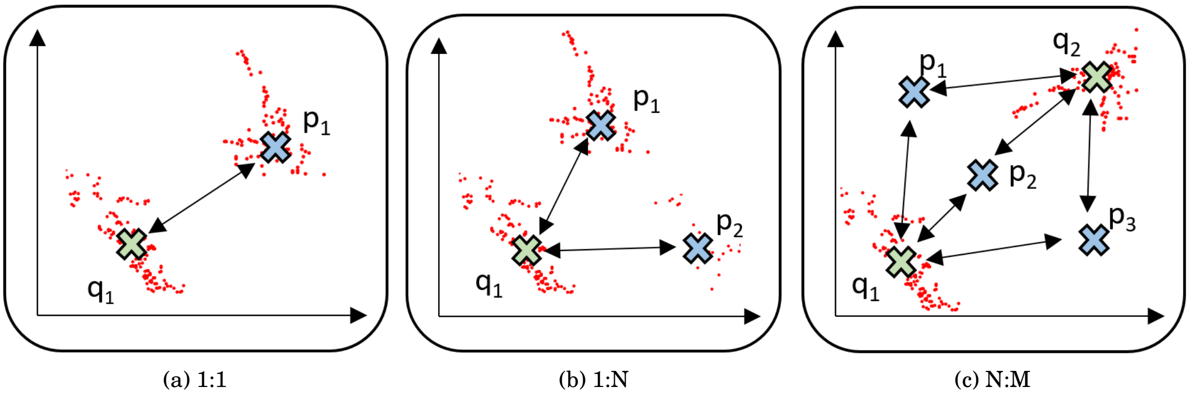


Figure 3.7: Illustration of different matching relationships in example-based searches. In a 1:1 search, the distance of one query q_1 to the target pattern p_1 is computed as weighting. In 1:N relationships to nearest neighbor of q_1 is considered as distance and for N:M search scenarios, the minimum overall distance is computed.

minimize the overall matching distance of scatter plot patterns S to the query motifs of Q.

The search interface for motif-based queries is shown in Figure 3.8. We visualize the obtained dictionary entries by an overview glyph representation (see Figure 3.8 –bottom) and use it for designing new queries. The glyph representation depicts the visual appearance of a motif by showing the motif cluster prototype (medoid segment of a dictionary entry) and a small multiple view of various associated motifs. A query can be composed by selecting motif prototypes from the dictionary. Moreover, users can choose interesting motifs as visual words and relocate them on a template coordinate system to discover novel scatter plot compositions of motifs, as shown in Figure 3.6. The best matching results are shown on the right hand side of Figure 3.8.

3.5 Retrieval Results and Use Cases

Next, we will demonstrate the proposed search techniques by using well-known data sets from the UCI Machine Learning Repository [1] and the Statistical Office of the European Union (Eurostat) [2]. First, we present the sketch-based search technique by showing a proof of concept with a labeled data set and a real-world application with unlabeled data. In the following, we showcase the usability of our query-by-example approach with a motif dictionary.

3.5.1 Query-by-Sketch

To search for local patterns in scatter plot data, we introduced a non-parametric segmentation approach to obtain all local patterns for a given data set. However, to prove the functionality of

[1] https://archive.ics.uci.edu/ml/index.php

[2] https://ec.europa.eu/eurostat/data/database

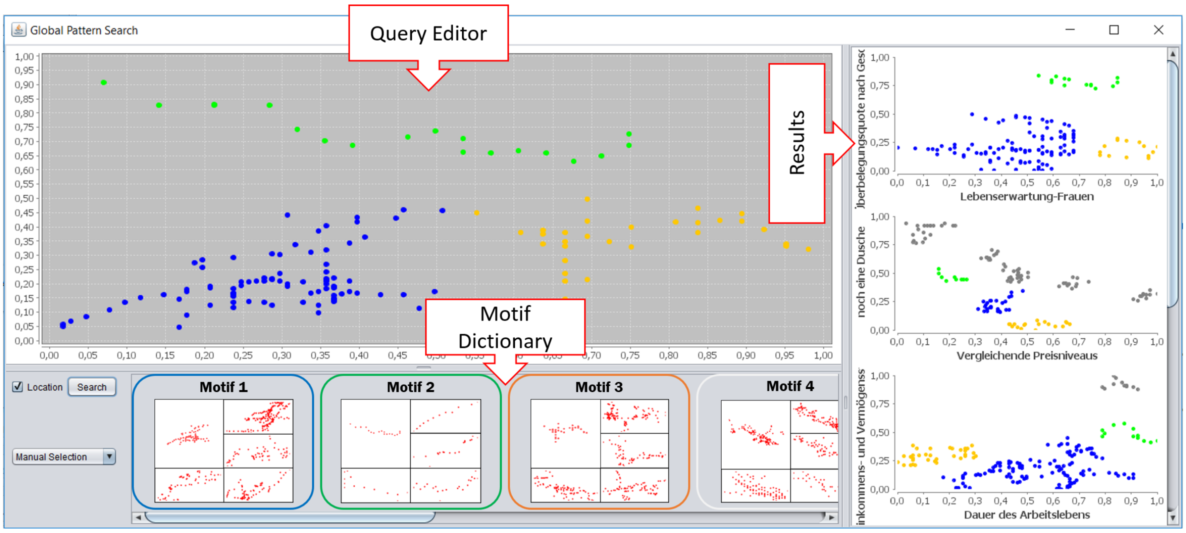


Figure 3.8: Scatter plot retrieval by visual words. A motif-based dictionary is used to compose scatter plot queries and enables the search-based discovery of novel scatter plot compositions. To create a query, users can select an interesting motif and freely position the motif prototype (enlarged plot in the glyph representation) on the query editor

our visual search technique and the underlying image descriptors, we rely on labeled data and use the class labels as pattern segmentation criterion.

Cars Data Set

The first data set that we investigate is a car data set with seven numerical attributes, i.e. *miles per gallon*, *cylinder*, *displacement*, *horsepower*, *weight*, *acceleration* and *year*. Figure 3.9 shows an overview of the data set by using a scatter plot matrix and highlights first search results. In this data set, we used the origin to extract local patterns of cars from the US (red), Europe (green) and Japan (orange), since there exist different car characteristics based on the country of origin. By using a data set with seven dimensions and three class labels, we obtain a search space with a total of 126 local patterns (42 scatter plots × 3 classes per plot).

As a first search example, we are looking for patterns that contain an exponential decrease. The drawn user query with the respective search results are shown at the bottom of Figure 3.9, as well as highlighted in the scatter plot matrix. For this search, the density descriptor (c.f. Section 3.3) is used to find the top 10 search results. One can see from the result set that the descriptor works quite well and is able to extract important features for finding similar structures. Please note that the result set is ranked based on their similarity to the user sketch; left most similar and right less similar. By considering the scatter plot matrix it can be noted that the most similar patterns to the user sketch are included in the result set. Furthermore, by taking a closer look at the results one may note that most found matches are patterns from the US, e.g. see MGP vs. Disp, MPG vs. HP, HP vs. Acc. This might indicate that Europe and Japan produce more cars

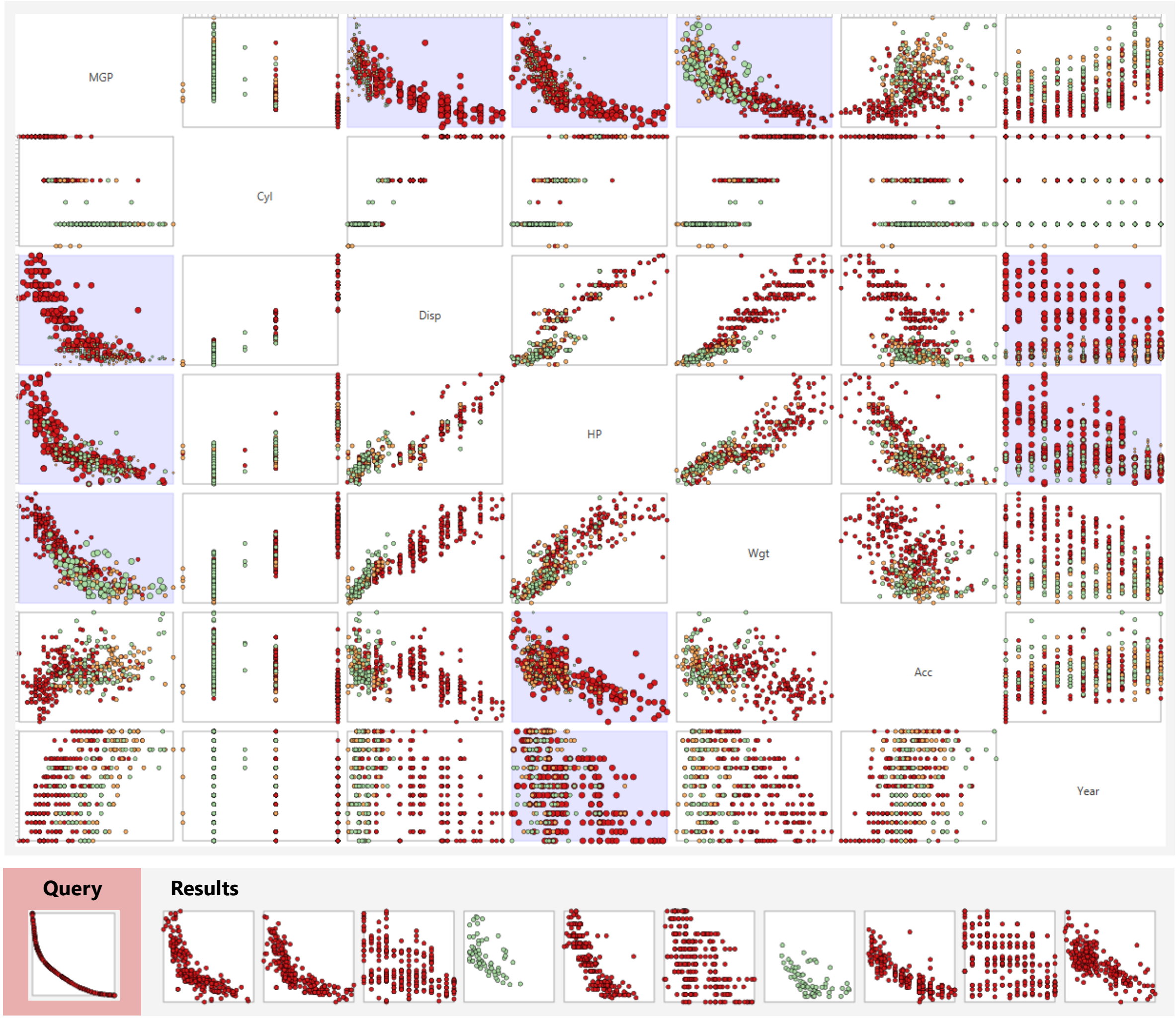


Figure 3.9: Demonstration of our sketch-based search system by using the density descriptor. Analysts can sketch a distribution of the data, such as an exponential decrease, to find similar patterns in the data. Best matching results are displayed next to the query and highlighted in the overview scatter plot matrix.

that are rather balanced in relation to the inspected attributes.

More search results are shown in Figure 3.10. We include two further queries to find patterns with strong positive correlations and no correlations in the data set. The results show that our approach is also capable of finding such structures in the data.

Wine Data Set

To demonstrate our second descriptor (edge histogram descriptor), the same approach of using class labels as segmentation has been applied to another bigger data set. In this case, we performed our experiments on the Wine data set as this data set has been used in various studies

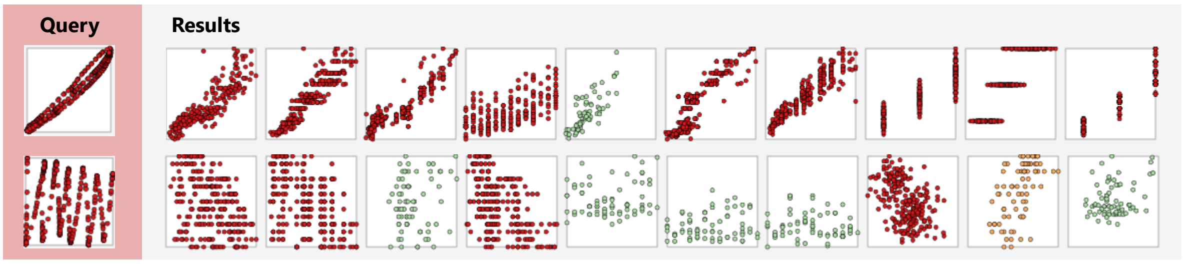


Figure 3.10: Further query-by-sketch examples of the car data set. The first query reveals that in many views US patterns (red class) have a strong positive correlation. The second query shows that it exists no correlation patterns in all three classes.

for distance metrics. The Wine data set contains thirteen attributes and 178 instances, each labeled as one of three classes.

Figure 3.11 shows search results of three queries and their respective results. This time, all results of the three queries are highlighted in the scatter plot matrix. Please note that our system provides brushing and linking for the examination of individual results and queries. In general, the data set contains many uniform distributed local patterns and do not contain many extraordinary shapes of patterns. As in the previous example, we search for patterns with positive correlations and sparse point clouds with no correlations. The visual search of the two queries reveals which classes contain strong positive correlations and no correlations. For instance, one can see that the red and green classes form stronger patterns with positive correlations compared to the orange class. Moreover, it turns out that the red class did not appear in the top search results for sparse point clouds, since it forms rather compact local patterns. The second query shows that most found patterns belong to the orange wine class.

However, the first two sought-after patterns could also be found by manual exploring the scatter plot matrix. Thus, we included a more complex pattern search that can not be easily found by a quick look at the data set. The third query shows a retrieval example for a L-shape pattern sketched by contour lines. The result set reveals that actually three patterns (ranked as 2., 3. and 4.) have quite similar shapes as the query and demonstrate that the search technique can be used to find similar patterns based on a user sketch.

Eurostat Data Set

Next, we apply our search approach to an unlabeled data set from the Eurostat. The Eurostat data repository provides a large collection of data sets each containing information about a European related topic, such as economy, population and industry. We extracted a data set containing 27 statistical attributes from 28 EU countries that show temporal changes over the last decades. The resulting scatter plot matrix contains 702 scatter plots from 27 dimensions in which each data point represents one country at a specific year. The data set is then further segmented by our adapted minimum spanning tree approach (c.f. Section 3.3) 2.561 local patterns.

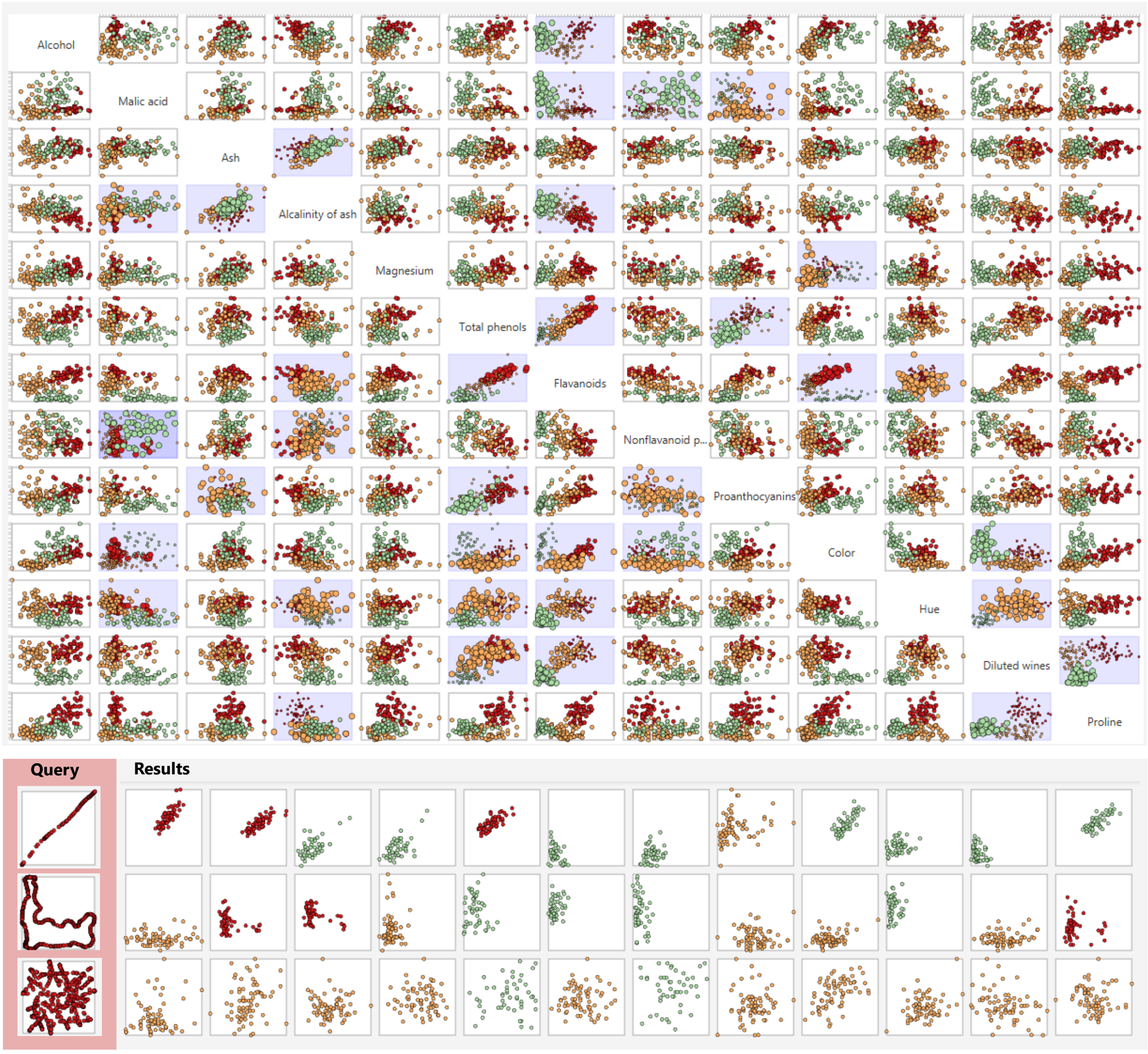


Figure 3.11: Retrieval results with the edge histogram descriptor. As the previously shown descriptor the edge histogram descriptor can be used to find similar patterns based on user sketches.

An overview of the scatter plot matrix is illustrated in Figure 3.12. The data contains a number of stringy patterns, especially in the first five rows and columns, that emerge through the segmentation of individual countries. This was reflected in search results by sketching previously shown correlation patterns. In Figure 3.12, we sought to determine whether patterns exist in the data that reflect ups and downs in the development of certain countries. Therefore, we chose a triangle-shape as query, since clusters of countries may form such contour lines as an overall pattern. According to the findings of the search, several local patterns exist that have a certain similarity in their visual appearance. In particular, the patterns ranked at first, second and fourth position show a degree of similarity if one considers the outer shape. On closer inspection,

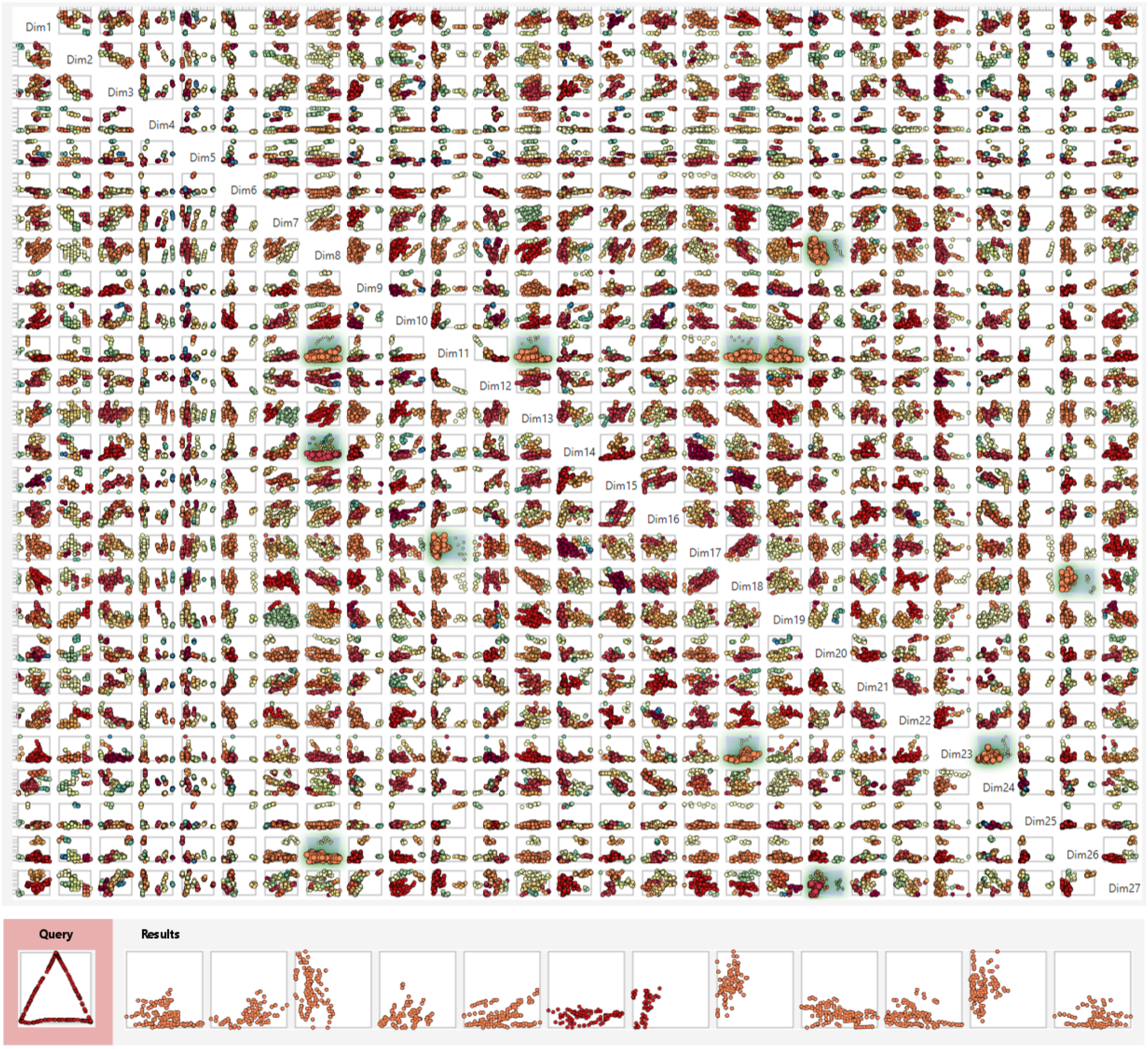
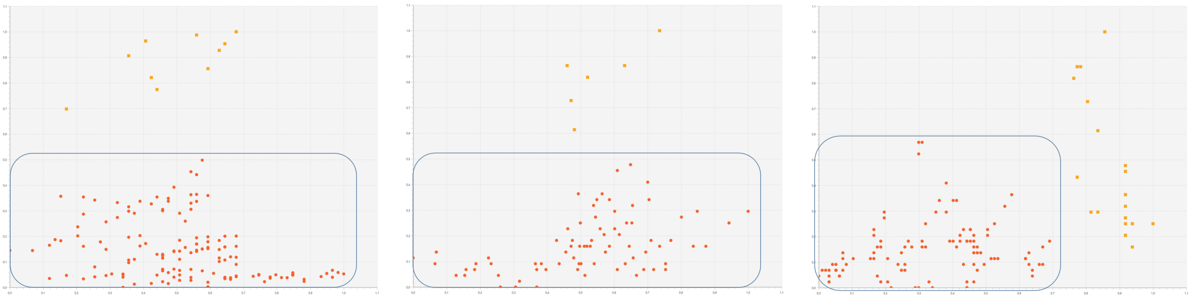


Figure 3.12: Visual search for automatically segmented patterns. In this search example, we are looking for local patterns in the Eurostat data set that contains triangle-shaped patterns.

one might see that the patterns consist of various smaller segments with positive and negative trends, see Figure 3.13. In addition, the patterns show that a number of European countries have been developed over the years similarly.

3.5.2 Query-by-Example

Selecting an appropriate dictionary size is difficult and has an impact on the subsequent process of building motif-based queries. Especially for large and complex data, it is crucial to define a good cluster parameter k. Therefore, we developed a visual exploration tool to support analysts in the search process and find appropriate parameter settings for generating a dictionary (see Section 7.4.1).



(a) 1. Rank: Greenhouse Emissions - Exports

(b) 2. Rank: Sustenance - Sanitation

(c) 4.Rank: Death rate from chronic disease - Sanitation

Figure 3.13: Good matching results with a high visual similarity to the user sketch.

The tool involves a global overview in the form of a scatter plot matrix and a detailed dictionary view of all clustered motifs, as depicted in Figure 3.14. It allows analysts to experiment with different clustering settings for a given data set. In this case, we used a synthetic data set with a few distinctive patterns as a proof-of-concept. The dictionary view provides insights into the quality of the parameter setting and shows core information like cluster representatives and cluster size. The cluster size indicates the frequency of a particular representative motif in the scatter plot space. Moreover, it hints on the practicability of the chosen clustering parameter k. To represent the cluster, we chose the local segment, which is the nearest neighbor to the k-means prototype. By clicking on a dictionary entry, all cluster members of a motif will be highlighted in the linked scatter plot matrix. Conversely, it is possible to highlight all corresponding motifs by clicking on a given segment in the scatter plot matrix. Moreover, we distinguish the different motifs occurring in the data set by using various color-codings. Thus, users can quickly recognize the distribution of individual motifs across a large scatter plot space. A further benefit of this overview is that users can estimate whether the cluster extraction threshold is configured appropriately, or whether the number of clusters should be increased or decreased, to fit the application need.

Once the dictionary has been generated, analysts can use the dictionary entries to search for a scatter plot with certain motif configurations. Figure 3.15 shows the retrieval results of an example-based query by using two motifs of the Eurostat data set, as used before. In this example query, we are looking for scatter plots that contain two kinds of motifs, one bigger widespread cluster (green) and a smaller cluster with a negative correlation (blue). Furthermore, we include the positioning of the motifs into the query and restrict the search to views that contain the green motif at the bottom and blue motif at the upper right corner. When viewing the top-ranked results, one can see that our search approach works reliably and retrieves views that fulfill the conditions quite good. All search results contain local patterns, which are similar to the query motifs and have in most cases the right positioning of the motifs. The top four results have a bigger widespread cluster at the bottom and a negatively correlated pattern right above it, as

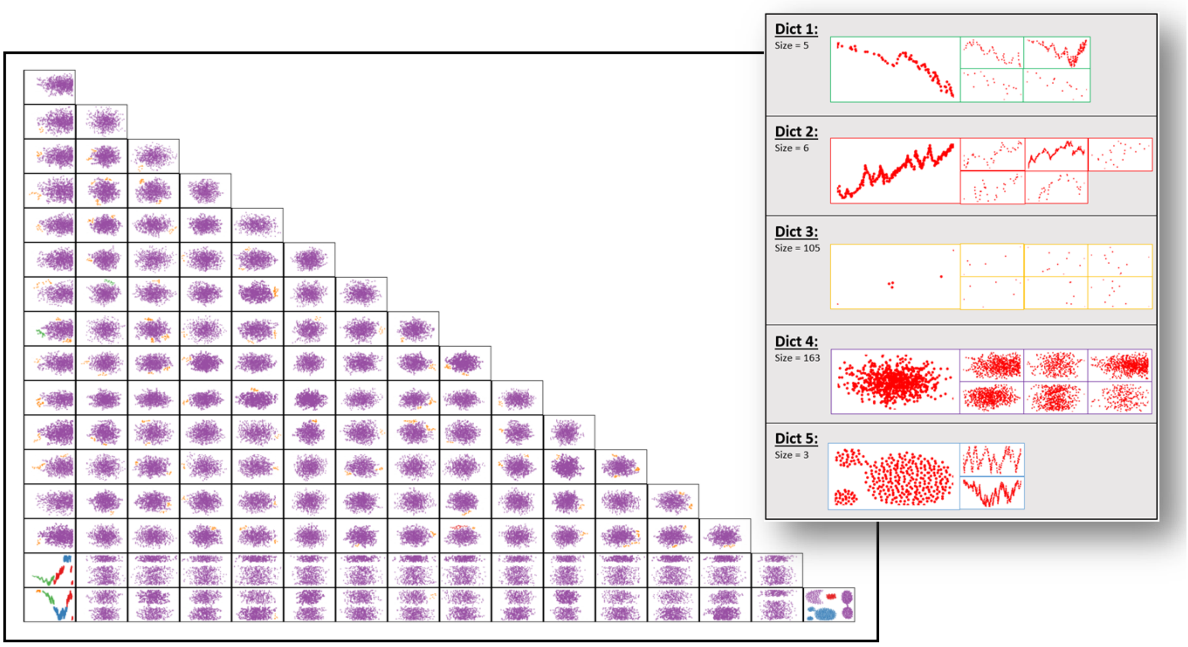


Figure 3.14: Overview of the synthetic data set with the generated motif dictionary. Please see Section 7.4.1 for more information about the data.

desired. The further results show plots with good matching motifs but with slight shifts, which lead to lower similarity scores. Finally, we found a set of scatter plot view with varying number of patterns that comprise the relevant motifs.

3.6 Discussion and Future Work

In this chapter, we proposed two visual search approaches for the discovery of local patterns in scatter plot data. Our prototype systems allow to search for local patterns based on a user sketch and configure queries based on multiple local patterns. The sketch-based approach may be a good opportunity for analysts to search for given local patterns in the data, since sketching a visual query may often be easier than defining a textual query. An example-based search interface allows to design a query based on pattern templates and is particular suited for a constellation of multiple patterns on certain positions. In our current example-based search approach, we use a motif dictionary for the input templates, since it represents a data set well and provides available patterns in the data. This step in the pipeline could also be exchanged by other catalogs of input templates. For instance, one could use the scatter plot characterization clusters by Bertini et al. [145] that involve a broad set of patterns based on human similarity perception. Another useful extension could be the integration of both proposed approaches so that one can sketch patterns as input templates and use a query editor to position the patterns for the query.

In the previous Section 3.5, we showed initial retrieval results and several use cases for our

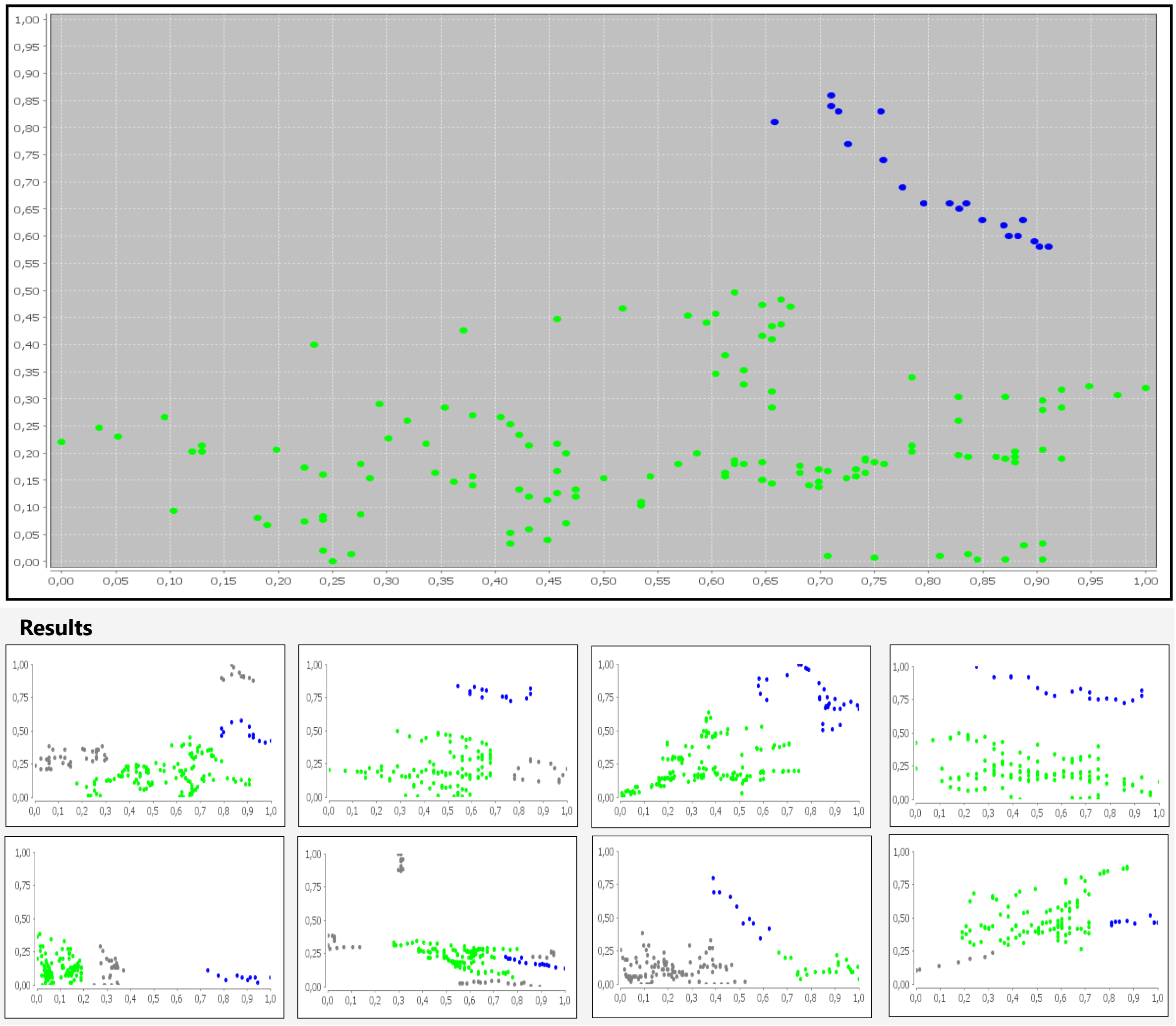


Figure 3.15: Retrieval example by using two motifs as templates. In this example-based search, we are looking for scatter plots that contain one bigger widespread motif located at the bottom (green) and one motif with a negative correlation at the upper right corner.

retrieval systems. We have chosen this way to demonstrate the usability of our approaches, since a performance evaluation will be hard or even impossible to achieve without ground truth. For sketch-based search, this would mean that we need a sketch classification for each pattern to measure the accuracy of a sketched query. Furthermore, user sketches and similarity perceptions for a given pattern can vary significantly from user to user. An interesting direction for future research may be the investigation in ground truth data for sketch-based search approaches. The user study by Bertini et al. [145] that aims at understanding how human judge on scatter plot similarity is an important step towards this goal. For instance, one could use the extracted set of scatter plot representatives from this preliminary work and conduct a study of the various sketch options for the given patterns.

Both proposed approaches, the sketch-based search as well as the example-based search use image descriptors to find matching results. The image descriptors, density and edge histogram descriptors, were developed during my masters thesis [172] and showed good results for retrieving global scatter plot patterns. In this work, we adapted the search functionality for local patterns by integrating a local pattern segmentation and clustering step to generate a dictionary of local motifs (see Section 3.3). It has been shown that this approach can be applied to local pattern retrieval and achieved appropriate search results for most patterns of the demonstrated data set. Furthermore, we identified the pros and cons of both methods when searching specific cases of scatter plot patterns. We noticed that the density descriptor outperforms the edge histogram descriptor (EHD) in retrieving patterns that contain large dense point distributions. This follows from the fact that the preprocessing steps of the EHD, i.e., blurring process and edge detection, may produce distorting artifacts of isolated points in the overall shape. On the other hand, the EHD achieves better results in finding patterns that consist of stringy line structures, since the structure can be well preserved by edge direction features. One idea to improve retrieval results would be to create a mixed descriptor approach that combines the strength of both approaches. For instance, additional image segments may be considered for computing visual features for certain regions of the pattern. Like in the approach by Park et al. [146] one could extend the feature vector with additional features from grouped histogram bins, e.g., by aggregating vertical, horizontal or corner subcells. Thus, patterns will be weighted higher than contain local structures connected over several regions. Please note that our search approach can also be replaced by other descriptors like scagnostics or regression models. However, for a sketch-based search, the descriptor approaches must be capable to recognize similarities between user sketch and target views. Regression models may improve the retrieval results for point distributions that represent a mathematical function but may not be the best choice to find complex shapes. In order to evaluate descriptor approaches, a user study could be conducted to determine which descriptors can be used for what kind of search.

Furthermore, it is important to note that also other factors like grid size, pattern segmentation, may influence the retrieval results. We experiment with different grid sizes and expand the grid size up to 64×64 cells in order to find out whether larger grid sizes will improve retrieval results. Depending on the pattern, we have noticed slight improvements, though increased computational costs. For example, by using the EHD a 4×4 grid results in an 80-dimensional vector whereas a 64×64 will result in a vector with 20.480 dimensions (64 × 64 × 5). As segmentation, we decided to use a minimum spanning tree clustering, since it produces similar structures in the constellation of connection pairs of points as humans [49]. Also, this step could be implemented differently.

In our current experiments, we noticed that with the increase of dimensions, the number of local patterns increases rapidly. This can affect the motif dictionary generation as well as the search for sketched patterns. Thus, sketching precise patterns and finding matching counterparts in a large collection may well be a difficult task. To this end, guidance features could be integrated to support users in sketching more accurate search queries. For instance, a shadow draw approach, like in [114], could be used to support the sketching process by showing template motifs as a shadow in the background (see Chapter 7). Besides that, this work can be extended in many different directions. Mixture model analysis may capture the notion of local patterns within scatter plots, and be a basis for visual query composition. We also want to investigate scalable visual representations for comparing sets of scatter plots. Glyph-based approaches could be interesting to this end. And finally, we want to investigate other application possibilities. Our search methods could be useful for screening and searching in the output of subspace projection and clustering algorithms, helping to narrow down the typically rich output of these methods.

3.7 Conclusion

We presented two approaches for search-oriented visual exploration of local scatter plot patterns. The proposed approaches are based on a visual search pipeline that includes scatter plot segmentation, feature extraction and motif dictionary creation. By using these approaches, we support the exploration process for locally interesting areas in scatter plots and bridge the gap between motif exploration and global scatter plot search. We adapted the minimum spanning tree algorithm for a non-parametric segmentation approach to extract local patterns form scatter plots. Image descriptors are used to compare visual queries with the extracted pattern from the dictionary. Appropriate search interfaces allow analysts to create visual queries by sketching desired patterns or composing motifs from the dictionary. We demonstrated the search approaches by using several well-known data sets and show intermediate results of our prototype systems. We showed that both methods can be used to find useful information with regard to local patterns. The proposed approaches are first steps into the direction of visual search techniques in large scatter plot spaces, and we have discussed a range of extensions to be done in future work.

Chapter 4

Visual Search for Movement Data

Contents

In the previous chapter, I demonstrated the benefits of using visual search approaches explicitly for the exploration and retrieval of scatter plot data. At this point, I would like to point out that high dimensional data can be represented by various visualization techniques and that sketch-based searches can be applied to other visualization techniques too. To achieve that, suitable features need to be extracted that can be mapped to an abstract user sketch as input query. However, currently sketch-based search is mostly applied in image, video and 3D object retrieval domain, and receives only little attention in the visual analytics domain. Besides scatter plot retrieval, it is also used for time series [86, 137, 215], graphs [206], geospatial data [47, 55, 95].

Recently, sports analytics has turned into an important research area of visual analytics, which may also profit from visual search systems, since it often has to deal with large and complex movement data. Soccer is a very popular and tactical game, which also attracted great attention in the last few years. However, the search for complex game movements is a very crucial and challenging task. In this chapter, I present a system for searching trajectory data in soccer matches by means of an interactive search interface that enables the user to sketch a situation of interest. Furthermore, I apply a domain specific prefiltering process to extract a set of local movement segments, which are similar to a given sketch. The approach comprises single-trajectory, multi-trajectory, and event-specific search functions based on two different similarity measures. To demonstrate the usefulness of the approach, I define a domain specific task analysis and conduct a case study together with a domain expert from FC Bayern München by investigating a real-world soccer match. Finally, I show that a multi-trajectory search in combination with event-specific filtering is needed to describe and retrieve complex moves in soccer matches.

This chapter is based on:

[177] **Visual-Interactive Search for Soccer Trajectories to Identify Interesting Game Situations,** L. Shao, D. Sacha, B. Neldner, M. Stein and T. Schreck. *Electronic Imaging Conference on Visualization and Data Analysis, 2016.*

4.1 Introduction

Nowadays, most mobile devices, such as smartphones, smartwatches, or tablets, are supporting Global Positioning System (GPS) tracking functions, and enable the possibility to let users log their activities during sport or leisure time. Often, spatial movement data is captured, giving rise to trajectory data. Due to the rapid development of GPS tracking devices and the continuous growth of data repositories that provide such trajectories, more and more application areas in geospatial data analysis emerge. For example, one upcoming research field is trajectory search for sports analytics. In recent years, there has been increasing interest in methods for analyzing statistics and tactical game plays in sports. Therefore, players get equipped with GPS sensors or are tracked by high-resolution cameras to extract the raw movement data. Also, plenty of approaches exist, which analyze whole matches, individual player statistics, or provide hypothesis prospects for sports, such as tennis, basketball or soccer [68, 97, 152].

In the field of soccer analytics, most of the existing approaches focus on statistical feature extraction, like speed, acceleration or running distance, which are visualized by a variety of techniques to discover insights in matches [97]. The drawback of these approaches is that they only utilize the statistical information and manually annotated events (e.g., goals, free kicks, fouls, etc.) to analyze interesting game situations. However, in a soccer game, there may exist many important match phases that are time-consuming and expensive for manual annotations. One unexplored research field is the direct trajectory search for untagged movement patterns, such as individual player movements, tactical team movements or rehearsed ball passing paths in matches. The search for such trajectories may help coaches, managers, scouts, and other decision makers to find crucial game situations, and thus analyze the performance of an individual player or the whole team. The search complexity of these large movement data is challenging and also referring to general geospatial analysis problems, since we are using interdependent trajectories on a small limited area. Soccer is a very tactical and strategic game, and thus the search space can quickly become overcrowded by 23 simultaneously interacting trajectories of players and the ball.

In this chapter, we present an approach for sketch-based visual search and exploration of soccer trajectories to discover crucial game situations based on single-player, multi-player and event-specific trajectories. Therefore, we apply two different similarity search approaches based on spatio-temporal point distributions and movement directions of a trajectory. For the purpose of comparing locally similar trajectory segments, we introduce a domain-specific filtering process, which extracts relevant trajectory segments. Our search interface is inspired by tactical drawings from soccer coaches and should reflect a search modality closely related to coaching and analysis practice in this domain. Furthermore, we design a task analysis for movement search in the context of soccer analytics and specify the needs for an easy and intuitive search interface. We demonstrate the usefulness of our approach by conducting a case study together with a professional youth coach of one of the top soccer clubs, FC Bayern München.

4.2 Related Work

In the following, we discuss a selection of related work in the context of our approach. Three main research areas are sketch-based search, movement analysis and general visual analytics of soccer data.

4.2.1 Movement Analysis

Due to the growth of mobile computing and GPS supported devices, the analysis of trajectories has become an interesting task. For instance, by using visual analytics tools and mining trajectories of animals, it is possible to discover movement and migration patterns of certain groups of animals [182, 187]. In [170], textual descriptions are combined with trajectory data of a traveler for trip planning and recommendations. Nowadays, many applications for visual analytics of movement data exist, including the analysis of vessel movements [218], traffic jams [211], or sports data

[115, 223, 224]. Moreover, a survey of existing visual analytics approaches concerning movement data is given in [6, 8].

Important data analysis methods in this area include the similarity search, segmentation, or clustering of trajectories. Recent research on trajectory search approaches focus on optimizing the retrieval process, which is an interesting problem in itself, but beyond the scope of this paper.

4.2.2 Soccer Analytics

Recently, knowledge discovery methods have successfully been applied to analyze large amounts of soccer and sports data. A recent approach was introduced by Bialkowski et al. [24], in which the authors detect player formations within a whole match including potential positional interchange. The authors approach this by applying a role-based representation that dynamically updates each player's relative role at each frame. Another recent work on analysis in sport has been presented by Lucey et al. [129], where a novel technique is presented to estimate the probability of a successful chance in soccer. The authors discovered several factors, which contribute to the probability of a chance. These factors were determined and evaluated by analyzing a whole season of player and ball tracking data. In another related work by Janetzko et al. [97], a system for analyzing high-frequency position-based soccer data at various levels of detail covering player and event-based analytical views is presented. Among other things, the authors performed single and multiple player analysis, for example, for the detection of player phases. Furthermore, constellations and formations were analyzed to, for example, evaluate the performance of the back four formation. Another system for the analysis of soccer data has been introduced by Perin et al. [148]. They developed a tool combining different perspectives on soccer data after segmenting the data into meaningful units. The segmented units can be analyzed in different visualizations. Other presented systems focus on the detection of events of interest. For instance, Rathod et al. [156] retrieve these events by a fuzzy inference system with an input of features and grass percentages extracted from key frames. Xiong et al. [221] extend this idea by extracting audio sources, such as applause. Other mentionable approaches for analyzing soccer players and teams have been presented by Gudmundsson et al. [70, 71]. They presented several approaches based on position data to extract basic events, such as kick-offs, corner kicks or throw-ins, and applied a cluster analysis on single player's subtrajectories to identify frequent movements during the match.

4.3 Sketch-based Search for Trajectories in Soccer Data

In this section, we describe the relevant search space and tasks for the soccer domain. In addition, this section defines the scope of the work presented in this paper.

4.3.1 Soccer Search Space

In order to describe the sketch-based search space, we make use of the types of movement analysis tasks proposed by Andrienko et al. [6]. Movement analysis can be described using four different foci: Movers M, spatial events E, space S, and time T. With that respect, in our soccer case movers are all moving (spatio-temporal) objects on the soccer pitch ($M \in \{Player, Ball\}$), spatial events can be described as movement or rule-based events ($E \in \{Movement, Rule\}$), space can be referred to moving areas of interest, such as empty/free or occupied areas, or fixed areas such as dangerous areas near a goal ($S \in \{Moving, Fixed\}$). Finally, time is related to the temporal constraints and units are given by a soccer game (e.g., half-time) or the duration of soccer moves ($T \in \{Unit, Duration\}$). In addition, especially in soccer, there are several further dimensions that can be combined with the focal set ($\{M, E, S, T\}$) for searching. To describe the soccer search space as a whole we propose the following dimensions:

***D1*-What to Search:** All elements of the focal set ($\{M, E, S, T\}$) and their combination and the attributes of the movement (e.g., speed).

***D2*-Invariances:** Having specified a spatial query this can be *translated*, *scaled*, or *rotated* in order to search for similar elements.

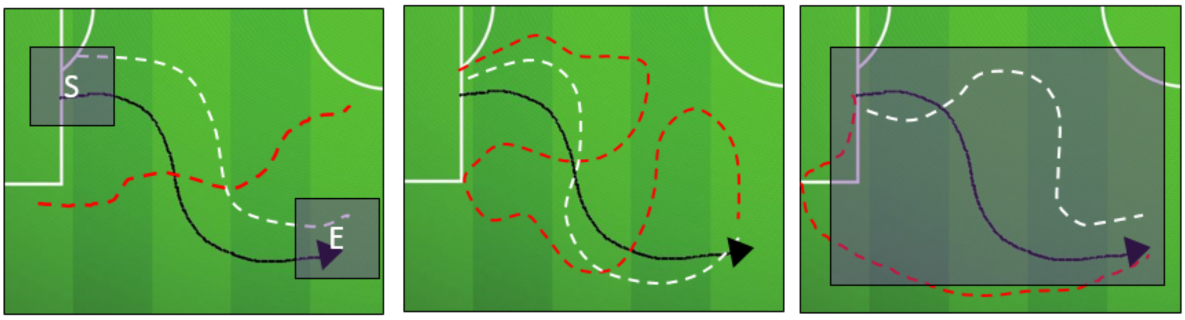
***D3*-Filtering/Constraints:** Filtering can be done by adding constraints to all elements of the focal set. In soccer, relevant constraints are temporal (e.g., only first half), event-specific (e.g., only shot-events), or object-attribute-specific (e.g., tactical position of a player or only movements including high speed values), and spatial (e.g., only in a specific region). Furthermore, these constraints can be applied for specific or all elements (specific vs. global).

***D4*-Cardinality:** Furthermore, we distinguish if the analyst is seeking for one or several elements as well as their temporal order. Synchronous searching refers to a search where all elements occur together at the same time, whereas asynchronous or sequence searching considers time shifts between the objects.

4.3.2 Our Scope

Search Space: Our approach is inspired by board sketches typically known from soccer coaches or analysts. Therefore, our approach enables the analyst to intuitively define a query object directly on the soccer pitch by applying a sketch of the desired trajectory at the desired place. That is why our search space is limited to moving objects M at first (*D1*). In this case, sketching is the creation of a spatial query object [7]. Furthermore, our similarity search is designed to find the exact position, rotation, and scaling of the trajectory (*D2*). The filtering and constraints can be added to the similarity search for individual trajectories by specifying the occurrence of specific movers, events or time intervals (*D3*). Finally, the similarity search can iteratively be refined by adding further sketched trajectories. The similarity search for these multiple spatial query objects is designed for synchronous occurrence that identifies situations with all the desired movements happening within a short time span (*D4*).

Tasks: Our approach is designed to support typical *relational-seeking tasks* where "items that are related in a specified way need to be detected" [7]. In our case, the system presents the user trajectories that are "equal" or "similar" to the query object. Furthermore, our approach enables the analyst to inspect the result set and therefore to perform further *lookup* or *comparison* tasks. Our design is guided to support the following two higher level analysis tasks: (1) searching a specific movement/situation that happened at a specific time, and (2) identifying, comparing and relating a set of similar movements. Concrete examples for these higher level tasks are given in the next subsection. In order to support these tasks, we propose the following analysis workflow. First, the analyst creates a query object (trajectory sketch). Then the analyst may inspect the result set and to further refine the analysis he or she can add additional filters, constraints or trajectory sketches and even remove old ones. This way every search can be refined, improved and narrowed down to the specific task that is needed. Furthermore, the analyst will be able to switch between two different distance functions that are adapted for our two analysis tasks (see Section 5). Finally, in each step the analyst can inspect the situation of interest with additional visualizations and animation (by showing all players, ball trajectory, heatmaps of player movements, etc.).



(a) Similar start and end position.

(b) Similar trajectory length.

(c) Similar marginal restriction.

Figure 4.1: Demonstration of our preprocessing steps to filter out dissimilar trajectory segments (illustrated in red). First, we only consider trajectory segments, which are running through a similar start and end area of the user sketch (a); then we compare the length of user sketch and remaining candidates (b) and finally, we are using a bounding box to check for deviations within their path (c). The black arrows illustrate the user sketches and the trajectories that fulfill the filtering constraints are colored in white.

4.3.3 Domain Tasks

This subsection lists some soccer search tasks that emerged during our discussions, observations and requirements analysis with the domain expert. In most cases, the first step for an analyst is to sketch the ball trajectory (or a player trajectory where the player is in ball possession) because the ball movement determines all the other movements. As a second step further filters and trajectories are added to characterize specific situations. Analysts prefer to search for attack movements instead of defense situations, as the domain expert explains, in attacks the trajectories and players' movements are characteristic and of interest, whereas in defense behavior spaces and groupings are more crucial. That is why our approach focuses on searching for attack situations. We are able to distinguish two kinds of attack situations (where a specific team is in ball possession): *Middle*, or *Side* attacks.

Middle Attacks: Middle attacks are situations in which a player in the middle receives the ball in order to control it (typically done by a robust forward player) or situations where the ball is passed in the back of the opponents' defense line (also known as a "through ball").

Side Attacks: Side attacks can be distinguished between attacks where the player tries to reach the baseline to pass (or cross) the ball in the penalty area of the opponents or attacks where the players try to reach the goal directly from the wings. A third class of side attacks covers crosses that are impacted from the half court towards the goal. The identification of these attack types determines the domain tasks. Further, the domain expert describes *Preparation Phases* for attacks.

Preparation Phases: Preparation phases are situations in which the team in ball possession traverses and switches the ball from one side to another, zig-zagging towards the opponents' goal, forcing the opposing team to open up and produce free spaces. However, the preparation is an optional phase that can be included in an attack followed by a finish (e.g., shot on goal).

A concrete domain task for the identification of a specific situation could be that the coach wants to find an attack over the right hand side at the beginning of the match where the side player reaches the baseline and passes the ball to his forward who then performs a direct shot on target. Another concrete domain task could be to identify how many attacks have been performed via the left hand side and thus derive new tactics or analyze the strengths and weaknesses of the opponent or rather own team. This query could be refined by adding sketches of the forwards that run into the penalty area in order to receive a cross. Further examples and solutions to solve these tasks are illustrated in Section 5.

4.4 Visual Exploration of Soccer Movements

In this section, we describe our sketch-based search techniques to find similar movements and provide an overview of the developed components of our system. The focus of our work is on the development of an easy and intuitive search technique for finding complex movements in soccer matches. Our prototype system for discovering interesting game situations is depicted in Figure 4.2.

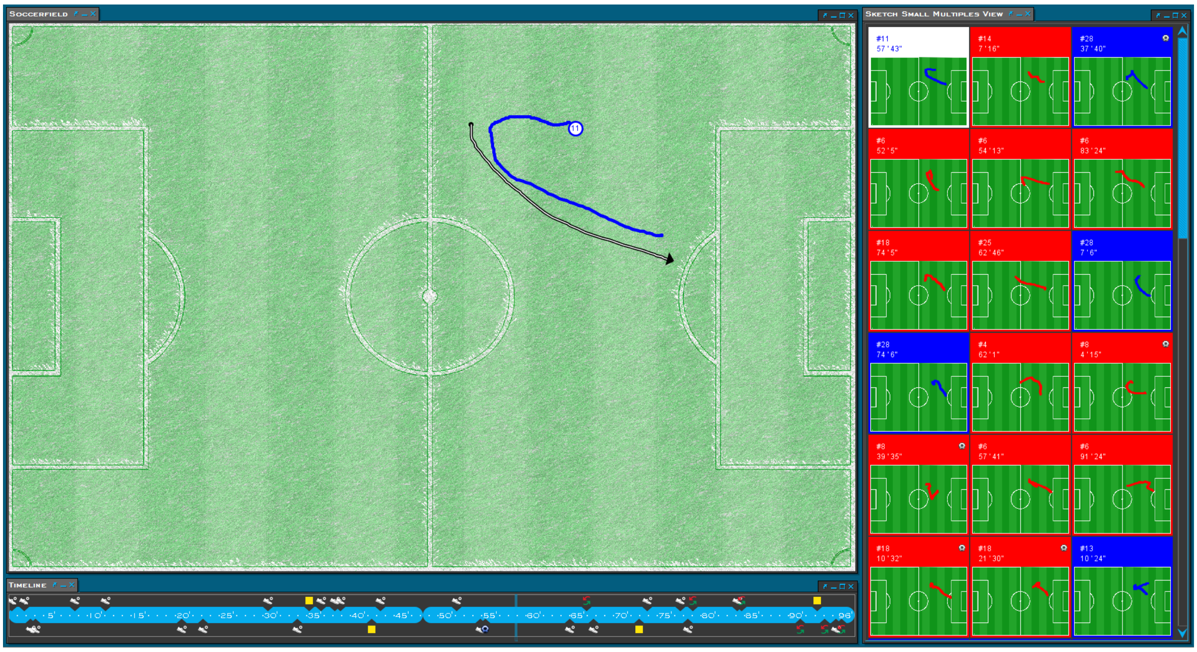


Figure 4.2: Our implemented prototype showing a single-trajectory search by applying the spatial distribution approach. In this example, we are searching for movement patterns that start from the outer midfield and run into the direction of right penalty area. The best matching results are shown in the small multiple view on the right hand side. By clicking on a thumbnail (highlighted in white), the particular trajectory will be directly displayed on the soccer pitch. Additional information about the scene like jersey number, player name, time or trajectory type (displayed in different colors) are shown in the small multiple thumbnail.

4.4.1 Movement Search

Our approach to search for movement patterns is based on a three-step process including trajectory filtering, feature extraction and similarity measure. Since a soccer match lasts at least 90 minutes and involves 23 individual trajectories of $M \in \{Player, Ball\}$, the search space can quickly become too large for manual search and hinder the search process for a particular movement pattern. Furthermore, a trajectory over 90 minutes can be very long and has to be investigated in terms of their local similarity. We therefore use a preprocessing step to eliminate trajectories that have no locally similar movements to a given user search based on a coarse filtering step. The preprocessing step comprises three further filter constraints, which are depicted in Figure 4.1. In the first step, all trajectories will be checked if they contain a local path that runs through a similar start and end position as the user sketch. After that, the path lengths of all remaining trajectory segments are compared to the length of the drawn sketch. All segments which exceed or do not fulfill a certain threshold length will be eliminated. The last filter constraint refers to the overall spatial similarity of the trajectory area. If a trajectory segment exceeds a marginal area of the user sketch it will also be filtered out. These filtering steps utilize the original geospatial positions on the soccer pitch and have the benefit that the computation is very fast and does not need any trajectory segmentation procedures in advance. It quickly returns a rough set of similar local paths for the further search process. Another advantage against a pre-segmentation is that our approach is capable of searching trajectories without interruptions and thus even player movements during breaks (e.g., throw-in) can be found. The parameter settings for the three filtering thresholds influence the resulting number of similar trajectories and can be adjusted by the user. We propose to use at least a tolerance range of 20% on the user sketch length and a range of 10% on the sketch location (bounding boxes), to find enough candidates for the similarity ranking.

In order to identify the most similar movement patterns and to rank the result set, we are using a feature-based similarity function. Hence, we need to describe the characteristics of the not filtered segments as well as the user sketch by a suitable feature vector. Therefore, we provide two different descriptors based on the spatial distribution and the movement directions of a trajectory, each tailored to one of our two analysis tasks from Section 3. The descriptor for measuring the spatial distribution is used to accomplish task 1 - searching a specific movement/situation that happened in the game. For this task, we assume that the analyst watched a match and wants to retrieve a specific situation. In this case, the spatial orientation and the trajectory path are the most important feature that describes the happened scene (e.g., corner kick). Our descriptor uses a flexible grid approach, which adapts the grid size according to the user sketch size on the soccer pitch and thus limits the space of interest ($S \in \{Moving, Fixed\}$). This ensures that small sketches (e.g., a dribble action on a specific area) will be considered more precisely than a rough sketch that runs over the whole soccer pitch (e.g., a long diagonal pass). The benefit of this flexible grid approach is that imprecise sketches on the right place are sufficient to detect an existing situation that happened in the game. By default, we superimpose an 8×8 grid on the user sketch and apply the same grid size to the comparing trajectories. Each cell corresponds to a unique index and the feature vector is composed of the particular indices where the trajectories pass through. Figure 4.3 demonstrates the composition of a feature vector.

The second descriptor focuses on the characterization of the general structure and is therefore better suited for our second analysis task - identifying and comparing similar movements. In contrast to the first descriptor, our aim here is to ensure that similar trajectories with slight shifts are rated significantly better than dissimilar trajectories, which pass the same cells. This is especially relevant when it comes to complex analysis tasks that require more generic queries for searching typical movements or novel tactics of the opponents. To this end, we defined eight different direction types (0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°) to describe the structure of a given trajectory. The direction types are encoded as numerical values and the feature vector represents the direction types from start to the end position of the user sketch. This has the advantage that we can prioritize the similarity search by the characteristics of a movement ($M \in \{Player, Ball\}$). Finally, the last step is to compute the distance between the feature vectors to rank the trajectories. Here we use an edit distance metric [121] for measuring the minimum

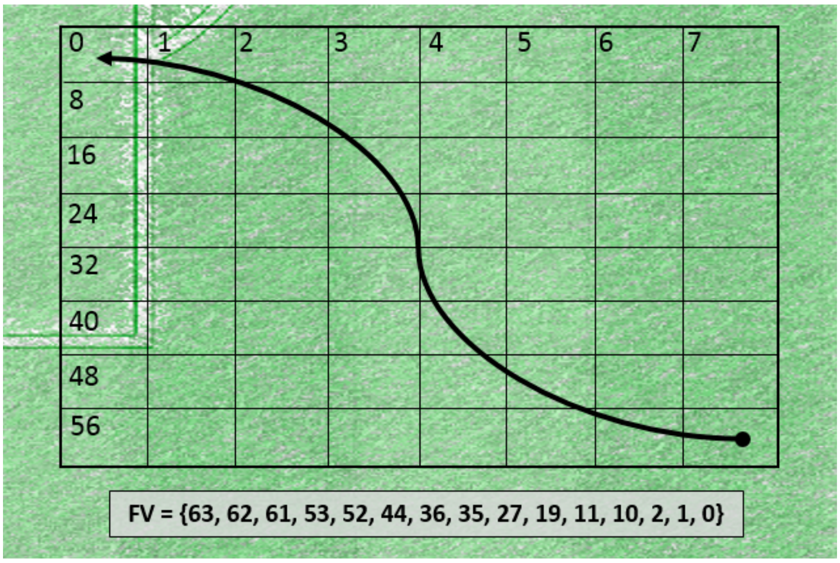

0
1
2
3
4
5
6
7
8
16
24
32
40
48
56
FV = {63, 62, 61, 53, 52, 44, 36, 35, 27, 19, 11, 10, 2, 1, 0}

Figure 4.3: Composition of a feature vector by using our grid-based approach (spatial distribution). The trajectory starts at the bottom right corner (index 63) and traverses the grid towards the upper left corner (index 0).

distance between the user sketch and the trajectory candidates.

This applies to a one-to-one movement search (single-trajectory), but in the event of a multi-trajectory search, we perform individual queries for each drawn sketch and aggregate the computed distances for the overall similarity. Furthermore, we verify whether the matched trajectories correspond to a joint action with respect to the time. This means that we are only searching for movements that took place during the same period of time. At this point, we are considering two different temporal constraints to define synchronously occurring movements. The analyst can choose between a chronological sequence and a simultaneous occurrence of trajectories (D4). For sketching general attack situations like middle attacks or side attacks, we recommend to use the search for simultaneous occurring trajectories, which is more flexible and only considers whether the found movements occur within a short time span. In contrast, to fulfill the constraints of the chronological sequence, the found movements have to occur within a short time span and also in the same order as the drawn trajectories. Thus, this search setting can be used to sketch situations including aftereffects (e.g., a counterattack after a failed side attack). Eventually, we aggregate the distances of the trajectories that belong together and calculate the overall similarity for ranking.

A last important aspect that needs to be considered is the temporal constraint of the players positioning, which depends on the two half-times of the game. By that, we mean that players will change their positions after the half-time break, and thus makes the search for spatial queries of individual players more challenging. To perform spatial- and player-dependent queries, we provide a position mirroring function that maps the player and ball of the second half formation to their origin (first half positioning). Depending on the analysis task, this feature can enabled or disabled.

4.4.2 Interaction

For searching soccer movement trajectories, we provide a sketch-based interface that allows analysts to sketch their queries. Since soccer coaches often tend to sketch tactical movements on boards, we utilize an interactive soccer pitch that enables users to directly sketch movements of $M \in \{Player, Ball\}$ on the field (see Figure 4.2). This helps the user to precisely draw spatial movement queries such as middle attacks or side attacks. Alternatively, a query can also be created by sketching two rectangular areas that describe the start and end position of a trajectory, as shown in Figure 4.8 and 4.9. The benefit of this alternative trajectory search is that more general queries may be drawn and a greater proportion of deviating but also similar directed trajectories may be found. The rectangular areas will replace the spatial constraints of start and end area in our preprocessing steps (cf. Section 4.1) and thus increase the number of potentially interesting movements. The original path of the query runs from the center to the center of both drawn rectangles and will be added automatically to the search. All sketching interactions are implemented with mouse-events and thus allow query drawings by mouse, digital pens or even fingers on touch tables.

Further filters are used to explore and navigate the result set (D3). To indicate a drawn sketch as a specific movement type of $M \in \{Player, Ball\}$, we equipped the search interface with a radial layout menu, which shortens the distance and time to navigate to an item [160]. By means of this menu users can filter for trajectories of a particular team (home, away), player groupings (defender, center/midfielder, right/left-winger, striker), or the ball. Figure 4.4 shows the radial menu and a chosen filter constraint on ball trajectories. In order to select a certain player, we also provide an tabular view of all players in a additional option panel. For an event-specific search of $E \in \{Movement, Rule\}$ the query can also be connected to a particular event, such as shot, corner kick, free kick, cross, or chance. This ensures that the found scenes are initiated or produce such an event in a short time. For instance, drawn side attacks may be filtered for shot events to receive only the completed attempts. Moreover, by selecting a time range $T \in \{Unit, Duration\}$ on the timeline, which is placed under the soccer pitch (see Figure 4.2), results can also be restricted by time. Consequently, analysts may search for situations that happened in the second half.

4.4.3 Visual Representation

When a query has been performed, the most similar trajectory will be displayed on the soccer pitch next to the user sketch (as shown in Figure 4.2). The found trajectories are colored according to the trajectory type. Red trajectories refer to movements of the home team, whereas blue trajectories refer to the away team and gray trajectories to the ball. We indicate the movement direction by showing a circular starting position, which also reveals additional information about the trajectory, like jersey number, or a ball label.

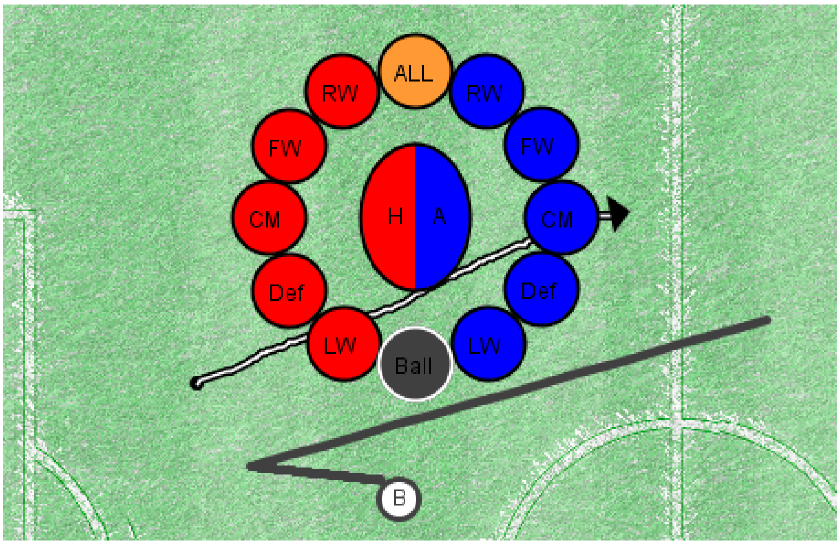


Figure 4.4: Our radial menu for filtering trajectories based on team, player groupings and the ball. By selecting a particular filter criterion the result set gets updated and the best ranked trajectory will be displayed on the soccer pitch. This view demonstrates the selection of the ball trajectory.

The temporal information ($T \in \{Unit, Duration\}$) of the found scene can also be highlighted as interval on the timeline (illustrated in Figure 4.2). This allows analysts to jump to the exact game phase and analyze the found situation. For browsing the remaining result set a small multiple view is used that represents the same information for all individual trajectories. The visual encoding of the thumbnails in the small multiple view is as follows: The background and trajectory color corresponds to the trajectory type (home team, away team or ball) and additional information like player name, jersey number and time are shown on top. In case of a multi-trajectory search, the background color encodes the percentage of ball possession of the respective time interval, since several sketches of different teams could be drawn. Our small multiple view is shown on the right hand side of Figure 4.2. By clicking on a particular thumbnail, the trajectories will be enlarged, displayed on the soccer pitch and the selected time interval will be updated. This helps the user to accomplish our tasks from Section 3.

4.5 Use Cases

In this section, we demonstrate the usefulness of our search system by performing our domain tasks that we defined in Section 3. To evaluate both of our descriptors, we structured the domain tasks into two similarity areas: (1) searching for the exact spatial position of a movement and (2) searching for the main directions of the movement. The search tasks for the spatial distribution descriptor focuses on searching *specific situations* (1) and the direction-based descriptor on identifying/comparing *similar movements* (2) in a match.

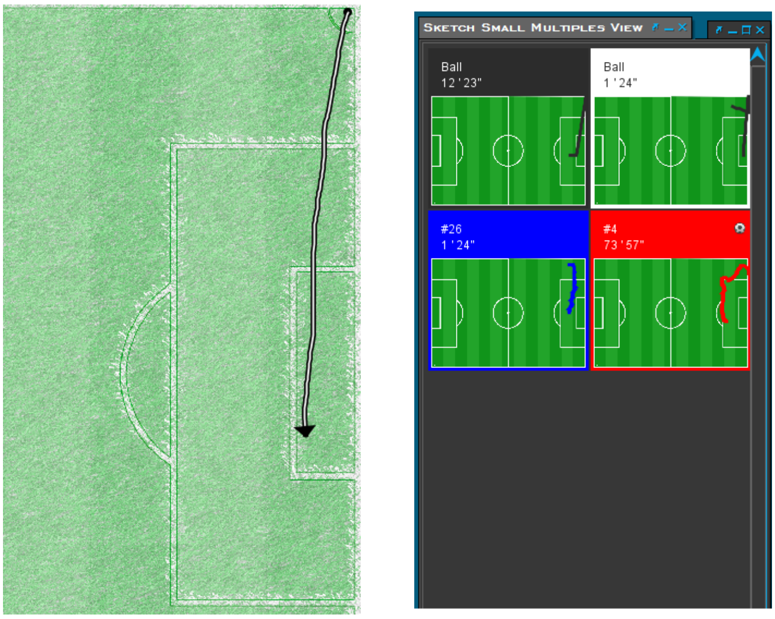
Therefore, we used professional soccer match data provided within a collaboration with the sports analytics provider Prozone[1]. The data set is not publicly available and has to be anonymized as it was a professional game. For each player of both teams, two-dimensional position data are available with a temporal resolution of 100 milliseconds. The data are enhanced by manually user annotated events containing information about the position, time, involved players and event-specific details. We designed a qualitative case study to evaluate our approach together with a soccer expert. The invited expert has been an active soccer player for 24 years and has been working as a coach for 10 years; he is currently employed by the German soccer club FC Bayern München. The evaluation was performed on a notebook, connected to a 24 inch Full HD monitor, where the system was running. The program was executed in full-screen mode. The domain expert was asked to express his thoughts and opinions while using the system, according to the *Thinking Aloud* [28] method. During the evaluation, we took notes of comments and own impressions for subsequent analysis.

4.5.1 Task 1: Specific Situations

A very important task in soccer analysis is to retrieve situations of interest that are already known to the user. This might be important when the analyst identified an interesting situation during the match and later wants to re-watch it to gain knowledge. Hence it is important, to retrieve movement patterns based on the exact spatial position on the soccer pitch. The task in this section is designed to retrieve a match situation with our system after watching the video recording. Therefore, we investigated a professional soccer match from a first-class league in Europe. We selected a number of match highlights from a video summary and tried to retrieve the respective scene together with our domain expert. The selected scenes were:

Scene 1: Side Attack (Corner Kick)
The first highlight of the match was a corner kick in 1:30 minute of the game that also leads to the first opportunity for the home team (colored in red). As our domain expert recommend to start sketching with ball trajectories and then include additional movement trajectories to specify the query, we start the search by sketching a trajectory from the respective corner position into the penalty area. The sketch and corresponding results are shown in Figure 4.5. After the initial query, we found the only two corner kick situations (from the respective position) in the game and two player movements with similar trajectories. Since a total of only four trajectories have passed this way, all other trajectories were filtered out by our prefiltering conditions (cf. Section 4). The searched corner kick scene at 1:30 minute can be found in second place (highlighted in white - starting a little bit earlier). In the event of a larger result set the matched trajectories can also be restricted in terms of time (e.g. only first half) or moving object (e.g. only ball) to detect such set-plays (D3).

[1] http://www.prozonesports.com/



(a) Query for Scene 1.

(b) Result set of query 4 (a).

Figure 4.5: User sketch and the corresponding results for Scene 1. In total the search system found four similar trajectory paths including two corner kick situation. All found trajectories start at the upper right corner and go through the penalty area.

Scene 2: Middle Attack

The second scene depicts a more complex move from the blue team in 22:00 minute, which includes a rehearsed ball passing and multiple player shifts to create an opportunity on goal. The selected scene shows a counter attack that starts in the midfield of the soccer pitch by controlling and distributing the ball to perform the "through ball" (a pass in the back of the opponents' defense line). Therefore, the central player passes the ball to the wing player on the right hand side and runs to the right baseline in order to create a new "open man" (undefended player) and opens the space for a secondary striker at the same time. To retrieve such a scene, an advanced query with additional information about secondary trajectories and trajectory types are needed. We roughly sketched the whole ball passing path and a second running path of the striker that runs into the open space and has the opportunity on goal. To specify a trajectory type the sketches can be additionally assigned by our radial menu (see Figure 4.4).

Figure 4.6 illustrates the two drawn user sketches and the best matching results of our spatial distribution descriptors in a small multiple view. This time, we explicitly indicate the first drawn trajectory as ball trajectory (upper sketch) and the second (lower sketch) as any player trajectory of the away team (blue) to keep the results clear and reasonable. Without specifying the types of both trajectories the result set would contain more than hundreds of possible trajectory combinations that happened during the match. By means of our filter constraints, we guarantee that all matching results include a zigzag kind pattern of the ball trajectory and a straight

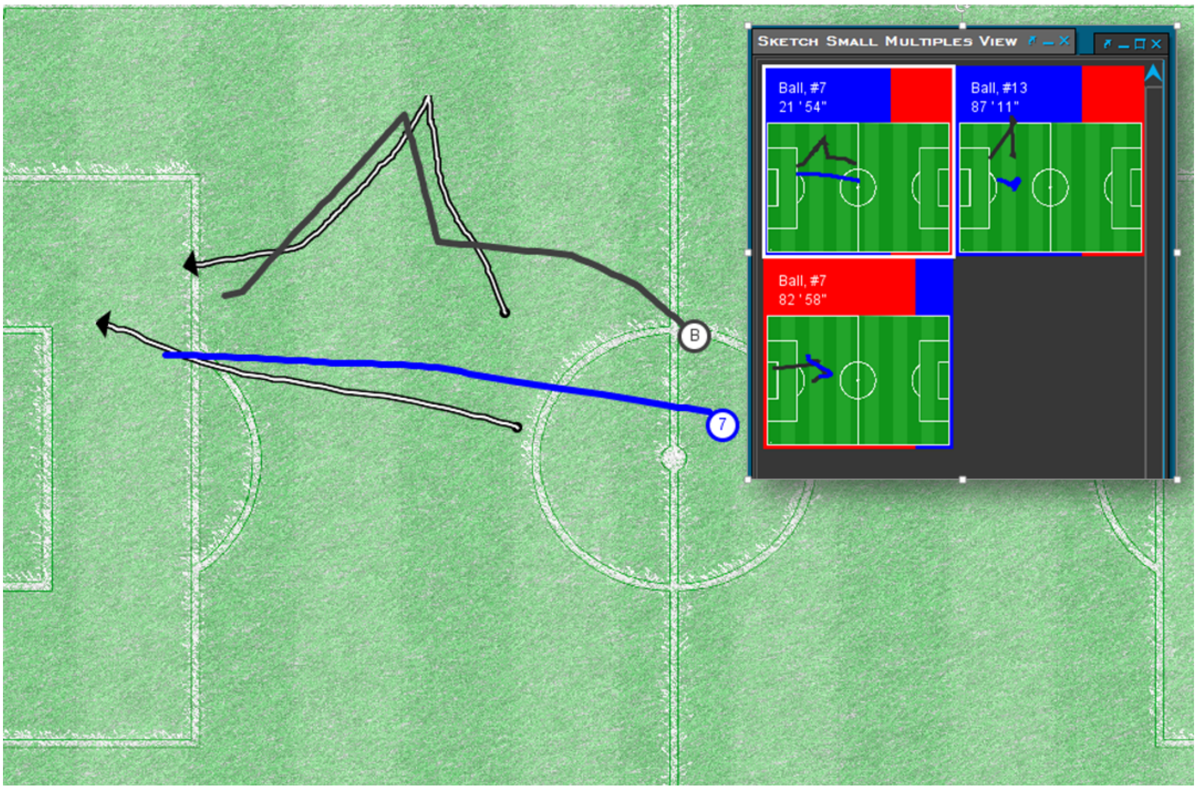
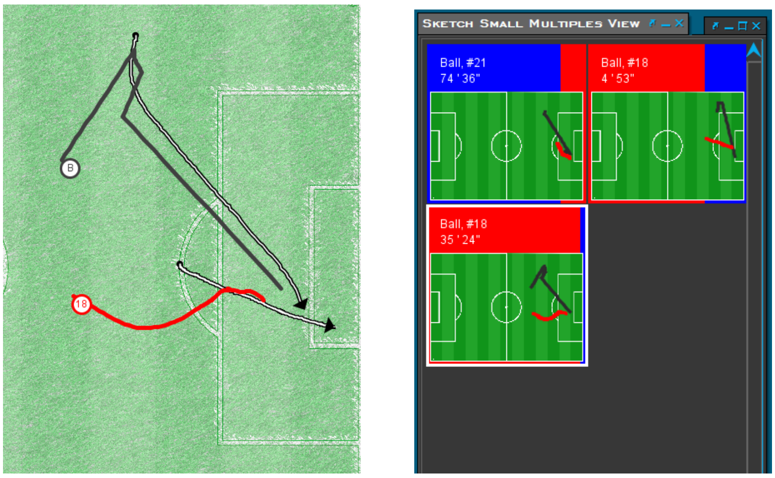


Figure 4.6: Illustration of a multi-trajectory query for Scene 2 including the best matching result (visualized on soccer pitch) and further results in the small multiple view. In order to rediscover the situation, we determined the first sketch (upper arrow) as ball trajectory and the sketch arranged below as player trajectory of the blue team (away team).

running path of any player of the away team that is located below. The result set reveals that such a situation has occurred three times during the match, namely at 21:54, 82:59 and 87:11 minute of the match. Moreover, we rediscover the trajectories of the desired scene and ranked the situation on first place.

Scene 3: Side Attack

The third and last scene that we want to retrieve is the last chance for the home team before the half-time break in 35:30 minute. In this scene a player of the red team is crossing the ball from the upper outside area into the penalty area while a teammate and striker is running into the box to complete the attack. To sketch this query, we again start with sketching the ball trajectory from corresponding position into the penalty area and include another trajectory that indicates the running path of a player of the home team. This time, we again found the desired scene (ranked on third place) and two other very similar situations with homogeneous movements of ball and player, shown in Figure 4.7. This might be a rehearsed movement of the team that occurs three time during the game, but on closer inspection the small multiple thumbnail reveals that the situation ranked on first place happened in the second half of the game and the ball



(a) Query for Scene 3.

(b) Result set of sketch 6 (a).

Figure 4.7: Multi-trajectory sketch and the corresponding results for Scene 3. The sketched trajectories are determined as ball (upper sketch) and player (lower sketch) trajectory (a). The background color of the small multiple thumbnails reveals additional information about the game phase (b).

possession of the blue team is more than 80% (background color) at this situation. Consequently, the situation ranked on first place is a defensive play of the home team, which involves similar movement patters like the attempted attacks at 4:53 minute and 35:24 minute.

4.5.2 Task 2: Similar Situations

Another task is to identify and explore several similar situations with respect to a given search pattern. This is important when an analyst wants to find out, for example, how upcoming opponents structure their offensive play. In this respect, it is also important to formalize a more generic query that covers typical movements and identifies novel tactics of the opponents, which are not yet known. Hence, for this task, we apply our position mirroring function that maps the playing direction of both teams into a uniform direction (i.e., no changeover after the half-time break) and include our spatial area sketches combined with the direction-based descriptor to make the search more flexible. For the experiment, we again investigated the same match from Task 1 together with our invited expert and let him answer a few predefined questions by using our tool. Our analysis questions were:

Question 1: Which defender of the away team participates in offensive situations at most?

By means of the mirroring function, the players of the away team are always playing from the right to the left side and spatial queries can be performed on both half-times simultaneously. To answer this question, the expert sketched two rectangular areas that describe the start and the end position of a trajectory. Moreover, he limited the search space by selecting only the defender

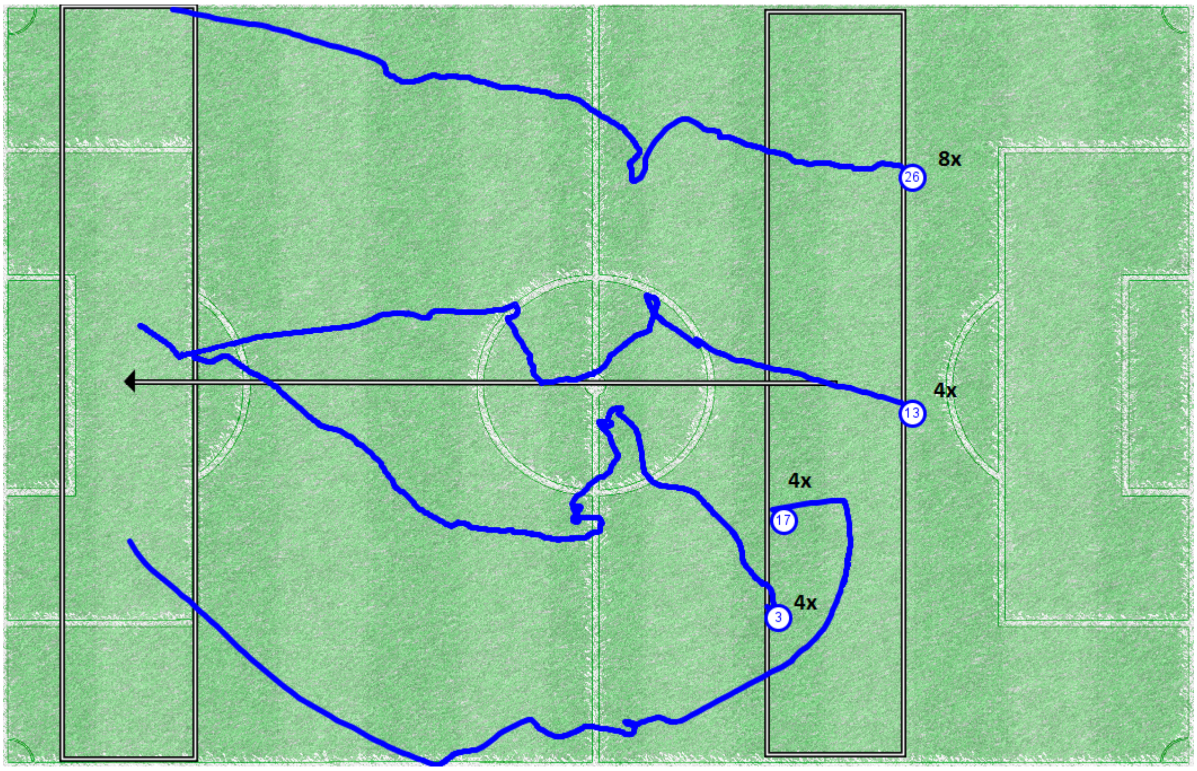


Figure 4.8: Overview of the offensive attempts of defenders from the away team. The two rectangular boxes determine the start and end position of the searched trajectories. This query reveals that the right defender participated eight times in offensive situations.

of the away teams in our radial menu. An illustration of the result is presented in Figure 4.8. We found in total twenty movements of the away team defenders that start in their own half of the court and run into the danger zone (near opponents goal). The three defenders with the jersey number 3, 13 and 17 participated four times in offensive situations whereat right defender with the jersey number 26 (the defender who puts in the most effort in offensive play) attempts eight times. Also interesting to consider are the simultaneous attempts of the defenders. For instance, in 1:56 minute all four defenders participated in offensive play, which also led to an opportunity for the away team.

Question 2: How often did the away team perform side attacks over the right hand side?

We assume that this question is addressed to side attacks where players try to reach the baseline and cross the ball into the penalty area. Figure 4.9 shows the suggested query from our domain expert and the final result set of found side attacks. After applying the shown query, our visual search system identifies 21 trajectories that have a similar main direction, and similar start- and end positions. However, the problem is that also defensive movements or movements without any ball possessions were found, which are not related to side attacks. To prevent such search results, we include another filtering constraint on the rule-based events ($E \in \{Movement, Rule\}$)

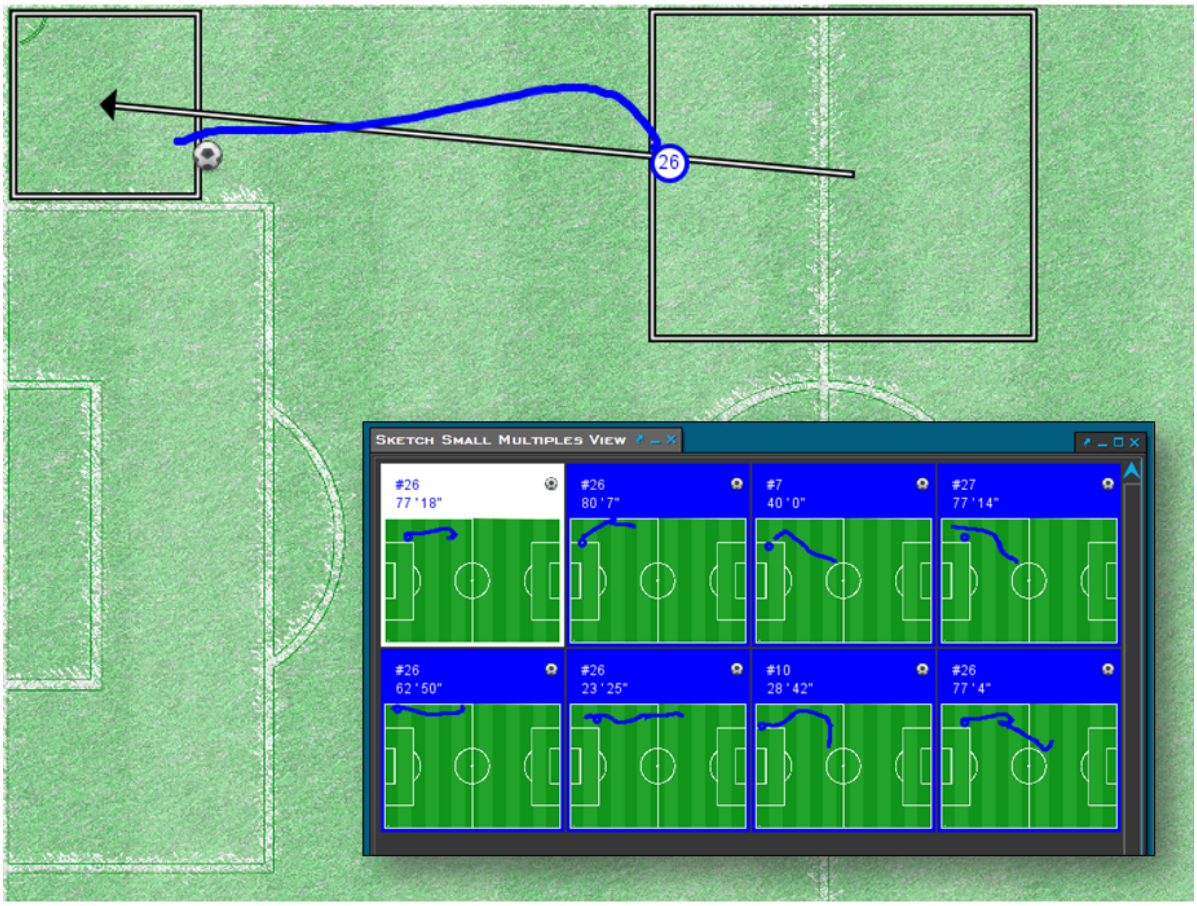


Figure 4.9: Spatial area sketch to identify the side attacks from the away team over the right hand side. By including the event information it only returns real attack situations that involves a cross.

to receive only movements, which contain a cross event. These results are marked with a ball symbol in the small multiple view and show the respective position on the soccer pitch where the cross happened. Finally, we found eight side attacks from the right hand side, as shown in Figure 4.9.

Question 3: How often has the right striker from the away team attacked over the left wing?

The right striker from the away team is a typical center forward player whose main function is to receive passes, win long balls and score goals. In the interest of receiving through balls these forward players have to continuously change their running paths and try to find open spaces at the right moment. Hence, it is appreciated when several forward players change their positions over time to create new attack situations and confuse the opponents' defenders. However, our analysis task is now to find out how many times he changed the sides and attacked from the left wing side. Therefore, our expert sketched a typical left wing attack pattern that runs from the left wing into the penalty area (curved movement on the border) and specified this as a movement of

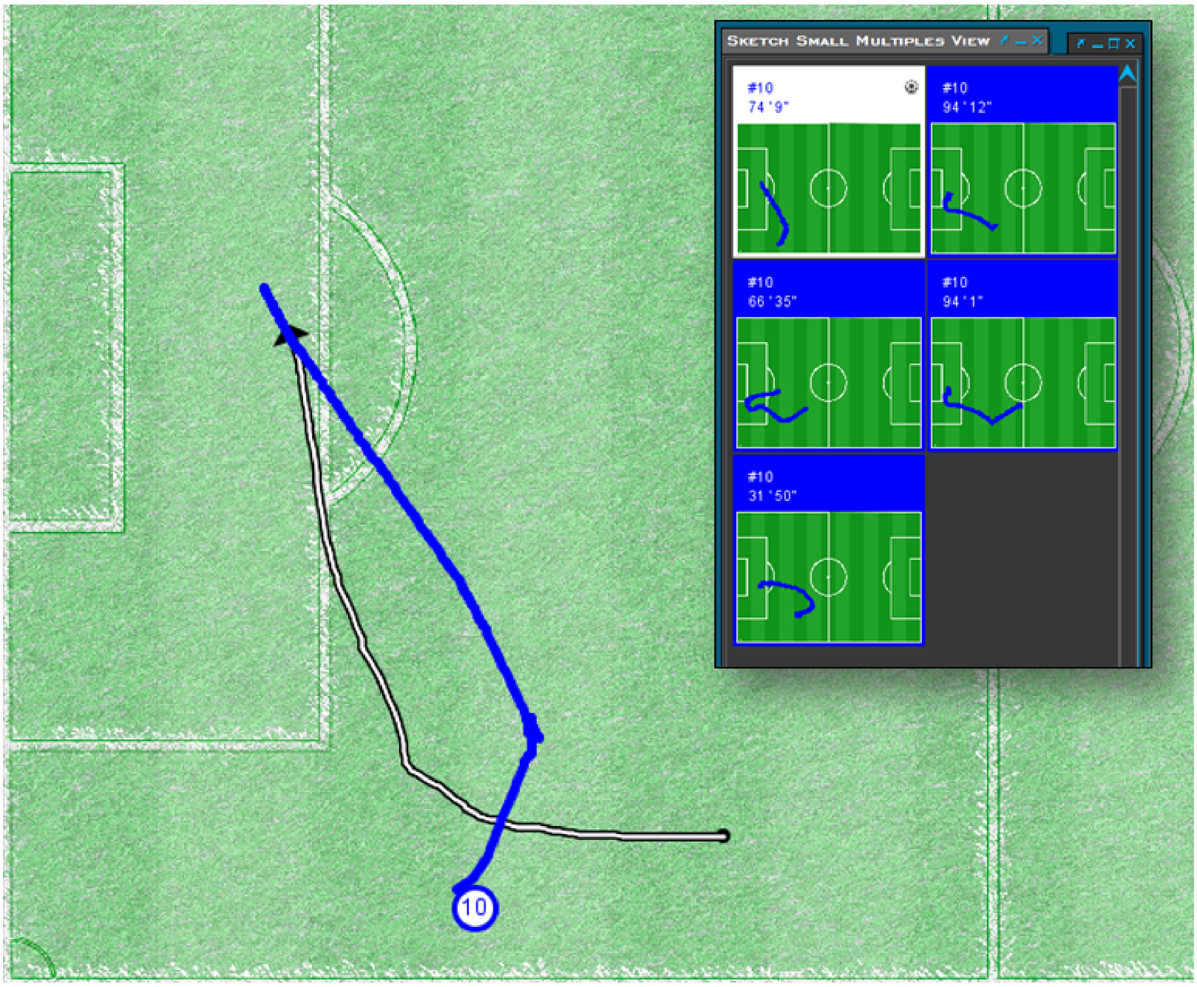


Figure 4.10: Search for a specific movement pattern of the right striker from the away team. It reveals that the player attacks the opponent six times from the left wing side.

the right striker. The small multiple view in Figure 4.10 reveals that such a situation occurs six times in the match and indicates that the player is following this scheme.

4.6 Discussion

We presented a system that is capable of finding selected scenes from the video summary and can be used to answer analytical questions about the performance of individual players, team groupings or even the whole team. Moreover, we realized that a basic single-trajectory search is insufficient for complex movements in soccer data, and additional information about secondary trajectories, trajectory types, and associated events is needed. For Task 1, we might not always find the desired scene on the first try, but this is due to the fact that we focused on retrieving the scenes by rough sketches, and hence did not optimize our user sketches. Of course, we could adapt our sketches to the original trajectories in order to get the best ranking, but that was not our aim and we decided to use the best results that we achieved after a few trials. We aimed to develop an interactive search system that can be quickly and easily used. In our experiments, it was usually sufficient to perform a trajectory search with up to two movements including filter constraints. However, we also tested the performance of our system with a higher amount of trajectories and reached an average query time below 2 seconds for a trajectory search with up to six simultaneous movements.

Eventually, we also conducted an expert interview asking questions about the usefulness, limitations and suggestions for improvement. First of all, our domain expert was excited about the use of our visual search system and stated the contribution as substantial to soccer analytics. He enjoyed to play with the tool and confirmed our requirements of an easy and intuitive search interface. He also sees great potential for the search approach and indicated the usage of our tool for individual player analysis or to quickly search for interesting game situations, during half-time breaks (half-time analysis).

Furthermore, we identified together limitations of the search and provided a number of improvement ideas that we want to realize in the future. For instance, to enhance our search approach by adding translation-, rotation- or scale-invariance to enable a more accurate search for local movement patterns. It is also conceivable that an integration of a recommendation system, as presented in [173], could improve our search system by showing popular subsequent movements during the sketching process. Another important issue is to enrich the analysis possibilities of the system. Game phases could be automatically recognized by the system and integrated into the results. Also, it would be helpful to include derived performance measures about the players into the system, eventually allowing to compare performance (e.g., distance covered, goals on average etc.) with play movements. The system could also be extended for multiple matches and thus, enable the possibility to search for typical behavior over a whole season. In addition, the system may annotate identified game phases of the opponent's previous matches to automatically learn their tactical strategies or build a summary report of favorite attacks. Finally, the search space of movers ($M \in \{Player, Ball\}$) could be increased by adding the trajectories of the referees for decision analysis. Thus, questionable decisions can be analyzed by comparing the referee's point of view and the event position to measure the uncertainty of the referees.

4.7 Conclusion

We presented a novel visual search approach including multi-trajectory and event-specific search options for identifying tactical movements and unannotated scenes in soccer matches. By means of our interactive search interface that supports area and path specific queries, sports analysts can conveniently sketch movement patterns based on individual, formation, or team dependent trajectories. We used a preprocessing step to extract a set of locally similar trajectory parts to a given sketch and applied two trajectory descriptors (similarity functions) to rank the similarity based on two different tasks. The two tasks include the search for specific situations that happened in a match and the identification of similar movement patterns. Moreover, we designed a domain-specific task analysis for movement search in soccer data and conducted a case study together with a domain expert to show the applicability of the tool. Our system was designed as an expert tool that can reveal individual player behavior and tactical team movements in a match, which can be important for decision makers such as coaches, scouts or managers. However, our visual search system may also be applied to other application areas, such as animal movement analysis, or team sports where the search for simultaneous movements is needed. Thus, for instance, our methods could be used in sports like American football, basketball or rugby to identify important team movements, which creates open spaces and game deciding points.

Part II

Interactive Lens Techniques for Exploring Local Patterns

CHAPTER 5

INTERACTIVE LENS FOR EXPLORING SCATTER PLOTS

Contents

So far I discussed retrieval approaches to find local patterns in the data. However, it is often the case that analysts do not have a particular pattern in mind and have to explore the data set fist, in order to find interesting patterns that could lead to insights. In such cases interactive exploration tools may help to search through large view spaces and detect patterns of interest. For instance, by using *details on demand* techniques analysts may interactively select parts of data and immediately gain detailed information about the selected region –local pattern.

In Statistics, often models are used to explain patterns in data and reduce possible large data set to more compact model-based representations. Among others, regression models are widely used to explain data. However, regression analysis typically searches for the best model based on the global distribution of data. On the other hand, a data set may be partitioned into subsets, each requiring individual models. While automatic data subsetting methods exist, these often require parameters or domain knowledge to work with.

In this chapter, I present a system for visual-interactive regression analysis for scatter plot data, supporting both global and local regression modeling. I introduce a novel regression lens concept, allowing a user to interactively select a portion of data, on which regression analysis is run in interactive time. The lens gives encompassing visual feedback on the quality of candidate models as it is interactively navigated across the input data. We show, by means of use cases, that our regression lens is an effective tool for user-driven regression modeling and supports model understanding.

This chapter is based on:

[171] **Interactive Regression Lens for Exploring Scatter Plots,** L. Shao, A. Mahajan, T. Schreck and D. J. Lehmann. *Computer Graphics Forum, Eurographics Conference on Visualization (EuroVis). The Eurographics Association and John Wiley & Sons Ltd., 2017.*

5.1 Introduction

In the big data era, relevant data is constantly growing in many domains and visual-interactive techniques are becoming more and more important. There already exist techniques that help analysts to explore and explain different types of data, in different application domains. The scatter plot is a well-known basis technique to explore correlations, trends and clusters in bivariate data. Exploration with scatter plots can benefit from interest measures like Scagnostics [216] or regressional features [163], to search and identify informative views in larger sets of scatter plots, e.g., a scatter plot matrix (SPLOM).

An interesting extension for analysis of scatter plots is the investigation of *local patterns* in single projection views [92, 132, 178]. Prior research has shown that multivariate data sets may contain locally valuable information that has to be extracted and properly visualized. In this regard, local patterns can be represented by local regression models that in sum can describe a global scatter plot of a set of local models. The analysis of local regressions, also known as segmented regression [207], plays an important role in statistic modeling and is used to find substantial changes in relationships among variables. Therefore one dimension, usually the independent variable, has to be partitioned into intervals for computing the local models. But typically, the partition breakpoints are not known before the analysis and have to be estimated.

Automatic approaches for building regression models are typically limited with respect to incorporating domain knowledge in the process of selecting input variables (also known as feature subset selection). Furthermore, the data must either be labeled or well clustered to compute local regression models automatically. However, there are many clustering algorithms and parameterizations to choose from, which in practice often result in many possible cluster segmentations. The challenge here is to choose the best clustering algorithm including the parameter setting for a given data set. Other potential limitations of algorithmic local regression analysis include the identification of local structures, transformations, and interactions between variables.

This work focuses on the visual-interactive extraction and representation of local regression models in scatter plots. Therefore, we introduce regression lens, a novel concept for visual-interactive regression analysis that allows users to select a portion of data on which they want to conduct regression analysis. The lens can be interactively modified in terms of position and size, to find the best fitting model for areas of interest. Detailed interactive feedback allows to compare different models and data selections effectively. Using the regression lens, users can explore scatter plots in a novel way and are enabled to investigate a plot by their individual constituents of regression models. Based on a best-fit algorithm including data sampling and cross-validation, it determines the best coefficients for the selected data independent of the model direction i.e., $f_y(x)$ or $f_x(y)$. We provide a set of appropriately defined and visualized statistical measures, for judging the quality of candidate models.

The remainder of this chapter is structured as follows: In Section 5.2, we discuss related work. Section 5.3 gives an overview of the regression lens approach and describes challenges in regression-based data analysis. Section 5.4 introduces our prototype system including an implementation of the regression lens concept and its usability. Next, in Section 5.5, we apply the approach to different data sets and showcase the exploration benefits. Limitations and possible extensions are discussed in Section 5.6. Finally, we conclude in Section 5.7.

5.2 Related Work

The work that we present in the next sections relates to local data pattern analysis, feature extraction and interactive lens techniques for scatter plot visualizations. Now we discuss some related works in that area and how our work differs from them.

5.2.1 Correlation Analysis and Feature Extraction

In scatter plot analysis advanced data analysis tasks, such as feature computation, pattern extraction or statistical analysis, require important initial steps of assessing correlations, trends and clusters. Regression analysis is widely used to explore statistical relations between selected pairs of variables. In [11], Anscombe explored the importance of graphs and looked into the usefulness and importance of statistical analysis of scatter plot data using regression analysis. Nowadays, data analysis tools like Tableau[1] are available that provide such analysis possibility but, to date, are limited in focus-plus-context techniques such as interactive lenses.

An interactive framework for building and validating regression models is presented by Mühlbacher et al. in [136]. The framework helps analysts to understand relationships between observed variables and a dependent target variable, and explains the most useful feature and its partitions by using a regression model. Another related work is [72]. There, Guo et al. defined model space visualizations including heatmap-based displays, which help identify linear dependencies for multivariate data. Li et al. [122] conducted a study about the effectiveness of judging correlations in scatter plots. It turned out that scatter plots are more effective in supporting visual correlation analysis than parallel coordinate plots.

For high-level analysis tasks, a combination of techniques from data mining and interactive visualization can be applied to facilitate finding patterns in possibly large data. For instance, Scagnostics [216] characterizes the global distribution of points based on geometrical and topologic properties. Specifically, nine features including density, shape, stringiness and outlier measures are defined. These features can serve to rank and select plots for interactive inspection. In [173], image-based descriptors are used to search for scatter plot patterns based on user sketch queries. The similarity between a scatter plot pattern and a user sketch is measured by the density of points and the frequency of different edge orientations in the image space. Scherer et al. [163] presented a scatter plot descriptor based on regression features for comparing scatter plots with each other. Specifically, they proposed a feature vector based on the goodness-of-fit of a set of globally applied regression models.

A well-known problem of scatter plots is the degree of overdraw on local regions as the number of points increase. To tackle this problem, hexagonal binning [36] can be applied, encoding point density with a colormap within hexagonal binning regions. Further, Mayorga and Gleicher [132] developed an abstraction approach to automatically group dense data points and used color blending and contour lines to reveal hidden data distributions. In [40], a hierarchical multi-class sampling technique is used which simplifies the distribution by preserving relative density features.

Besides visual abstraction approaches, an investigation of local patterns can also be helpful to reveal further insights, which may be hidden in the overall view. For instance, scatter plots could be extended by sensitivity coefficients to visualize local variation of one variable with respect to another [39]. They represent the sensitivity information as velocities so that the resulting visualization resembles a flow field. In [92], a method is presented to emphasize a local area of interest based on depth of field and a multidimensional focus selection body. In [178], a scatter plot interest measure is presented, which is based on an adapted tf×idf approach computed over sets of local clusters. This measure is used to rank scatter plots based on the frequency of local

[1] `https://www.tableau.com`

patterns, useful to propose views to a user from a large scatter plot view space.

For high dimensional data sets, scatter plot matrices [36] in combination with the brushing and linking technique [13] may be useful for finding related patterns across multiple scatter plot projections. Alternatively, the point distribution of multivariate data can be displayed onto 2D planes by using radial projection-based visualization techniques like Radviz [84] or Star Coordinates [103, 104]. The effect of judging correlations for these projection-based visualizations are published in [141, 142]. In [116], guidance pictograms are presented to support standard visual search tasks, such as correlation and distribution analysis, for projections like scatter plots, Radviz and Star Coordinates.

5.2.2 Interactive Lens Techniques

To interactively explore local scatter plot regions for interesting patterns, virtual lens techniques like the magic lens or magnification lens may be used [25, 112], which provide on demand an alternative visual representation of the underlying data. There already exist a number of different interactive lens techniques for various applications and data domains. For instance, there are lenses to show temporally aggregated information of trajectory data (time lens [198]), to explore multivariate network data (network lens [101]) or to magnify volumetric features in 3D representations (magic volume lens [210]).

Moreover, there are specific lens techniques to support the analysis of multivariate data in scatter plot visualizations. Ward and Yang [213] have presented an overview of interaction operations that can occur in data and information visualization including lens techniques (distortion) for scatter plot matrices. To overcome the overdraw problem Ellis et al. [52] have introduced the sampling lens, which estimates a suitable sampling rate for the underlying selection and shows a clutter-reduced representation. SemLens [77] is another lens technique for scatter plots, which assists local analysis by adding further analytical dimensions to certain regions of the scatter plot. A structure-based semantic lens for scatter plots and graph layouts is presented in [90]. This lens technique keeps the selected data records –under the lens surface– unchanged and continuously deforms the data out of the focus in order to maintain the context around the lens. Bertini et al. [23] have introduced an extended excentric labeling lens, which dynamically displays labels to the selected data records around the lens. In [118], a data-dependent magic lens is presented to minimize the projection related distortions in Radviz and Star Coordinate visualizations. A survey on visual interactive lens and distortion-oriented presentation techniques are given in [120, 199]

5.2.3 Delineation and Contribution

Previous works have defined useful methods to visualize correlations and features to rank and select scatter plot patterns. Also, several interactive lenses for scatter plot visualizations introduce distortion or sampling techniques to support the data exploration process by preserving

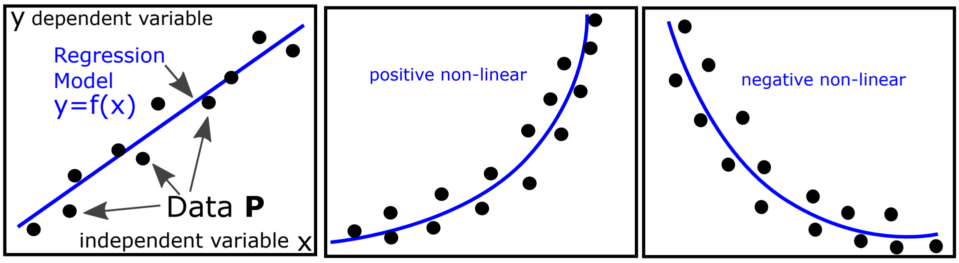


Figure 5.1: Regression models: (left) linear and (middle-right) non-linear models (blue) to explain the data (black).

an overview of the data during drill-down operations. Our approach is novel in that we extend the scatter plot lens concept to compute and visualize in interactive time candidate regression models for user-selected subsets of data. The tool supports modeling of data by sets of local regression models, hence contributing a novel tool for scatter plot analysis.

5.3 Concept of the Regression Lens

Next, we describe basic concepts and issues of regression-based data analysis. We will then derive our interactive regression lens concept and its visualization building on these concepts.

5.3.1 Regression in Practice

We start with relevant background knowledge for univariate regression analysis of data: Given a 2D set of data elements $\mathbf{p}_i = (x_i\ y_i)^T; i = 1, \dots, m$ with $\mathbf{P} = (\mathbf{p}_1\ \dots\ \mathbf{p}_m)$. A regression model is a function $y = f(x)$ explaining the data elements $(x_i\ y_i)^T$ best. Figure 5.1 illustrates this regression model concept.

The idea of univariate regression data analysis is to find a functional relation in the data. It allows to compare different data sets with each other, it supports to have options to compress or to classify the data, and it mutually binds two (or more) variables with each other, which were unrelated and independent before. Finding these phenomenological relations between variables/dimensions is the most relevant aspect of a regression-based data analysis.

Typically used univariate regression models are exponential models

$$y(x)_e = \beta_{e_1} \cdot e^{\beta_{e_2} x},$$

logarithmic regression models

$$y(x) = \beta_{l_1} + \beta_{l_2} \cdot \ln(x),$$

power fit regression models

$$y(x)_p = \beta_{p_1} \cdot x^{\beta_{p_2}},$$

or polynomial regression models of degree n

$$y(x)_n = \sum_{i=0}^{n} \beta_i \cdot x^i = \boldsymbol{\beta} \cdot \mathbf{x}, \quad \boldsymbol{\beta} = (\beta_0 \ldots \beta_n), \ \mathbf{x} = (x\ x^2 \ldots x^n), \tag{5.1}$$

where the parameters β_i are called regression coefficients, while x is the independent and y the dependent variable.

The polynomial regression model is appropriate to substitute or mimic a set of further models. In fact, the power fit $y(x)_p$ is a subset of $y(x)_{n=\beta_{p_2}}$, positive exponential models $y(x)_e$ can be approximated with $y(x)_3$, etc. Since a wide area of regression cases is covered, we focus on the family of polynomial regression models in this work. In this regard,

$$y(x)_1 = \sum_{i=0}^{1} \beta_i \cdot x^i = \beta_0 + \beta_1 \cdot x^1 \text{ is called a } \textit{linear model},$$

$$y(x)_2 = \sum_{i=0}^{2} \beta_i \cdot x^i = \beta_0 + \beta_1 \cdot x^1 + \beta_2 \cdot x^2 \text{is called a } \textit{quadratic model}, \text{ and}$$

$$y(x)_3 = \sum_{i=0}^{3} \beta_i \cdot x^i = \beta_0 + \beta_1 \cdot x^1 + \beta_2 \cdot x^2 + \beta_3 \cdot x^3 \text{ is called a } \textit{cubic model}.$$

The estimation of a good regression model comes with a set of issues to be handled, which are denoted as the issue of *underfitting vs. overfitting*, *independent vs. dependent* variable, and *uniformity vs. clumpiness*. Subsequently, we introduce and discuss these issues.

5.3.2 The Regression Issues

We describe common issues when using regression in practice.

Underfitting vs. Overfitting: A model is well chosen if it fits the data and if it is the simplest model for doing so. To fit the data, the in-sample error $e(f)$ for model f is

$$e(f) = \sum_{i=1}^{m} (y_i - f(x_i))^2, \tag{5.2}$$

also known as sum of squared errors (SSE). An appropriate model f minimizes $e(f)$ to prevent data underfitting. See Figure 5.2 (left-middle), where e is stressed by red lines.

In contrast, choosing the simplest of such in-sample error minimizing models prevents overfitting: “simple” is such a model with a small complexity, e.g., a small number of regression coefficients β_i. In fact, considering a number of m data points, with $y_i(x_i) \neq y_j(x_j)$; there is always a polynomial $y(x)_n$ of degree m fitting the data perfectly (i.e., it interpolates the data), with $e(y(x)_{n=m}) = 0$ (see Figure 5.2 (right)). Does it mean that $y(x)_{n=m}$ is still the optimal model? For obvious reasons, this is not the case: this model is not simple but complex with a large number of m parameters $\beta_i, i = 0, \ldots, m$; the model has a large waviness and thus it is also geometrically complex; and for each new data element, every time the model requires one new term and one more regression coefficient –which does not seem to be plausible at all– known as overfitting.

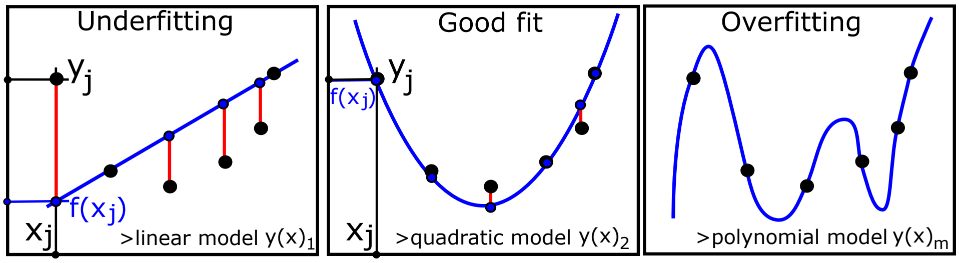


Figure 5.2: Underfitting vs. Overfitting.

To prevent overfitting, a model $y(x)_n = f_n$ of degree n is assigned with the out-of-sample error $o(f_n)$ given as:

$$o(f_n) = \frac{1}{2} \cdot \sum_{j=0}^{m} (f_n^{\mathbf{P}_1}(x_j) - f_n^{\mathbf{P}_2}(x_j))^2, \tag{5.3}$$

where $\mathbf{P}_1, \mathbf{P}_2$ are subsets of $\mathbf{P}$. These subsets are mutually disjoint with $\mathbf{P}_1 \cap \mathbf{P}_2 = \emptyset$, have a similar number of elements –ideally each set has a number of $m/2$ data representatives, i.e., $\mathbf{P}_1 \cup \mathbf{P}_2 = \mathbf{P}$– and each subset should have a similar distribution behavior as $\mathbf{P}$ in order to mimic its statistical properties, making a comparison fair. When using more than two disjoint subsets, i.e. $\mathbf{P}_1, ..., \mathbf{P}_k$, the out-of-sample error o is given by averaging. Then, a good model f_n preventing overfitting minimizes $o(f_n)$, known as cross-validation. In total, an optimal model f is given by the optimization process for in-sample error e and out-sample error o as

$$f_n \text{ with } arg \min_{\boldsymbol{\beta}, n} \; (o(f_n) + e(f_n)). \tag{5.4}$$

Independent vs. Dependent Variable: The choice of the independent variable for model f is arbitrary. Clearly, both options are reasonable: either choosing $y = f_y(x)$ or choosing $x = f_x(y)$ as dependent or independent variable. A concept to describe the influence of the chosen independent variable is the *correlation*.

The correlation $corr(x, y)$ is a measure describing how unimportant the choice of direction is, so $f_y(x)$ or $f_x(y)$, because both choices lead to the same image of the function. If $|corr(x, y)| = 1$, it means that $f_y(x)$ and $f_x(y)$ look identical, while $|corr(x, y)| = 0$ means that both directions are unrelated and look different. For a linear model $y(x)_1$, the correlation is described by the angle α spanned in between $f_y(x)$ and $f_x(y)$, giving the correlation measure $corr_{y_1}(x, y) = cov(x, y)/(\sigma(x)\sigma(y))$, with the covariance cov and the standard deviation σ. So, if $\alpha = 0$ then $corr_{y_1}$ is 1, if $\alpha = \pi/2$ then $corr_{y_1} = 0$. Figure 5.3 illustrates this. A generalization of this correlation concept for higher order models is known by

$$corr(x, y)_{y_i} = \sqrt{1 - \frac{SSE}{SST}} = \sqrt{1 - \frac{e}{SST}} = \sqrt{1 - \frac{\sum_{j=1}^{m}(y_j - f(x_j))^2}{\sum_{j=1}^{m}(y_j - \overline{y})^2}}$$

with

$$\overline{y} = \frac{1}{m} \sum_{i=1}^{m} f(x_i).$$

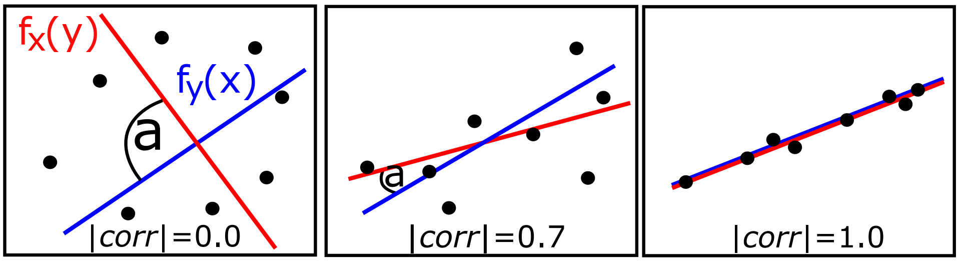


Figure 5.3: Correlation: influence of the chosen independent and dependent variable to the appearance of a linear regression model.

The correlation is an important measure to judge the quality of a regression model.

Uniformity vs. Clumpiness: One last issue is that the data along a good model ought to be uniformly distributed. In fact, a model may run through different clusters in the data, raising the question of how good a model is that would connect clusters of data. Not quite good we argue, as (i) such a model is locally influenced by a varying information density (dense areas and sparse areas), which sophisticate the model, and as (ii) two or more clusters are generally not well described by one model (see Figure 5.4).

For two clusters, two models (one per cluster) seem to be a better choice in terms of fitting the data, which also motivates our concept of a local analysis with our regression lens. However, to measure the distribution of the involved data along the model's pathway, we define a distribution measure $h(f)$ of the non-uniformity for the data elements of model f as

$$h(f) = \frac{1}{2} \cdot (h_x + h_y) \tag{5.5}$$

where h_x and h_y are the deviations of the discrete uniform distribution $P(X = x_i) = \frac{1}{n}$ for $i \in \{1, ..., n\}$. The deviation of the uniform distribution is defined by the goodness of fit measure, which is the sum of differences between observed and expected outcome frequencies of an interval i as

$$h_x = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \tag{5.6}$$

where O_i is the number of observations in interval i and E_i is the expected number in interval i on the x-axis and y-axis respectively – known as χ^2-distribution. A model is considered to be more appropriate if it minimizes $h(f)$.

Note that our lens concept handles all these issues by using appropriate visualization concepts, allowing to observe and compare the quality and applicability of regression models for interactively selected data. The following section describes how our lens is constructed, handling issues of overfitting/underfitting, dependent/independent variables, and uniformity/clumpiness.

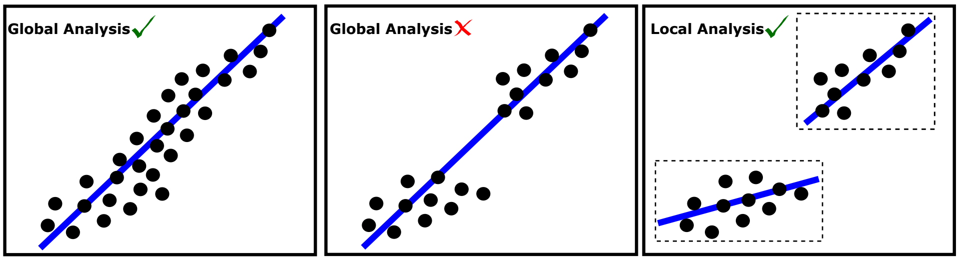


Figure 5.4: Uniformity vs. clumpiness or global vs. local analysis.

5.3.3 Construction of Regression Models for our Approach

By considering our data $(x_i\ y_i), i = 1, \ldots, m$, we define the $(n+1 \times m)$ power data matrix $\mathbf{X}$ as

$$\mathbf{X} = \begin{pmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^n \\ 1 & x_2 & x_2^2 & \ldots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \ldots & x_m^n, \end{pmatrix} \tag{5.7}$$

and write the polynomial model $y(x)_n$ with $\mathbf{y} = (y_1\ y_2 \ldots y_m)^T$ as the linear system

$$\mathbf{y} = \mathbf{X} \cdot \boldsymbol{\beta} + \mathbf{e} \tag{5.8}$$

for which the in-sample error $e(f) = \sum(y_i - f(x_i))^2 = \sum e_i^2,\ e_i \in \mathbf{e}$ is minimized by solving the linear system [60], e.g., by choosing the regression coefficients $\boldsymbol{\beta}$ as

$$\boldsymbol{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{5.9}$$

Please note that the inverted matrix, $\mathbf{X}_{inv} = \mathbf{X}^T\mathbf{X}$, is a $(n+1) \times (n+1)$ matrix, is not growing with the number m of considered data but only with the degree n of the polynomial regression model $y(x)_n$. For instance, $\mathbf{X}_{inv}$ is a 2×2 for linear models $y(x)_1$, 3×3 for quadratic models $y(x)_2$, 4×4 for cubic models $y(x)_3$ etc. Thus, inverting $\mathbf{X}_{inv}$ and solving Eq. (5.9) is a cheap operation as long as the polynomial degree n is not too big. From our experience, a regression model fitting the data well has usually a degree less than $n \leq 5$.

To find an optimal model, our approach solves the optimization process of Eq. (5.4) within a two-step process.

Step 1: To minimize the in-sample error e, our technique considers a set of k different polynomial regression models $y(x)_1, \ldots, y(x)_k$ and ranks them regarding their minimized in-sample errors e, $e_1 < \cdots < e_k$. Figure 5.5 (left) illustrates the first step.

Step 2: From the ranked regression models, we consider a number k_T of the best ranked models as candidates and select the candidate as optimal which has the minimized out-sample error $o(f)$ in the list of candidates. To compute $o(f)$ for the different candidates, our approach uniformly samples the data $\mathbf{P}$ to get the subsets $\mathbf{P}_1$ and $\mathbf{P}_2$, as is seen in Figure 5.5 (right).

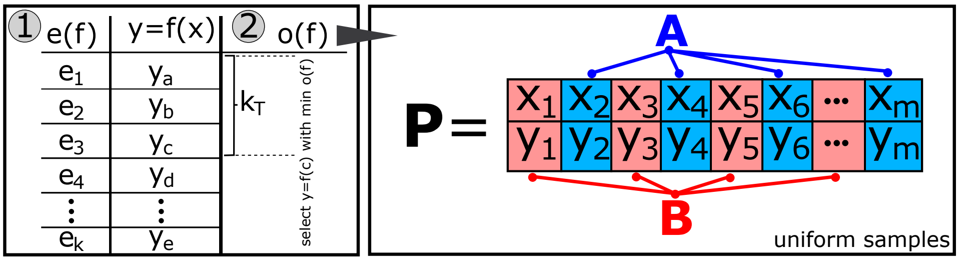


Figure 5.5: Two step process to find the best fitting regression model.

If $k = k_T$, the ranking process in step 1 is not required. For proof of concept, we do so by choosing $k = k_T = 4$, i.e., we consider only polynomial regression models up to degree 4, from which we choose the one with the smallest out-sample error as optimal.

Finally, for the purpose of later use, from the chosen model $y(x) = f_y(x)$ our approach calculates the correlation $corr(x, y)$, the model $f_x(y)$ by switching dependent & independent variables and uniformity properties, as described above. By having the regression model and the attributes, we are prepared to subsequently explain our visual design for the regression lens.

5.3.4 Visual Design of Regression Lens

To select a subset of data elements for our regression lens, a box, circle, free-form, or a manual selection can be basically used, illustrated in Figure 5.6 (a-c). In that regard, a circle or free-form selection does not naturally yield axially parallel edges that are required to further visualize dimension-wise aligned (statistical) information. Moreover, a free-form selection causes an additional cognitive effort and a lot of further interaction steps, which may be exhausting. On the other side, a manual selection is too expensive and time consuming if a large number of data points need to be selected. We like to keep it simple for proof of concept. Thus, considering these reasons, we rely in this work on a rectangle selection by a simple user mouse drag operation. However, integration of further interaction schemes, if needed is straightforward to do.

With the rectangle selection scheme, the user selects a subset of data elements, to which regression analysis, quality computation and optionally, user guidance regarding improvement possibility (c.f. Chapter 7.3.2) is applied. Please note that by the choice of a rectangle, the “locality vs. globality” level of the analysis is implicitly user-defined.

As already explained, both variables x and y can be seen as a correct choice for the independent variable for the univariate polynomial regression, and both models $f_y(x)$ and $f_x(y)$ are correct in a way. This is also justified by the fact that a plot p_{xy} for any dimension x and y of the SPLOM is equivalent to the transposed version of the plot for the dimensions y and x, i.e., $p_{yx}^T = p_{xy}$ where variable x and y are interchanged. Consequently, an order of the variables does not exist by nature. Figure 5.6 (d,e) illustrates this. Thus, our approach needs to draw both found regression models in the lens selection area, to allow a complete insight for the users. While $f_y(x)$ can be drawn with standard techniques in the x-y-space, an inverse of $f_x(y)$ (which is given in y-x-space)

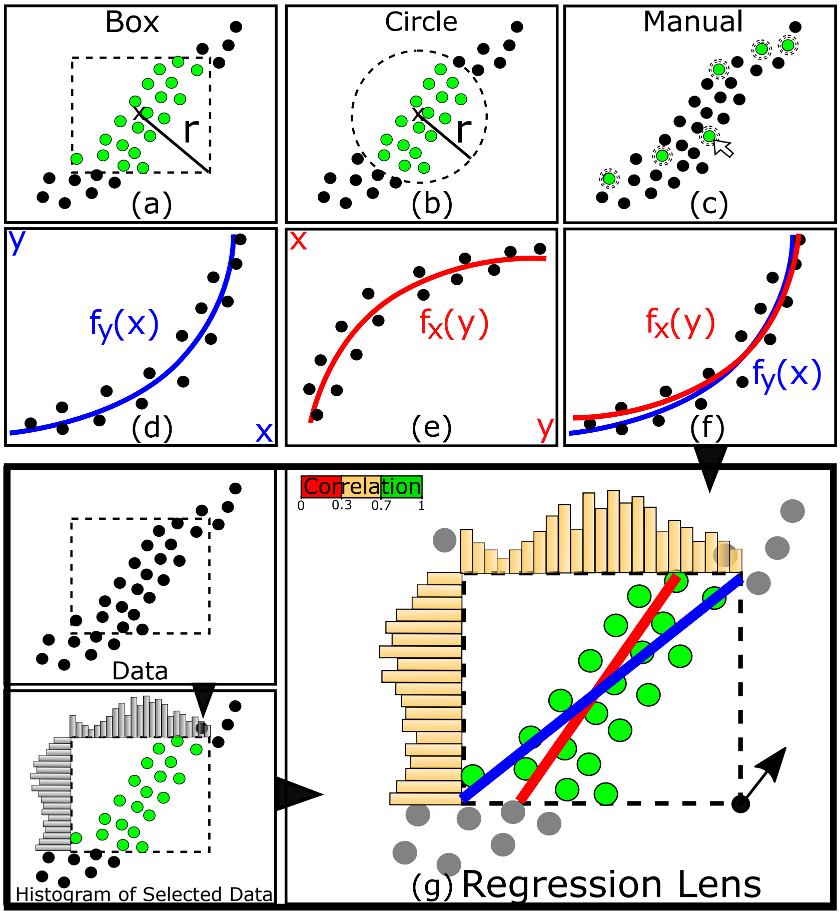


Figure 5.6: Visual Design of the Regression Lens.

does not necessarily exist. Thus, our approach exploits the symmetry that point (y,x) for $f_x(y)$ is equivalent to point (x,y) in the space of $f_y(x)$, to draw $f_x(y)$ in x-y-space: $(y,x)_{f_x(y)} \rightarrow (x,y)_{f_y(x)}$ (see Figure 5.6 (f)).

To judge distribution properties, our approach visualizes a normalized histogram for each variable on the boundaries of the selection box (see Figure 5.6 (g)). Since the boundaries of the selection box are axis-parallel and thus assigned to the variable directions of the plot, the histograms can easily be mentally connected with the variables. This also motivates to use a box selection instead of other options.

Due to the fact that the color for the histogram bins and the boundaries of the box are free usable visualization parameters, our approach maps correlation values to discrete colors. Specifically, $corr(x,y) < 0.3$ is mapped to red, $0.3 \leq corr(x,y) < 0.7$ is mapped to orange, and $0.7 \leq corr(x,y) < 1.0$ is mapped to green, to visually stress the level of correlation. Note that this is a pragmatic choice and other color mapping schemes, including continuous mappings, are possible in principle.

As part of our concept, both directions $f_y(x)$ and $f_x(y)$ of the respective optimal polynomial regression model are drawn, as well as both univariate histograms for x and y. In addition, we show the correlation information $corr(x,y)$, as can be seen in Figure 5.6 (g).

To distinguish the different directions of the drawn models, we color models of $f_y(x)$ in blue and models of $f_x(y)$ in red (shown in Figure 5.6 (d - f)). Furthermore, we provide another optional color coding to directly visualize the in-sample error on the models' pathway. By means of this visual feature, users can quickly identify the quality of the regression model according to the selected points. Therefore, we compute the Euclidean distance of the model's pathway to the nearest point and map the distance by using a diverging red-green color coding, as demonstrated in Figure 5.7(b) – bottom lens. Moreover, users can compare the in-sample errors of different models (e.g., linear vs. quadratic) and spot unsuitable parts on the models' pathway. This support users e.g., to split inappropriate selections (colored in red) into individual subsets for modeling.

5.4 System Overview

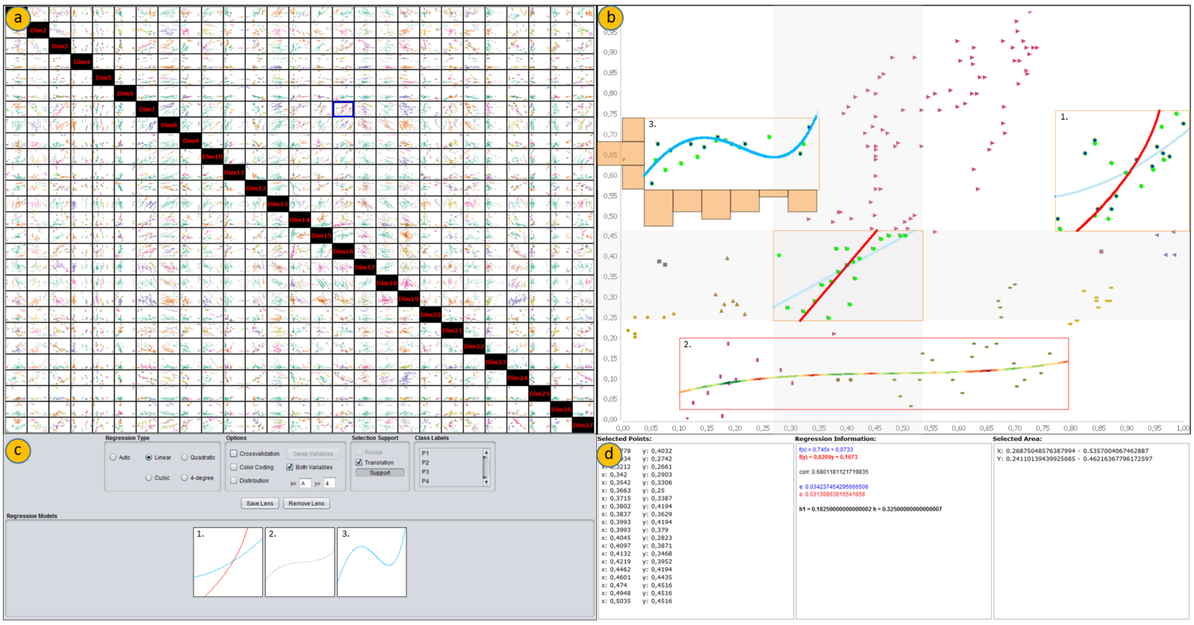
In this section, we present the design implementation of our regression lens concept together with our prototype implementation. Figure 5.7 shows an overview of the system.

5.4.1 System Design

To analyze scatter plots from high-dimensional data sets, we present all pairwise combinations of dimension variables in a SPLOM, as shown in Figure 5.7(a). Data points that belong to a particular class label (if available) are visualized by different colors and can be filtered out for further investigation. The user selects one cell (plot) from the SPLOM which is shown in detail in (b). The result of the regression analysis for interactively selected data subsets is shown directly on the lens in this view (Figure 5.7(b)). Individual settings for local lenses, such as model selection, activating distribution histograms or class label filtering, can be performed in the setting view (Figure 5.7(c)). Moreover, the user can save interesting findings and previous settings of lenses in this view. Detailed information about the current lens selection is shown in Figure 5.7(d). This information includes selected points, boundaries of the area and statistical measures like $corr(x,y)$, $e(f)$ and $h(f)$ values.

For specific analytical tasks, users can limit the degree of the model and manually switch between the polynomial models as well as the direction of the model $f_y(x)$ and $f_x(y)$. Alternatively, users can let the system choose the best fitted model and the direction according to the selected points. If both directions are simultaneously drawn, the system automatically reduces the saturation of the less fitting direction to highlight the better model, as shown in Figure 5.7(b). To measure the best model and direction respectively, the $e(f)$ values of each combination can be compared.

Since the computation of the distribution measure $h(f)$ depends on the histogram bin size, we not only allow users to adjust this parameter setting but also provide an automatic selection



Regression Type

○ Auto ● Linear ○ Quadratic
○ Cubic ○ 4-degree

Options

□ Crossvalidation Swap Variables
□ Color Coding ☑ Both Variables
□ Distribution x= A y= 4

Selection Support

□ Resize
☑ Translation
Support

Class Labels

P1
P2
P3
P4

Save Lens Remove Lens

Regression Models

1. 2. 3.

Selected Points:

```
    778    y: 0,4032
    934    y: 0,2742
x: 0,3212  y: 0,2661
x: 0,342   y: 0,2903
x: 0,3542  y: 0,3306
x: 0,3663  y: 0,25
x: 0,3715  y: 0,3387
x: 0,3802  y: 0,4194
x: 0,3837  y: 0,3629
x: 0,3993  y: 0,4194
x: 0,3993  y: 0,379
x: 0,4045  y: 0,2823
x: 0,4097  y: 0,3871
x: 0,4132  y: 0,3468
x: 0,4219  y: 0,3952
x: 0,4462  y: 0,4194
x: 0,4601  y: 0,4435
x: 0,474   y: 0,4516
x: 0,4948  y: 0,4516
x: 0,5035  y: 0,4516
```

Regression Information:

```
f(x) = 0,745x + 0,0733
f(y) = 0,6209y + 0,1673

corr: 0.6801181121719835

e: 0.034237454295666506
e: 0.03130863816541658

h1 = 0.18250000000000002 h = 0.32500000000000007
```

Selected Area:

```
X: 0.26875048576387994 - 0.5357004067462887
Y: 0.24110139439925685 - 0.46216367796172597
```

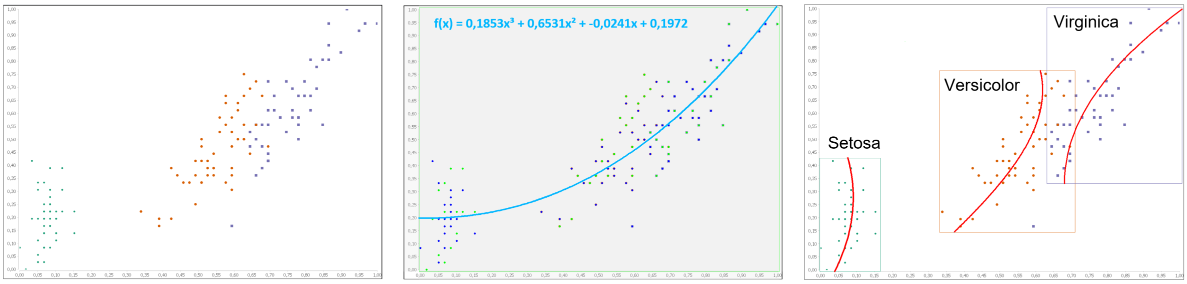
Figure 5.7: Our prototype is separated into four views: (a) visualizes multivariate data by means of a SPLOM; (b) is the interactive analysis area for the investigation of local regression models for a chosen cell from the SPLOM; (c) is a settings window to control regression computation; (d) shows additional information like selected points, regression coefficients and correlation measure for the current selection to be investigated.

based on the equal-width binning approach. For automatic selection, we determined the bin size by the square-root choice that takes the square root of the number of samples in the lens. In this way, the comparison of different local lenses is not influenced by the sizes of the lens.

By default, we color-code points selected inside the lens box in green. The cross-validation subset is coded in dark blue, and the regression models in red and blue, respectively. All color-schemes, for the figures in this paper and the application, are taken from Colorbrewer [74]. These color settings can be changed individually, e.g., to circumvent color perception disabilities, if present.

5.4.2 Implementation Details

We implemented the user interface of the regression lens in Java and integrated an R-Environment for the statistical computations. To render scatter plots, the JFreeChart library is used. To provide a smooth exploration, we implemented our system using two threads, which separate foreground and background computations. The foreground thread handles user interactions and display updates. The background thread translates screen coordinates to plot coordinates, calls R to compute the candidate regressions and other needed information. For computing the regression models, we used the standard linear models function `lm()` in R.



(a) Iris data set – petal length/sepal length.

(b) Global regression analysis.

(c) Local regression analysis.

Figure 5.8: Comparison between global and local regression analysis with the Iris data set. (a) Shows a scatter plot that visualizes petal length as independent variable (x) and petal length as dependent variable (y). The color coding denotes the different class labels of the data (species). (b) By applying a usual global regression analysis, we obtain a cubic function as best fitting model according to the test data subset (colored in dark blue). (c) Shows the result of local regression analysis for the different flower species setosa, versicolor and virginica.

5.5 Case Study

Next, we demonstrate the usability of our approach and evaluate the different regression issues by using well-known data sets from the UCI Machine Learning Repository [123].

Global vs. Local Analysis: One primary goal is to enable an interactive exploration for local regression models in scatter plot data. Figure 5.8 exemplifies this intention with the help of the Iris data set and shows an interactive outcome compared to a general global regression analysis. The first figure on the left (Figure 5.8a) shows the distribution of petal length against sepal length, and highlights the different iris species (setosa in green, versicolor in orange and virginica in purple). The plot shows clear separated patterns of the different classes. However, if we apply the regression lens over the whole data (Figure 5.8b), it returns a cubic model $f_y(x) = 0.185x^3 + 0.653x^2 + 0.024x + 0.197$ as best fitting model (cf. Section 5.3.3). By interactively positioning three local regression lenses for the different classes (Figure 5.8c), one can see that none of the three models has a similar trend compared to the global one. In fact, the main directions of the models have changed to the direction $f_x(y)$ and describe different models for the three classes.

Underfitting vs. Overfitting: Next, we consider the underfitting/overfitting issue with respect to our regression lens concept. Figure 5.9 demonstrates the effect of including and excluding cross-validation for the regression computation. In the background, the actual lens selection is shown, which at first glance seems to capture only little data. Actually, the selection contains 49 data points which are mostly overdrawn due to their similarity. The data point framed in red actually contains 30 overdrawn points and highly influences the regression model computation. This is the reason why the lens tries to match this point in a very precise manner in the previous

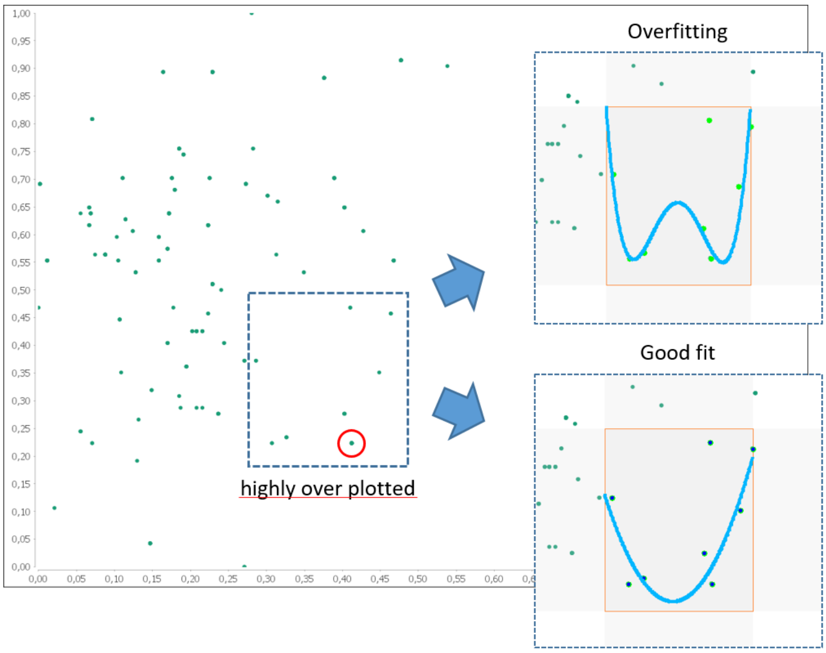


Figure 5.9: Overplotting issue – Bosting Housing data set: By using the raw selected data points our regression lens returns a polynomial model of degree 4 as best fitted model (Overfitting). However, if we include the out-of-sample error into the regression determination, it changes the model to a quadratic model (Good fit).

example (Overfitting). By activating cross-validation, we decrease the weighting of this single area and receive a quadratic model, which seems to be more suitable for this selection.

Independent vs. Dependent Variable: Since our approach provides two different directions ($f_y(x)$ and $f_x(y)$) for each polynomial model, the number of possible models increases. To reveal how important the choice of a direction is, the correlation measure $corr(x, y)$ as defined in Section 5.3.2 is used. Figure 5.10 depicts the information content of the correlation measure. It shows the extreme examples for each model by using the Auto MPG data set. On the left, we compare a local area of the plot car weight against displacement, which has a strong positive correlation. In this case, it is obvious that it does not play a major role in which direction the user is going to choose, since all models describe the positive trend very well. Moreover, one can see that the $corr(x, y)$ value stays stable for each model and remains at around 0.81. On the other hand, if the correlation is weak, the choice of direction may influence the analysis process, or worse, lead to improper hypotheses.

Uniformity vs. Clumpiness: The last example covers the issue of finding uniformly distributed selections for the lens. To judge a selection, the user can compare the quality by the distribution measure $h(f)$ and the shown normalized histograms on the variable axes of the regression lens. Figure 5.11 exemplifies this functionality on the Wine data set. In the background, we show a bad selection example of two different wine clusters (class 1 and class 3). This is indicated by the

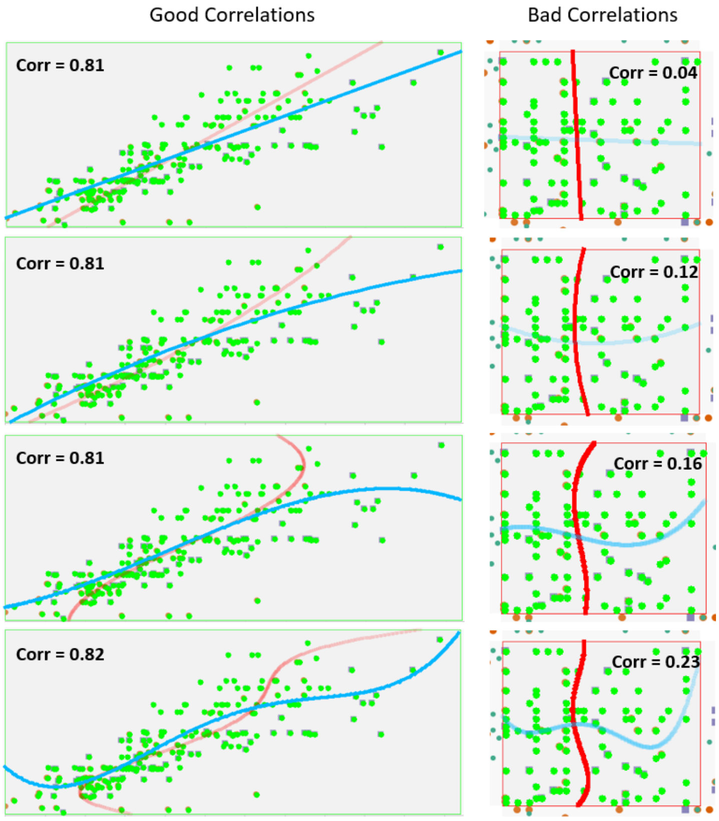


Figure 5.10: Independent vs. Dependent Variable – Auto MPG data set: Impact of correlation measure demonstrated on extreme examples for each polynomial model. Good correlation here means that the choice of direction is unimportant, whereas a bad correlation indicates that the models show to be very different.

relatively high $h(f)$ value of 0.65 and the unequal distributed histograms on the variable axes (i.e., two peaks on the x and y axis). To improve the selection, the user can gradually downsize the selection box and compare visual and computational improvements on the histograms and $h(f)$ value. Gaps along the histogram axes are good split indicators. If we split the two clusters (red and blue selection), one can see an improvement in the results. The red selection examples show an iterative improvement of the $h(f)$ value by minimizing the height of the box. In the end, we improve over the initial single selection (total distribution measure $h(f) = 0.65$) to a two-fold sub-selection (with $h(f) = 0.45$ and $h(f) = 0.44$ distribution values).

5.6 Discussion and Extension Possibilities

Our regression lens concept allows to define and compare models of user-selectable locality. Local modeling involves the segmentation of data which is typically a hard problem to do automatically, since often parameter settings are required that do not fit for all data or user interests. Our lens approach allows to easily factor in user background knowledge. Compared to fully-automatic

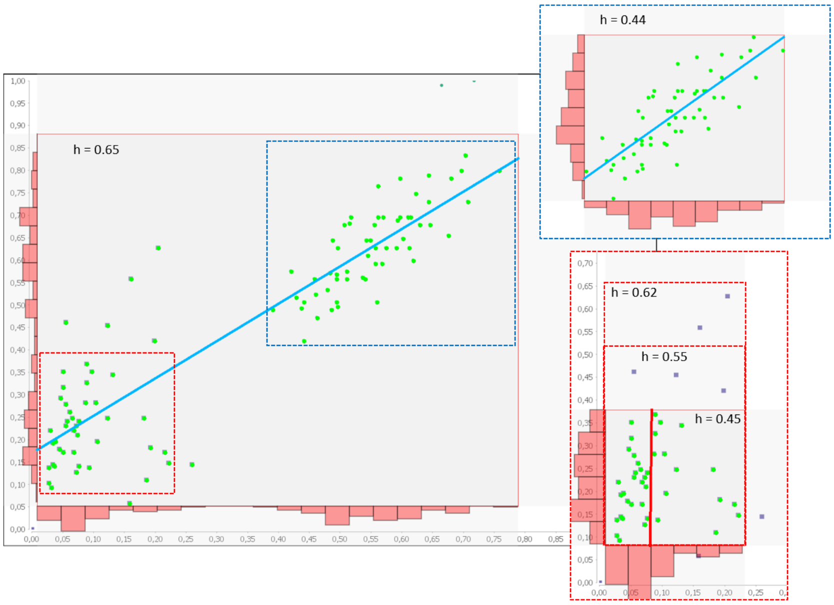


Figure 5.11: Uniformity issue – Wine data set: Lens selections can be verified by comparing the axes histograms (visually) or by the $h(f)$ measure (analytically). In the background, a bad selection with a relatively high h value is shown, which can be improved by using two separated lenses for the two clusters (cutouts on the right).

analysis approaches, our lens technique provides the degree of freedom for exploring local patterns in classified or unclassified data. It can complement automatic approaches of searching for local regression models, which typically require information about clusters or class labels, or need to specify an automatic data segmentation step.

We use a set of plausible statistical scores to select regression models. A possible extension is to include additional scores or domain-dependent quality measures. For example, one could use Scagnostics features [216] for quantification of patterns. Features like clumpiness or monotonicity can be integrated for validating user selection, and thus help to identify good local selections.

Another important aspect to be considered is the scalability of our approach with respect to the number of selected data items. Therefore, we empirically evaluated the performance regarding different models and data samples. The evaluation was performed on a notebook with an Intel i7-6500U CPU and 16 GB RAM, the results are shown in Table 5.1. The tested sample size includes 100, 1.000 and 10.000 data points. We observe that the response time increases by taking models of higher degrees and, of course, by increasing the data size. Since the automatic lens selection computes all models to find the best fitting model, its response time is accordingly the longest. Our implementation can process lens selections up to 1.000 data items for a pre-selected model at response times in the range of 100-200 milliseconds, which can be considered fully interactive. For larger data size and automatic model selection, we observe response times between 0.7 and 22 seconds. A speedup may be achieved by data sampling or increasing implementation efficiency.

	100	1000	10000
Linear	43ms	104ms	1941ms
Quadratic	30ms	152ms	4189ms
Cubic	31ms	220ms	7287ms
Degree 4	44ms	294ms	10589ms
Automatic	83ms	738ms	22193ms

Table 5.1: Computation time for the regression models.

A recurring problem in data analysis is the issue of dealing with outliers, a well-known problem in practice. Approaches exist to detect and handle outliers in data analysis. Consequently, a guidance approach may be helpful that includes a simple outlier detection to indicate selected points that may negatively influence the model computation.

It will also be interesting to devise methods for higher-dimensional regression. To this end, the definition of variable and data selection is expected to become more difficult. A first idea is to define a process by which the user incrementally adds more variables. Then, we note our approach shows a smaller number of models in-place in the lens. We may also think about using comparative visualization to compare more models against each other. Finally, we note that modeling with the regression lens is an interactive process. It may be useful to record the interaction operations or intermediate models considered by the user. This sequence of operations or models could be shown using provenance techniques, to make plausible how a particular choice of models was obtained by an analyst.

5.7 Conclusion

We introduced the regression lens, a visual-interactive approach for the exploration of global and local regression models in scatter plot data. This approach allows users to interactively select a portion of data on which the regression computation is run. It provides statistical measures and visual feedback features to judge the quality of a given selection as well as the output model. Furthermore, we pointed out and evaluated important analysis issues for our regression lens concept, and demonstrated the applicability and benefits of our approach with use cases on example data sets. We also discussed several extension possibilities.

CHAPTER 6

DISCOVERING LOCAL PATTERNS IN HIGH-DIMENSIONAL DATA AND ITS EMBEDDED SUBSPACES

Contents

Interactive lenses can be used for a variety of analytical applications. However, most visual analytics systems use lens techniques to abstract, encode or filter selected data items in a given view [197]. Another interesting application for lenses is the interactive exploration of data subsets in various subspace configurations. Next, I will demonstrate how interactive lens concepts can be used to validate structures for a given region across multiple subspaces and see, for instance, if a structure remains robust in other subspaces. Dimensionality reduction techniques are often used for exploratory data analysis, transforming high-dimensional data to a lower-dimensional space. Most projection techniques produce a single static view of a selected lower-dimensional space. Often, these lower-dimensional spaces are the basis for data visualization and visual exploration. The selection of dimensions to be reduced can be done either interactively based on user selection, or automatically, based on heuristic measures. However, infinitely many projections are possible for a selected subset of dimensions, and thus any single static projection may be a shot in the dark. Changing the projection from one view to another may cause information loss and clutter in the projection view. To this end, interactive lenses can be used to explore projection changes of individual data points in various subspace configurations and help to determine whether clusters remain compact in other subspaces –variable-to-variable analysis.

This chapter presents a novel technique for projection-based visual data exploration of multiple subspaces. Rather than creating a single static projection, this technique creates a continuum of projections for each exploration step adding dimensions, through a process of continuous dimension weighing. The space of projections is visualized by displaying trajectories in the dimensionality-reduced space, which can be brushed and animated. In addition, an interactive lens supports the exploration of selected subsets of data points. Structures in the original data space can be recognized by interactively exploring the continuum of projections, guided by visual aids for grouping and comparing data subsets along subspaces. As we show by use cases, the approach provides an explorable visual overview of subspaces, and is useful for verifying or falsifying the presence of structure in the high-dimensional data space.

This chapter is based on:

[175] **Integrated Projection Paths for the Discovery of Patterns in High-Dimensional Data and its Embedded Subspaces,** L. Shao, S. Kloiber, M. Chegini, K. Andrews, T. Schreck. and D. J. Lehmann. Submitted to IEEE Transactions on Visualization and Computer Graphics.

6.1 Introduction

Visual analysis of multidimensional data is challenging. It often involves projecting high-dimensional data to a lower-dimensional space, whilst at the same time attempting to preserve or approximate any structures present in the original high-dimensional (HD) data space. To this end, subspace visualizations and multivariate projection-based data visualizations have been estaeblished to visually explore the data in 2D or 3D space, yet provide hints of possible structures in the original high-dimensional data space. Such structures are typically observed as correlations, co-locations, or distances among the data points. However, multidimensional projections can result in distortions between the original data space and the embedding space (projection view), which introduce uncertainties for visual analysis [139].

In radial axes plots, e.g., RadViz [141] and Star Coordinate [104] visualizations, n-dimensional (nD) data is mapped to a 2D plane by arranging the coordinate axes as circular spokes. Analysts interactively explore the embedding space by manipulating the order and position of the projection axes, which in turn steers the embedding process. Changing the arrangement of the axes results in a new view of the embedded data. The complete embedding space can be explored by iteratively re-arranging the projection axes. However, only one projection can be seen at any one time, and any single 2D configuration of the data points can lead to following misinterpretations:

i) *False Neighbors*: Data points belonging to different structures in HD space might be mapped close to one another in the current embedding, leading to the false conclusion that they belong to the same structure.

ii) *Missing Neighbors*: Data points belonging to the same structure in HD space might be mapped to different positions in the embedding space, giving the false impression of dealing with different data structures.

iii) *Constant Structures*: Data points which form similar structures in a lower-dimensional subspace might form different structure in a higher-dimensional subspace, which cannot be deciphered from analyzing a single embedded visualization.

Simple linear interpolation from one view to the next would produce straight-line trajectories and collapse information from the higher-dimensional space. An interesting observation is that an embedded point follows a specific *path* or *trajectory* in the embedding space, when the axial configuration is smoothly changed. Even more, if two points belong to different structures in the data, the trajectories of the two points often behave differently.

To reveal such structure in the data, we introduce trajectory-based projection paths, which allows observing the impact of individual dimensions by integrating them smoothly into the mapping. The use of projection paths enables an additional encoding for similar structures regardless of spatial position. In our novel approach, analysts rely on data-specific information-bearing trajectories to explore the projection space by gradually changing the subspace view from one state to another and observing the smooth re-arrangement of data points and resulting trajectories. To deal with clutter for large numbers of trajectories, it is possible to restrict the projection paths to specific subsets of data points by using an interactive lens. Moreover, the lens can be used to provide an in-depth trajectory analysis for local data of interest. Informally speaking, analysts may visualize a subspace by considering a subset of dimensions for a subset of data points – variable-to-variable. A data-driven evaluation demonstrates the effectiveness of the approach and overcoming the three issues (i-iii) described above.

6.2 Related Work

In this section, we present work that relates to our approach, and discuss delineations.

6.2.1 Interactive Projection Techniques

A problem of static projection views is that point distances in the projected space may differ from those in the original data space. Also, they do not describe why data points are located nearby or far away. Interactive projection techniques like [37, 48, 98, 189] help users to interpret the meaning of projection views. Stahnke et al. [189] presented an approach to display HD information for data points and clusters, and corrects projection distances, by moving points closer or further away. Dual projection techniques that closely link attributes and projection view [37, 48] allow users to interactively change attributes and discover how the projection will be influenced, and vice versa. This concept has also been applied to multiple coordinated views [98] and linked tables [202]. In [202], observations and attributes are represented as two juxtaposed tables: showing all items with a selection of features; and showing all features with a selection of items. Andromeda [169] is a very similar visual data analysis system, which allows users to change the position of data points and group clusters in the projection space. The system adapts the weights of a model according to defined clusters and uses a linear animation to update the projection view.

Static views cannot explicitly show the relations between different projection view configurations and lead to manual exploration between single views. To this end, animations can help to show how configuration changes will affect data positions in projected space. The ScatterDice approach [54] uses 3D transitions to visualize dimension changes within a scatterplot matrix. To navigate from one view S_{xy} to another S_{xz}, it performs a vertical transition to a new dimension z, applies a 90 degree rotation, and projects it back to a 2D view with x and z. In [39], scatterplots are extended by sensitivity coefficients to highlight local variation of one variable with respect to another. Therefore, sensitivity is visualized by means of velocity so that the resulting visualization resembles a flow field. TripAdvisor^{N-D} [138] is a multivariate data exploration tool that allows users to smoothly tilt the projection plane of subspace views. It provides additional navigation hints and cluster information trajectories – motion trails – to visualize viewpoint changes. Identifying patterns in different subsets of dimensions (subspaces) is also an important task. Many existing techniques visualize and compare patterns in subspaces [12, 43, 61, 62, 89, 196]. However, most techniques use static or single projection views, which may create numerous individual views that are difficult to relate. Liu et al. [125] introduced an interactive framework based on dynamic projections that creates smooth transitions between pairs of projections. A more recent approach by Jäckle et al. [102], called PatternTrails, visually orders and compares patterns between subspaces. We follow these techniques in spirit, but include a continuum of projections and aim at supporting the interactive selection of both subsets of dimensions and subsets of data records.

6.2.2 Interactive Lens Techniques

An early interactive lens technique, called Magic Lens, was introduced by Bier et al. [25], to interactively manipulate and filter objects for specific local areas, while leaving the rest of the scene unchanged for context. This general concept has been widely developed over recent years and is nowadays applied to numerous research areas ranging from multivariate data analysis [53, 65] to volume rendering [151, 210]. In [79], an interactive lens technique, called *ProxiLens*, is used to highlight focus points based on the proximities in the HD spaces. It moves false neighbors to the border of the lens through an interpolation animation. Krüger et al. [111] used the lens technique on an interactive map to identify geospatial areas of interest (AOI) in movement data. Besides initial concepts such as overview+detail, focus+context and geometric/semantic zoom, interactive lenses can also be used to support data analysis such as searching and filtering. In previous work, we suggested a lens approach for regression modeling of local patterns in scatterplots [171]. It can be used for interactive data modeling and incorporates guidance functions, helping to find appropriate models (data subsets) by means of regression quality scores. In [118], a data-dependent magic lens is presented to minimize projection related distortions for RadViz [141] and Star Coordinate [104] visualizations. The latter works inspired us to include the interactive lens concept for a trajectory-based projection visualization. For a comprehensive survey of lens techniques, we refer to Tominski et al. [197, 199].

6.2.3 Delineation of Our Approach

ScatterDice [54] animates transitions between views by performing a 3D rotation while one dimension remains constant. To explore a path along a set of dimensions multiple view rotations are needed, while losing the information of previous stages. The flow-based visualization in [39], uses the derivative of a third variable to show tri-variate correlations. In radial axes plots [104, 118, 141, 181], the interaction is limited by only changing one single dimension at a time. Our approach is able to summarize paths of multiple subspace configurations by drawing a line at offset position given by data coordinates p on the radial axes, and then shape of the path given by the values of additional dimensions. Similar to the Andromeda approach [169], we use animations to visualize the projection change by means of trajectories. In contrast to Andromeda, we constantly update the projection matrix to obtain a more precise description through non-linear trajectories. When changing the projection space, our projection matrix will be adapted step by step to reflect the target dimension subset, and thus creates a meaningful path that describes how a data point is transformed from a lower-dimensional space into a higher-dimensional space or vice versa. This higher-dimensional information is only valid because of the general linear projection matrix and is not available in non-linear projection spaces like T-SNE or MDS. RadViz [141] and Star Coordinates [104] are two of the most popular projection-based visualization techniques with a radial axis arrangement. The main difference between the techniques is a non-linear normalization step in RadViz. However, a comparative study

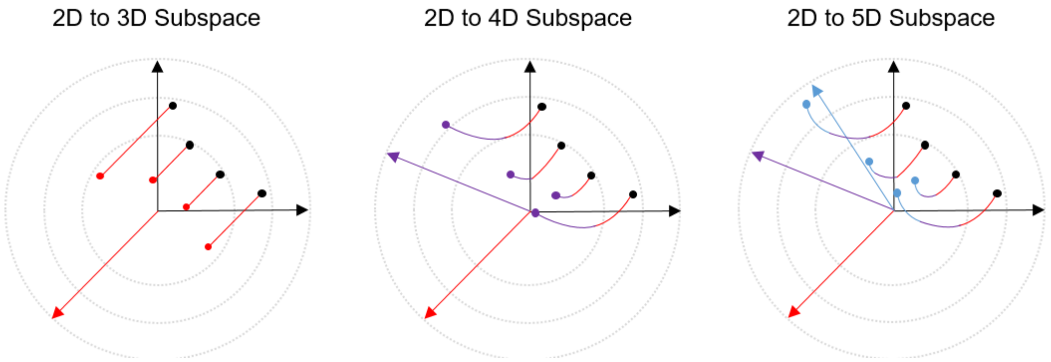


Figure 6.1: Illustration to create integrated projection paths via trajectories. Starting from a 2D view, we visualize the projection change by using smooth trajectories, which describe the impact of added dimensions using the radial visualization metaphor.

between RadViz and Star Coordinates has shown that both techniques have their advantages in data analysis tasks [157]. Our approach combines a linear projection-based plot with non-linear projection paths. A benefit of using trajectories is that the path of each data point's motion is visualized on the screen, making the movement of each point during interaction operations more apparent [98]. In [138] trajectories are used to visualize motion trails of data points when viewpoints are changed. Depending on the data, this may cause clutter in the view space. To overcome this shortcoming, we suggest using an interactive lens tool to focus the exploration on a subset of data points, and thus limiting the number of trajectories. In previous work [171], we extended the lens metaphor to support regression analysis for local areas of interest, by allowing analysts to interactively select a portion of data, on which regression analysis is run. Instead of analyzing bivariate data plots, in this work, we use a lens approach to investigate projection-based patterns in nD subspaces.

6.3 Integration Projection Path Concept

We want to give analysts a way of understanding the influences of different dimensions on the projection. To do this, we allow them to add additional dimensions iteratively and visualize the changes through trajectories. The trajectories are created with a set of intermediate projection matrices which interpolate between the projections of the lower- and higher-dimensional subspaces, i.e., the initial and the extended subspace. We construct intermediate projection matrices using interpolating functions. They are staggered to order the impact of the dimensions and they smoothly increase the dimensions' impact. Figure 6.1 illustrates this idea by adding one, two and three additional dimensions to a 2D subspace (scatterplot). In these cases, the trajectories visualize the impact of the third (red), fourth (purple), and fifth (blue) dimension. Resulting trajectory shapes and point positions reveal properties of the data, depending on the added dimensions and their properties.

On an abstract level, our main concept is based on the idea to start by any projection **A** and to define then a multivariate projection of interest **B** that considers more dimensions as **A** did. Moreover, we sample projections from **A** to **B** to generate the related trajectories and visualize them in an integrated projection view. Please note, going from projection **A** to **B** with a constant interpolation step size, or let us say *constant acceleration*, would end up in linear trajectories, due to the nature of multivariate linear projections. However, linear trajectories additionally visually encodes just distant information of a data attribute to its origin in the data space (the more distant, the larger the line-esque trajectory would get by going from **A** to **B** with constant acceleration). Since we are interested in more detailed information about subspace integrations and changes, we have to go beyond linear trajectory views. From our experience, curved trajectories may help at this point and provide a better comparison for individual data points and their structures in embedded subspaces. Thus, by going from projection **A** to **B** with *non-constant acceleration* creates curved trajectories, see Fig. 6.2 (right). We call this concept *dimension axes acceleration* and can apply it two modes: 1) for all data points at once (*global projection*), or 2) for a selected subset of data points (*local projection*).

6.3.1 Global Projection

The general approach can start from any projection configuration including the coordinate origin –0D, 1D, 2D, or nD. By iteratively adding more dimensions to the projection process, we create a multivariate linear projection and corresponding trajectories from the current projection view to a k-dimensional subspace. Please note that Figure 6.1 starts with a 2D projection that is extended by three further dimensions. The steps for creating the trajectory-based visualization are:

1. Create the final projection matrix for a k-dimensional subspace.
2. Use interpolation functions to generate intermediate projection matrices at different time steps.
3. Apply a sampling for the projections for all time steps and visualize the resulting trajectories, i.e., the covered path of a projected point during the sampling steps.

The following will discuss the definition of projection matrices, the choice of axes for the projection, the acceleration functions, and the creation of trajectories.

Projection Matrix

For a multivariate projection, an n-dimensional projection matrix **A** is a $2 \times n$ matrix. The entries of this matrix are 2-dimensional axes $\mathbf{a_i}$ for each of the n-dimensions: $\mathbf{A} = (\mathbf{a_1}, ..., \mathbf{a_n})$.
The $m \times n$ data matrix **X** for all m data is:

$$\mathbf{X} = \begin{pmatrix} \mathbf{x_1} \\ \vdots \\ \mathbf{x_m} \end{pmatrix} = \begin{pmatrix} x_{11} & \dots & x_{1n} \\ \vdots & \ddots & \\ x_{m1} & & x_{mn} \end{pmatrix}, \tag{6.1}$$

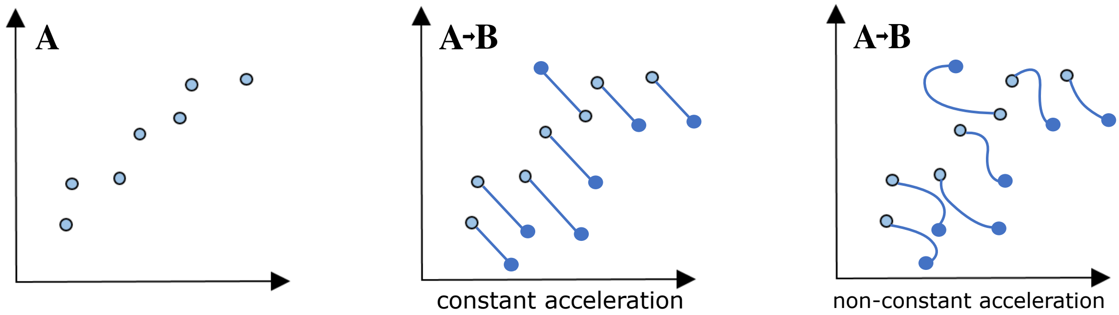


Figure 6.2: Generating curved trajectories by using non-constant acceleration for sampling the projection space from projection **A** to projection **B**.

with the j-th datum is denoted as $\mathbf{x_j} = (x_{j1},...,x_{jn})$. The projection $\mathbf{p_i}$ of a single datum $\mathbf{x_i}$ is defined as:

$$\mathbf{p_i} = \mathbf{A} \cdot \mathbf{x_i}^T \tag{6.2}$$

giving the projection for all data as:

$$\mathbf{P} = \mathbf{A} \cdot \mathbf{X}^T \tag{6.3}$$

To exclude dimensions from the projection, their corresponding axes in **A** are set to zero. For example, a two-dimensional scatterplot, which selects the second and third of three dimensions, is given as:

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{6.4}$$

and, e.g., a data point $\mathbf{x_1} = (2,5,4)$ of the nD data set will be projected to:

$$\mathbf{p_1} = \mathbf{A} \cdot \mathbf{x_1}^T = \begin{pmatrix} 5 \\ 4 \end{pmatrix}. \tag{6.5}$$

Axis Layout

When changing the selected dimensions, i.e. by including further non-zero axes to the projection matrix, the set of projection axes changes according to a predefined layout. For this, the projection matrix must be updated accordingly. Useful projections depend heavily on a wise selection of projection axes for each dimension. Hence, the 2D vectors making up each projection axis **p** must be chosen carefully. Ideally, we want to spread them uniformly while still giving the analyst an intuitive understanding of the trajectories when a new dimension is added. This creates a difficult problem: we need axes to be spread more or less uniformly, but new axes to be introduced should not displace current axes. Therefore, we propose two different approaches: *angular chop*

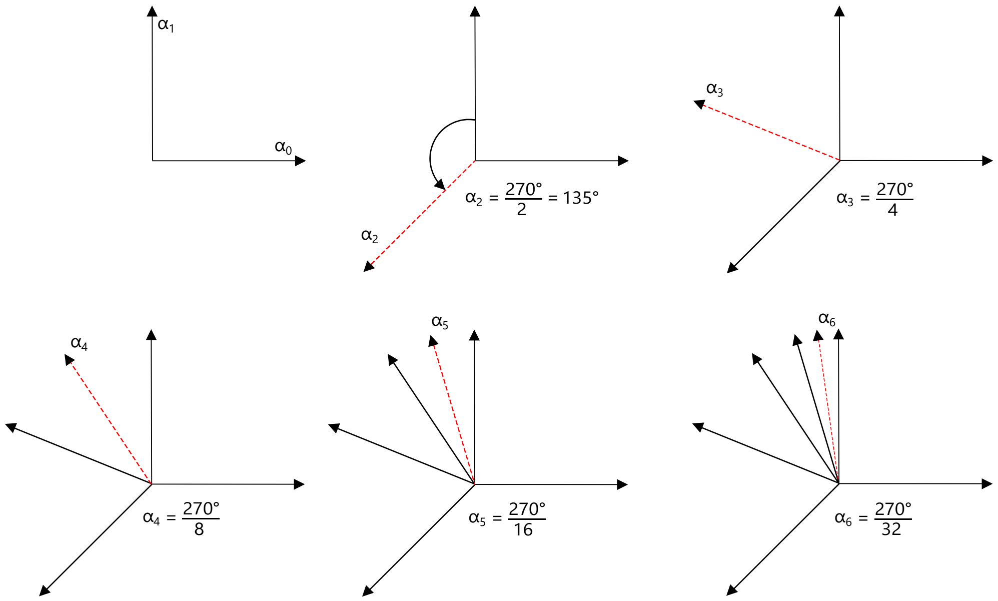


Figure 6.3: Angular chop: Lay new axes such that they halve the angle between the last and the first axis. New axes added per step left-to-right are illustrated by the red dashed line.

and *clockwise fill-in*. Figure 6.3 and Figure 6.4 shows how the axes are laid out for both layout techniques. For both layouts, all axes have unit length and we set the axes of the initial two dimensions to:

$$\mathbf{a_0} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \mathbf{a_1} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{6.6}$$

Angular chop sets all added axes halfway between the first axis and last axis. For k displayed axes, α_i denotes the angle between the i-th and the $(i-1)$-th axis and is defined as:

$$\alpha_0 = 0^\circ, \alpha_1 = 90^\circ \tag{6.7}$$

$$\alpha_i = \frac{270^\circ}{2^{i-1}} \tag{6.8}$$

Clockwise fill-in assigns axes in a clockwise manner, such that the first four axes occupy all 90° rotations and the following axes uniformly fill in the gaps in a circular manner. Axes 5 to 8 are spread between the first four, axes 9 to 16 are spread between the first eight, etc.

Combined with our trajectory approach, these layouts reduce the need for manual interaction when compared to radial axes plots. Since new axes are scaled from zero to unit length, analysts do not need to adjust the length manually to see their impact.

Dimension Axes Acceleration

When a projection configuration changes, we compute intermediate projection matrices to create a smooth animated transition between the initial projection and the final projection. To show the

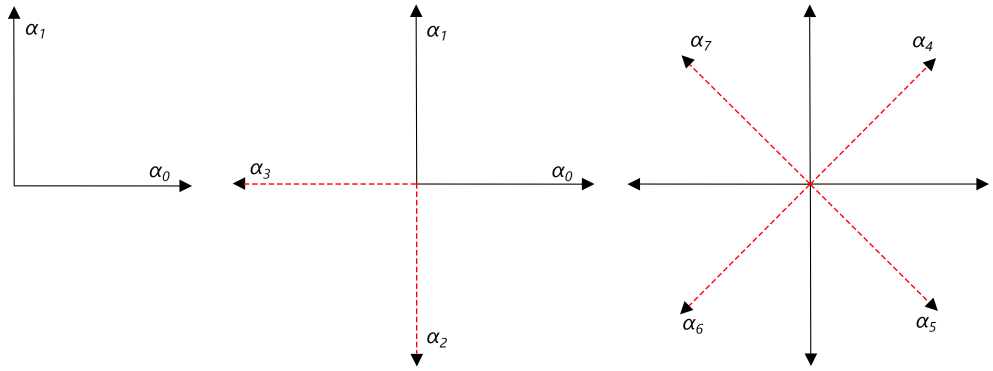


Figure 6.4: Clockwise fill-in: New dimension axes fill in the remaining space in a repeating clockwise manner. New axes are illustrated by red dashed lines and each iteration is labeled by the axes number h.

staggered impact of all changing dimensions, we use varying accelerations for each dimension's axis. This ensures a gradual change for the k selected dimensions. Please note that the intermediate projection matrices, which our approach uses, lead to intermediate positions that are valid finger prints of the data in the projection space (displayed space). A common approach is to use interpolation techniques in the projection space. However, the intermediate positions in this space may not be based on a valid projection matrix anymore, and thus may not be related to the data. Additionally, using interpolation techniques in the projection space introduces a systematic error w.r.t. data.

To achieve a smooth integration of changing dimensions, we use interpolating curves $c : [0,1] \to [0,1]$. These curves are created by using cumulative normal distribution functions that define the acceleration of all axes over time. To visualize a continuous path of all selected dimensions, the acceleration for the dimension axes k are staggered such that the movement of all axes overlaps with each other and increase differently over time. Therefore, we use normal distribution functions with equal variance σ^2 and varying means μ.

Figure 6.5 shows the acceleration functions for $k = 3$ with a total duration of $T = 50$. To stagger them, we assign an evenly spread mean μ of each selected axis k_i that defines the time at which the influence of the interpolating curve may be the strongest. The starting point of not selected axes is irrelevant since their axes are zero. The basis for the projection matrix $\bar{\mathbf{A}}_t$ at time $t \in [0,T]$ is:

$$\bar{\mathbf{A}}_t = (\mathbf{a_1} \cdot c_{t_1}(t), ..., \mathbf{a_n} \cdot c_{t_n}(t)) \tag{6.9}$$

We can now compute the trajectories by evenly dividing the time t and using $\bar{\mathbf{A}}_t$ in Equation 6.2.

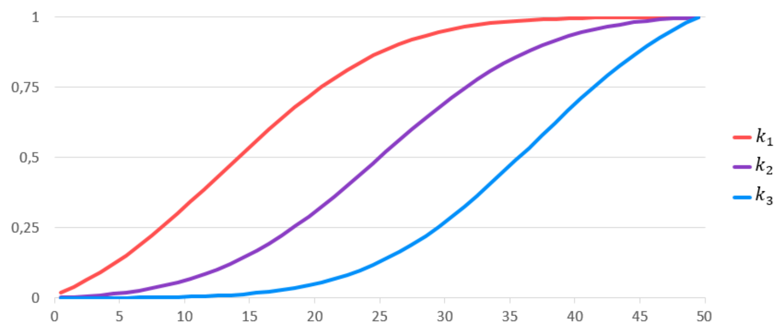


Figure 6.5: To compute smooth and staggered trajectories, we use cumulative normal distribution functions with equal variance σ^2 and varying means μ as acceleration functions. This example showcases the acceleration for $k = 3$ and a total duration of $T = 50$.

6.3.2 Local Projection of Subsets

As can be seen in Figure 6.7a, the global projection might suffer cluttering issues when longer and/or many trajectories are displayed. To give analysts focus+context while exploring the visualization, we include an interactive lens approach to our system. It allows users to focus only on subsets of the data, by selecting a rectangular region in the current projected space, for which they can locally change the selection of dimensions. The trajectories for this change of dimensions are computed in the same way as the global projection, but only trajectories originating from the focus region are visualized. Figure 6.6 shows this lens and the selection of dimensions in our exploration system (for more details, please see the video material).

6.4 Visual Representations and Interactions

Next, we present our prototype system for the exploration of projection-based patterns in n-dimensional subspaces. Figure 6.6 shows an overview of our projection system and demonstrates a subspace integration via lens selection.

6.4.1 System Overview

Our system is based on two interconnected projection components: The *Global Projection View* and the *Local Projection Lens*. Analysts can change the global projection (for all data points) by adding or removing dimensions via the *Dashboard* on the left. The lens is an additional view inside the global projection view for investigating projection changes of selected subsets of data. These two projection approaches allow analysts to interactively compare different subspace configurations on a global and local level. In the following, we discuss each component in more detail.

The **Global Projection View** displays subspace configurations and changes via animated projections paths. A projection change can be visualized by two different approaches, either by

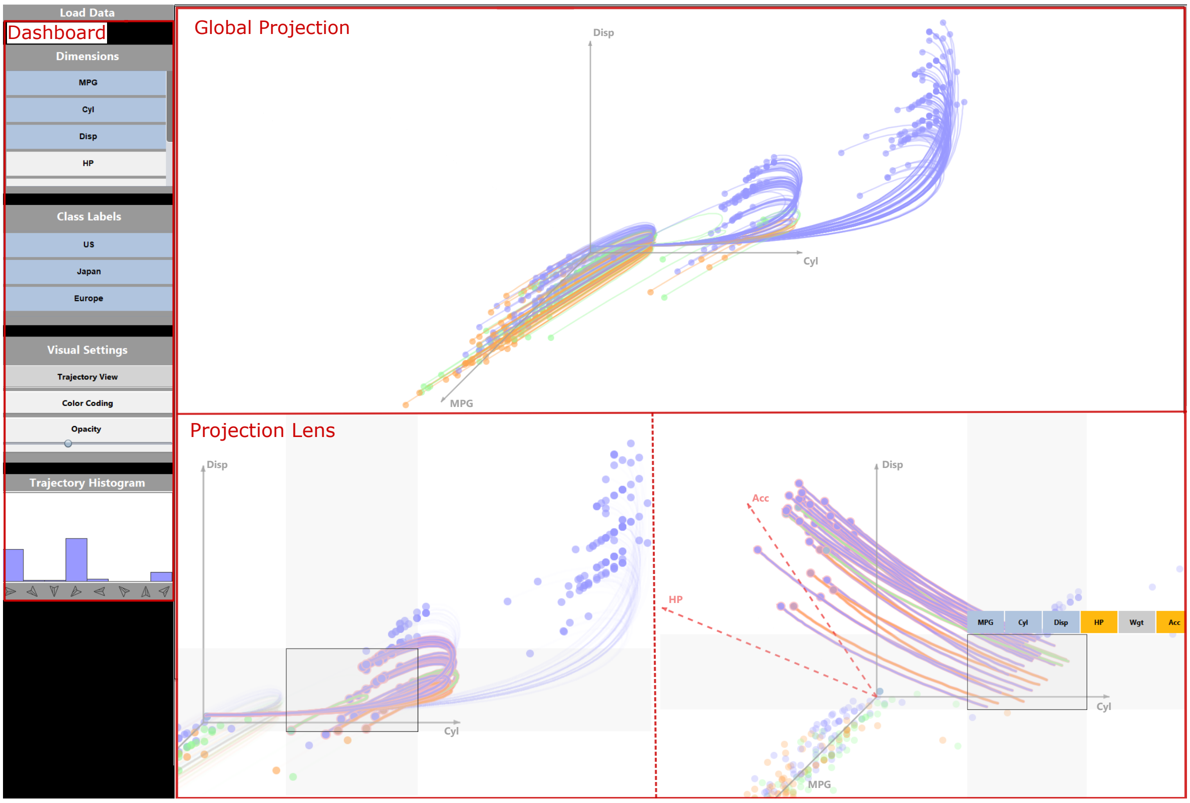


Figure 6.6: Demonstration of our trajectory-based techniques for global and local projections. The Global Projection view shows a subspace of a car dataset with the dimensions *Cylinder*, *Displacement*, and *Miles per Gallon*. Consistent trajectory structures intensify the correlation of data points. The Projection Lens can be used to interactively explore projection paths for subsets of data and include additional dimensions to compare subspaces changes for a given set of points. The colors of points and trajectories correspond to origin of the cars.

showing the continuous projection paths or by sequential projection paths. *Continuous projection paths* show the complete progress of subspace configurations starting from the coordinate origin (point O) such that the trajectories comprise the impact of all selected dimensions (see Figure 6.6 – Global Projection). The *sequential projection paths* visualize the projection change from the latest subspace configuration to the new configuration and allow analysts to investigate subspace changes more interactive and with successive transitions (see Figure 6.6 – Projection Lens). Through this, analysts can create projection paths for any projection to a final nD subspace. Furthermore, we link the projection axes to the projection matrix and show for each time step the actual growth of the dimensions (c.f. Section 6.3.1).

The interactive **Projection Lens** can be placed over the Projection View and can be dragged around. Thus, analysts can easily define a rectangular region of interest as the lens selection and interactively move the lens on the Projection View for further analysis of individual subsets or clusters. If the lens is active, only the projection paths of selected points are highlighted. This allows analysts to focus on local regions of interest and avoid cluttered views. When interesting clusters are found in one subspace, the lens can be used to apply projection changes to the selected data and see if the cluster remains constant in other subspaces. As in the Global Projection View, analysts can choose between the continuous projection paths or sequential projection paths. We currently use a rectangular selection scheme, but other forms such as circular or free-form selection are straightforward to implement. The advantages of rectangular selection are discussed in our previous work [171].

The **Dashboard** provides several menu items to configure the global projection view, e.g., dimension selection, filter functions based on class labels in the data (if available), or the following visual settings to improve the visibility of the trajectories. For instance, to highlight the similarity of trajectory clusters, a color-coding based on the trajectory curvature can be used. To prevent cluttered views, the opacity of points and trajectories can be reduced (see Figure 6.7a and 6.7b). To point out changes that emerge through subspace transition, the directions of trajectories are accumulated and visualized in a histogram (see Section 6.4.2).

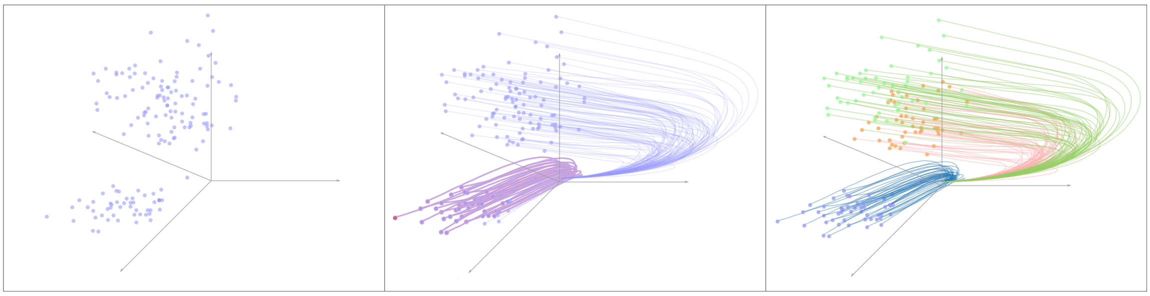
6.4.2 Structure Similarity

Similar trajectory structures can indicate a relationship within the chosen dimensions of the subspace. In fact, for a single multivariate projection it applies that data belonging to different nD structures might be mapped to similar positions in the projection space – False Neighbors (i). The opposite statement – Missing Neighbors (ii) – is also true (c.f. [117]). Thus, spatial closeness in the projection space does not necessarily correspond to structural similarity in nD (w.r.t. multivariate projections) and it introduces a source of misinterpreting the data. Following the Cramer-Wold-theorem [46], a sample of the projection space (i.e. a single projection) consists some amount of information about the original nD data and some amount of projection noise based on artifacts caused by properties of the used projection technique. Moreover, integrating continuous projections spaces enhance the information-part and suppress the noisy-part till its convergence, where all information of the original nD data are reconstructed. However, for our integrated views, which are based on a continuous set of projections, the resulting structures that appear to be similar in the trajectory view in the projection space might point to similar structures in nD, even if they are not spatial correlated. This property supports to resolve the above mentioned source of misinterpretation and this is why we subsequently focus on structure similarity.

Two different approaches are supported for comparing trajectories: 1) similarity search and clustering based on trajectory metrics and 2) visual exploration of trajectory curvature.

Similarity Based on Trajectory Metrics

To find similar nD structures, trajectory metrics based on *length*, *principal direction* and *curvature* can be used. Since trajectory length strongly expresses the influence of a dimension, we define a



(a) Projection view without projection paths.

(b) Projection view including projection paths and similarity search.

(c) Projection view including trajectory clustering.

Figure 6.7: The trajectories show the movement from the origin to the projection space and reveal important information about the selected dimensions. The paths can help to find similar or dissimilar nD structures in the data and provide insights about class affiliations. (b) shows that the trajectories of the class *Setosa* are rather similar (query object is colored in red). (c) by using a trajectory clustering one can discover different structures according to the three Iris classes.

metric that is not scale invariant to length. The length of a trajectory is computed by

$$l = \sqrt{\sum_{t=1}^{n} dx_t^{\,2} + dy_t^{\,2}} \tag{6.10}$$

where dx and dy represent the difference in coordinates at time t. The principal direction of a trajectory indicates its main structure and is computed by the average angle between trajectory segments.

$$\theta = \frac{1}{n}\sum_{t=1}^{n} \alpha_t \tag{6.11}$$

$$\alpha_t = arctan(dx_t, dy_t) \tag{6.12}$$

Additionally, we compute the curvature κ to distinguish deviations in trajectories. It is computed by taking the average angular deviation between trajectory segments.

$$\kappa = \frac{1}{n}\sum_{t=1}^{n-1} \alpha_t - \alpha_{t+1} \tag{6.13}$$

Finally, all trajectories within a threshold ϵ are considered similar. In our experiments, we chose a threshold of $\epsilon = 0.2$ and used the k-Means algorithm for clustering. Note that k is selected according to the number of classes in the data.

Figure 6.7 showcases different views of a subspace visualization of the Iris dataset: (a) without projection paths, (b) with projection paths and visual search results, and (c) with trajectory clustering. By considering the subspace configuration without trajectories (a), one might think that only two clusters exist –the upper cluster and the lower cluster. To verify potential clusters a similarity search can be used to discover missing neighbors (ii). The similarity search result in (b) shows that the lower cluster contains similar structures that are mapped close to one another. This might be an indicator of a compact cluster. However, (c) reveals that the upper cluster contains data points with different structures mapped close to one other (false neighbors (i)), which may not belong to the same class. In fact, the two trajectory clusters represent the data points of the class *Versicolor* and *Virginica* which are not linearly separable from each other. Note that trajectory color represents cluster results and point color represents the actual classes.

Visual Exploration

To visualize subspace changes, we use animations of points and trajectories in embedded spaces. Furthermore, we provide brushing and linking interactions to highlight and filter selected subsets. To show the degree of impact on subspace changes, the principal direction of trajectories can be considered. If a subspace change had a comparable impact on many data points, the principal direction of the corresponding trajectories would tend to be comparable, too. On the contrary, if it had a nonuniform impact the principal direction would vary strongly. To indicate the degree of impact, we visualize the principal directions of trajectories that emerge from the global or local projection view by histograms. The histogram contains eight bins for eight directions and shows the number of cases in each bin. An example of the histogram is shown in Figure 6.6 – lower left corner. The histogram here, summarizes the main changes that emerge in the global projection views. The uneven distribution in the histogram indicates that the projection configuration may contain several structures. Furthermore, it is useful for verifying common patterns in case of large numbers of cluttered trajectories. It might also be useful as a search modality. For example, the system could search for dimensions which when added, will produce certain directional changes, which analysts could indicate by specifying a direction distribution, or a sketch. We leave the latter option for future work.

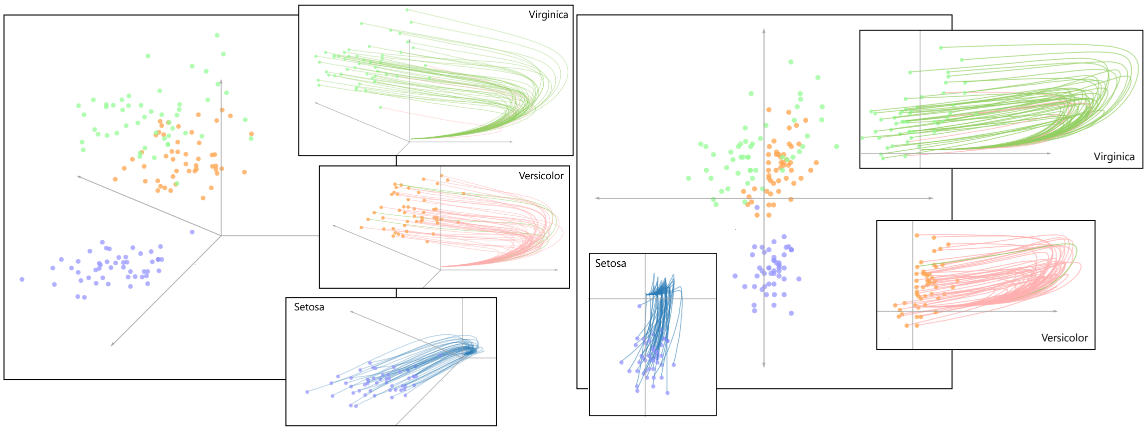
6.5 Evaluation

In this section, we provide a cluster validation as proof of concept and show use cases that address the issues (i-iii) described in Section 6.1.

6.5.1 Proof of Concept

To demonstrate the usability of our approach, we show how projection paths can be used to find similar structures in subspace configurations and cluster the data according to them. On the one hand, we compute cluster quality measures (intra-cluster distance) to compare the results based on the trajectories against data space, and on the other hand, we validate the class affiliations of the resulting clusters. Therefore, we use the two classified datasets *Iris Flower* and *Wisconsin Diagnostic Breast Cancer* (WDBC) from the UCI data repository[1]. The Iris flower dataset consists

[1] https://archive.ics.uci.edu/ml/index.php



(a) Angular Chop

(b) Clockwise Fill-In

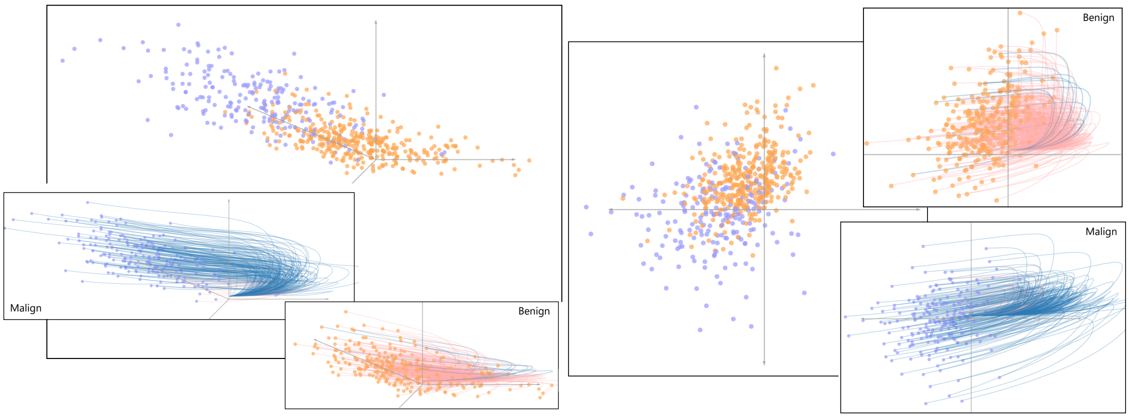
Figure 6.8: Cluster validation of projection path with the Iris dataset. Figures (a) and (b) show two projection views with different layout settings. In both views, the classes *Versicolor* (orange dots) and *Virginica* (green dots) have high overlap in the projection space and can not be well separated. A clustering based on trajectory similarity may help to separate classes in the data.

		C1	C2	C3	Total
Angular Chop	TP	50	46	46	**142**
	FP	0	4	4	**8**
	Dist.	0.17	0.19	0.22	**0.58**
Clockwise Fill-In	TP	50	45	49	**144**
	FP	0	1	5	**6**
	Dist.	0.17	0.22	0.20	**0.59**
Data Space	TP	50	42	40	**132**
	FP	0	10	8	**18**
	Dist.	0.17	0.23	0.20	**0.60**

Table 6.1: A detailed comparison between cluster results based on projection paths and data space (c.f. Figure 6.8). Both cluster results based on projection paths achieve better class affiliation (True Positives TP, False Positives FP) and a more compact intra-cluster distance (Dist.).

of 4 dimensions with 150 records and gives measurements of the sepal as well as the petal length and width for three iris species. The WDBC dataset consists of 30 metric dimensions with 569 records that describe a set of attributes of cell nucleus measurements that are revealed from breast cancer patients. The data is classified into malign and benign cells.

In Figure 6.7, we already provide a brief insight into how projection paths help to explore common structures and identify false neighbors (i) with the Iris dataset. Now, we will verify the cluster results in detail. Figure 6.8 shows the same data projection with the *Angular Chop* and *Clockwise Fill-In* layout including cut-outs of the three Iris classes. In both projection views the class *Setosa* is well separated in the space whereas the classes *Virginica* and *Versicolor* have a high overlap. The color-coding of the trajectories visualizes the cluster results. A close



(a) Angular Chop

(b) Clockwise Fill-In

Figure 6.9: Another cluster validation with the WDBC dataset. Here, we demonstrate the results on a subspace of the attributes texture mean, concave points (mean), perimeter (worst), and concave points (worst). Also in this case, trajectories can help to find class-related data points in overlapping projection spaces. Pink trajectories of the benign class and blue trajectories of the malign class are clustered correctly. Detailed results of the clustering can be seen from Table 6.2.

		C1	C2	Total
Angular Chop	TP	193	346	**539**
	FP	11	19	**30**
	Dist.	0.26	0.18	**0.44**
Clockwise Fill-In	TP	185	356	**541**
	FP	1	27	**28**
	Dist.	0.25	0.19	**0.44**
Data Space	TP	181	354	**535**
	FP	3	31	**34**
	Dist.	0.19	0.25	**0.44**

Table 6.2: Details of the cluster comparison with the WDBC dataset (c.f. Figure 6.9). C1 refers to the cluster of malign cells and C2 to the cluster of benign cells.

look at the cut-outs of classes Virginica and Versicolor shows that only a few data records are clustered incorrectly. The detailed results of the clustering are shown in Table 6.1. It shows that the projection path of both layouts provides valuable features to cluster the data and discover class-related structures. More specifically, the cluster results based on projection paths score a precision of over 95% (AC: 142 of 150 and CW: 144 of 150) and achieve a slightly better intra-cluster distance compared to the results based on the data space.

To confirm the results, we repeat the experiment on a subspace of the WDBC dataset, which is shown in Figure 6.9 and Table 6.2. As in the first experiment, both projection views contain overlapping data points with different structures in the projection space. This time one can see how the different structures related to the classes malign and benign cells. The detailed results

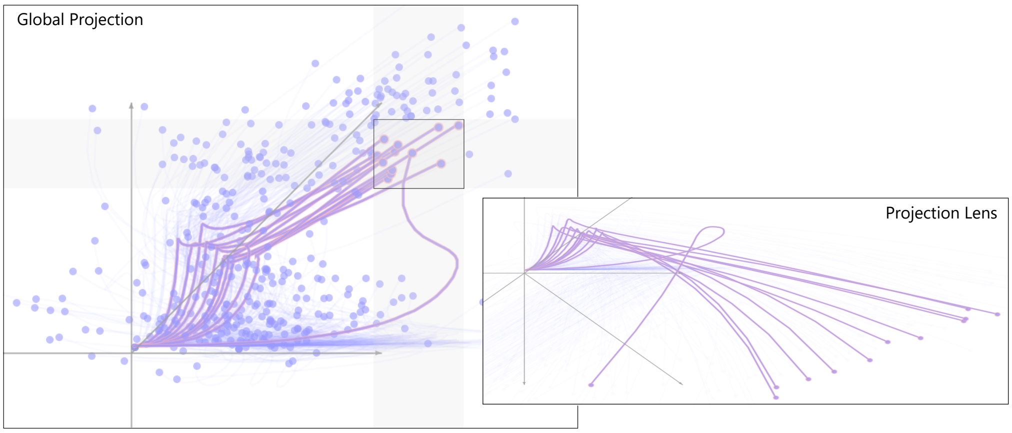


Figure 6.10: False neighbor in lower-dimensional space. A data point with a distinct structure is mapped closely to a cluster with different structures. By extending the subspace with further dimensions, one can see how the structure and distance differentiate.

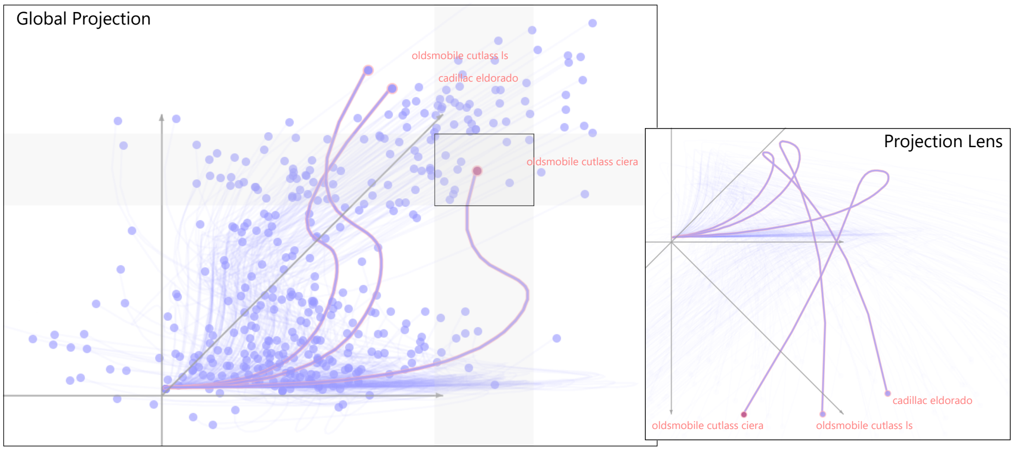


Figure 6.11: Missing neighbors in lower-dimensional spaces might be mapped together more closely in higher-dimensional spaces. The Projection Lens view shows that the false neighbor from Figure 6.10 is actually a missing neighbor that forms similar structures and close distance to two other cars.

in Table 6.2 show that most data points of the two classes can be clustered correctly according to the trajectories. Furthermore, it shows that a clustering on both projection layouts achieves better clusters that reflect the classified cells without a loss of intra-cluster distance.

6.5.2 Use Cases

Next, we provide use cases to analyze *false neighbors* (i), *missing neighbors* (ii), and *constant structures* (iii) in unclassified data. The first dataset that we investigate is a car dataset from the UCI Repository[1] that contains 8 dimensions, such as horsepower, weight, and origin, for 398 cars. As an alternative, we use the *Better Life Index* from OECD[2] wit 24 dimensions, which describe

[2] http://www.oecdbetterlifeindex.org/

well-being factors of 39 countries based on 11 topics. The topics include, for instance, housing, income, education, environment, and life satisfaction of the year 2017. Due to the better cluster results in Section 6.5.1, the following investigations are performed on the Clockwise Fill-In layout.

Car Dataset

Figure 6.10–Global Projection shows an integrated projection view of the car dataset based on the attributes *Miles per Gallon*, *Displacement*, *Horsepower*, *Acceleration*, and *Weight*. By using the lens, one can identify a portion of data where a projection path of one data point has an outstanding structure and is mapped closely to the data points with other structures. The Projection Lens cutout in Figure 6.10 shows that the data point also form a single structure in a higher-dimensional subspace and is mapped further away as before. This situation clearly reflects a *false neighbor* in a lower-dimensional subspace. Our next step will be to investigate the data point with the outstanding structure. A similarity search on the data point (Oldsmobile Cutlass Ciera) shows that two cars with similar structures are mapped further away, see Figure 6.11–Global Projection. By applying the same subspace change as before, it turns out that the structure of the three cars remains similar in a higher-dimensional subspace and form a compact cluster. Further analysis showed that the three cars also form similar structures in other subspace configurations with varying distances in the projection space. However, by just considering the point distance in the projection space, one might overlook the missing neighbors in the lower-dimensional space.

OECD - Better Life Dataset

In Figure 6.12, we start the exploration by comparing countries in a projection view based on *Personal Earnings*, *Years in Education*, *Air Pollution*, and *Water Quality*. In the Global Projection, the countries are well distributed over the projection space, making it difficult to spot clusters based on the point distance. However, by hovering the Projection Lens over the projection space, it becomes apparent that some countries like the Czech Republic, Poland, Slovenia, Slovakia and Hungary (Cluster A); Finland, Sweden, Estonia and Latvia (Cluster B); and Germany, Austria, Great Britain, Denmark and the Netherlands (Cluster C) form distinct structures and might be separate clusters. To confirm this hypothesis, we change the subspace configuration for the defined clusters to see if their structures and point distances remain similar in higher-dimensional spaces –constant structures.

Figure 6.13 shows the projection change when adding the attributes *Life Satisfaction*, *Household Financial Wealth*, and *Long-Term Unemployment*. By taking a closer look at the lens selections, one can see that the attributes about financial wealth and life satisfaction lead to strong changes in distance and structure similarity of the data points. In Cluster A the countries Poland and the Czech Republic show a slight difference compared to the countries Slovenia, Slovakia, and Hungary. Cluster B is clearly split into two subclusters that separate the Baltic

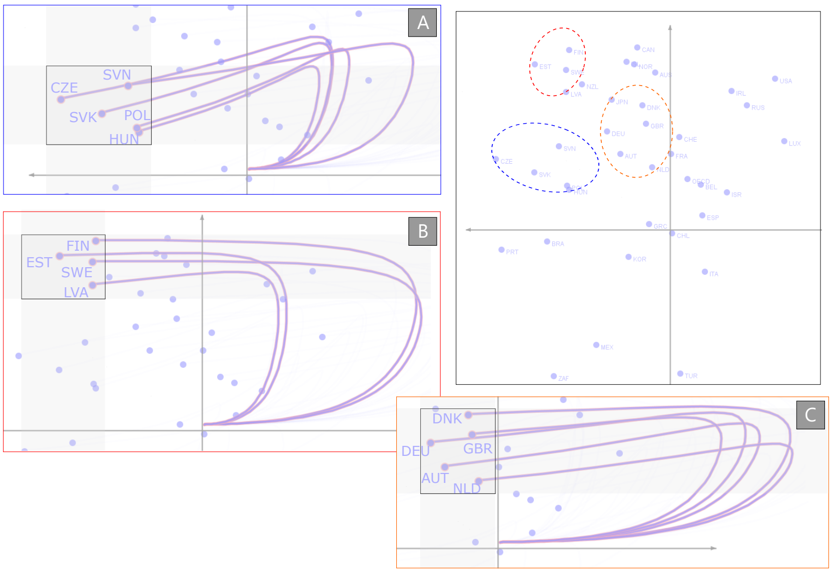
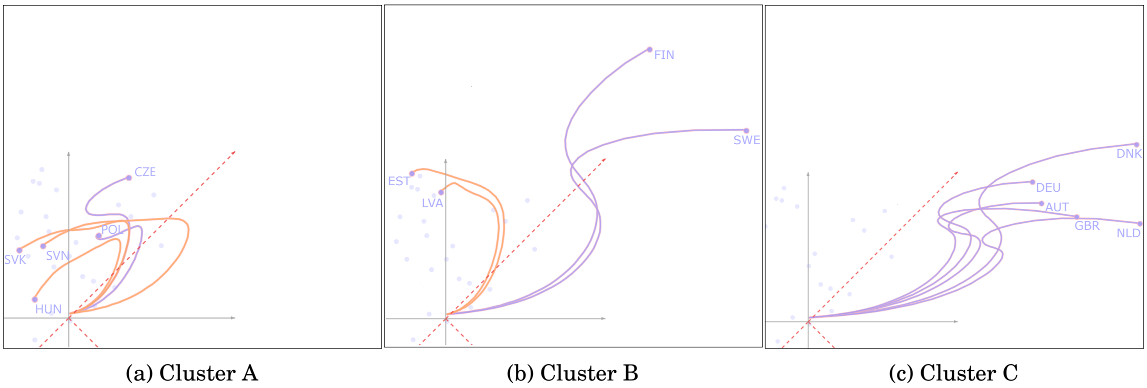


Figure 6.12: Discovering compact clusters in subspaces can be a difficult task. Our Projection Lens approach helps to discover cluster structures in lower-dimensional spaces.

States, Estonia, and Latvia, from the other two Scandinavian States. Finally, the countries from Cluster C remain rather similar. An interesting observation is that the subspace change based on financial wealth and life satisfaction also corresponds to the geographical distances between the countries. For instance, one can identify different structures between Eastern European countries, Central European countries, Baltic States, and Scandinavian States.

One could conclude from this finding that these Eastern European countries (orange trajectories) have a more similar classification regarding *Life Satisfaction*, *Household Financial Wealth*, and *Long-Term Unemployment*. Furthermore, this representation makes clear that Cluster C is the only cluster that remains stable in the changed subspace.



(a) Cluster A (b) Cluster B (c) Cluster C

Figure 6.13: Once interesting clusters are found by the lens, a subspace change to a higher-dimensional space can be directly applied to the selected data points. The lens selections (a)-(c) show how different clusters are effected by applying the same subspace change.

6.6 Discussion

Projection paths can integrate additional information of subspaces in projection views and offer more details when compared with single projections, or sequences of projections, like Projection Pursuit [61] or Grand Tour [12]. This gives analysts a more intuitive understanding how newly added dimensions will influence a given subspace, and support to argue for dimension choices for analysis. It enables a new way to analyze subspaces in a hierarchical manner by interactively extending the embedding space for a subset of points. Thus, analysts can validate structures for a given region across multiple subspaces and see, for instance, if a structure remains robust in other subspaces (i.e., variable-to-variable analysis). The similarity between trajectories can affirm a relationship between respective data points in a subspace. Therefore, a similarity-based brushing and linking technique is provided to help analysts in identifying a subset of the data of concern. This similarity-based search can validate or falsify the existence of specific structures in an n-dimensional subspace.

Projection Paths. By using vector layouts in combination with dimension axes acceleration functions, we create non-linear projection paths, which reveal complex data structures that are not visible through linear interpolation. The projection paths are closely linked to a projection matrix to minimize distortion between intermediate steps. This can not be applied to other dimension reduction techniques like PCA or T-SNE, since there the embedding space may completely change for different subspaces and projection paths have no relation to the previous view. The trajectories highly depend on several factors like dimension order, layout and acceleration function, and there exist infinitely many trajectories for one pair of points. For instance, a projection path of a view $P(a,b,c,d)$ is not equal to $P(a,b,d,c)$. However, experiments have shown that the trajectories of compact clusters also remain similar in different subspace configurations. The order and layout of dimensions will influence the general shape of the trajectories, but nevertheless common structures in the data can be revealed (see supplementary material).

Collapsing Dimensions. In most animation-based approaches the exploration and comparison of subspaces is limited to iterative and subsequent changes of subspaces, and thus causes difficulties to address all the aspects (i-iii) on for arbitrary subsets. Furthermore, most techniques use a simple linear interpolation to visualize the subspace change from one view to the next. If the altered view contains more than 2 dimensions, the interpolated trajectory would collapse the information from the higher-dimensional space. By using our integrated projection paths, one can smoothly observe how dimensions collapse (dimensionality decrease) or pan out (dimensionality increase). When the dimensionality increases in radial axes plots, the points might not move at all because the value along the axes can cancel each other out, especially if the values of these data points are normally distributed in all axes. Due to visible trajectories, projection paths can even visualize different structures between the identical start and end points when a cancel-out effect occurs.

Clutter. The intention of integrated projection paths is to provide additional information about the embedded dimensions in the projection space. A positive side-effect of the non-constant time sampling is that it creates cured trajectories, which can make similar trajectories easier to spot. Furthermore, by introducing the interactive lens concept, we provide analysts with a tool to locally explore trajectories while avoiding global clutter issues. Analysts can easily move the rectangular lens to observe trajectories of a subset of records that are located in the same neighborhood in the projected subspace. By using the lens selection, analysts can explore spatial point distances in the projection view, filter out the noise of irrelevant data points and trajectories, and integrated projection paths for selected points.

Limitations. Although both proposed layouts allow the system to smoothly integrate new dimensions into subspaces, their radial nature limits the maximum number of dimensions that can be included in the visualization. An increasing number of axes in these two layouts will lead to data overlapping. In our experiments, we achieved good results for subspace configurations up to 8 dimensions. This limitation is related to screen size, screen resolution and the analysts' perception abilities. Moreover, the presented approach creates trajectories by means of acceleration functions based on equally weighted vectors. As an alternative, it is possible to develop multiple trajectories based on different vectors' length and layouts to reveal hidden aspects of changes when new dimensions are integrated into a subspace. More research is needed to compare different layout techniques and their effectiveness, to discern trajectory patterns and to interpret them. For now, we consider the layout technique as a modular part of our approach.

Future Work. Future extensions include guidance concepts which could enhance the exploration and search functionalities. For example, based on interest measures for trajectories, the system might automatically suggest views (dimension selections) that lead to a more complete coverage of the projection space seen by the analyst. Projection quality measures, such as Projection Precision [165], could be computed based on (i - iii) and visualized in the projection view to communicate information loss in other subspaces. Also, based on subspace search methods [196], the number of combinations of dimensions could be reduced to a smaller set of candidates to consider iteratively. Experiments with layout algorithm (e.g., used in [181]) could be carried out to discover better layouts that result in more compact paths without losing information. We also plan to conduct a user study to confirm and describe the effectiveness of our approach, and find out how users interpret data structures in HD spaces, based on projection paths. This study should be based on comparisons with different state of the art techniques, which might show how animated trajectories will help to build mental models of data structures in comparison to static views.

6.7 Concluding Remarks

The visual analysis of high-dimensional data remains a challenging research topic. Existing visualization techniques typically show specific views which may approximate patterns in HD spaces. We presented a novel projection-based visualization approach that integrates the projection paths obtained from subspace configurations, and a continuum between them achieved by a gradually changed weighting function. We argued that the resulting paths comprise more and meaningful information about patterns in the subspaces, as considering only the two subsets alone. We showed via analysis use cases, that the resulting trajectory views together with appropriate interaction techniques are useful to verify clusters in HD data space, and analyze their structure when changing subspace configurations.

Part III

Visual Guidance and Recommender Systems for Exploring Data Patterns

CHAPTER 7

Guidance Concepts for Visual Exploration and Retrieval of Scatter Plots

Contents

The last part of this thesis focuses on visual guidance and recommender systems that support analysis in large data spaces. These systems can be very helpful when users get stuck in their analytical tasks and are not making progress by using standard retrieval or exploration techniques. Guidance can be provided in a number of ways, for instance by suggesting data visualizations based on the input data or automatically identifying subsets or features based on the users' interest. The basic idea here is that the system monitors the user interactions, like data selections during the exploration or at query formulation, to provide hints, recommend views and guide the user in critical situations.

Due to my previous work experience with visual search techniques (see Part I), I noticed that visual search techniques can be enormously rewarding if users are aware of interesting patterns in the data. However, one drawback of visual search techniques like in sketch-based searches is that the pattern sketch needs to be precise and available in the data to achieve good results.

In exploration tasks, the major challenge is to discover relevant views from a probably large exploration space and to properly use the provided analysis tools to find new insights. Visual guidance can be used to support such tasks by applying real time comparisons and evaluations of user inputs, and thus guide the analysis process by suggesting query templates or data selections.

In this chapter, I discuss several approaches to guide exploration and retrieval tasks in scatter plot data. The guidance concepts are built on top of previously mentioned techniques, i.e., sketch-based search, interactive lens and motif dictionary. To support the visual search process, I propose a shadow overlay technique that provides suggestions for possibly relevant patterns while query drawing takes place. The regression lens concept from Chapter 5 is extended with visual feedback on the quality of candidate models as it is interactively navigated across the input data. While the regression lens can be used for fully interactive modeling, it also provides user guidance suggesting appropriate models and data subsets, by means of regression quality scores. Inspired by the well-known tf×idf-approach, I compute local and global quality measures based on frequency properties of the local motifs to recommend potentially interesting views for exploration.

This chapter is based on:

[171] **Interactive Regression Lens for Exploring Scatter Plots,** L. Shao, A. Mahajan, T. Schreck and D. J. Lehmann. *Computer Graphics Forum, Eurographics Conference on Visualization (EuroVis). The Eurographics Association and John Wiley & Sons Ltd., 2017.*

[178] **Guiding the exploration of scatter plot data using motif-based interest measures,** L. Shao, T. Schleicher, M. Behrisch, T. Schreck, I. Sipiran and D. A. Keim. *Journal of Visual Languages & Computing, 2016*

[173] **Guided Sketching for Visual Search and Exploration in Large Scatter Plot Spaces,** L. Shao, M. Behrisch, T. Schreck, T. von Landesberger, M. Scherer, S. Bremm and D. A. Keim. *EuroVis Workshop on Visual Analytics (EuroVA). The Eurographics Association, 2014.*

7.1 Introduction

Typical tasks of data scientists are for instance exploring large data sets and searching for interesting patterns. Therefore, it exists a number of exploration tools, search techniques and visual analytics systems that support the analysts in their tasks. An efficient method to find interesting patterns in the data is by using visual search approaches like sketch-based search. It allows analysts to visually specify a query of interest and limit the exploration space without any indication of metadata, e.g., dimensions or timestamps. In case that analysts are not searching for a specific pattern, interactive exploration tools can be used that provide a greater degree of freedom with respect to discover unknown patterns.

However, these tools are often designed for domain experts, which are familiar with the data and the visual representation of interesting data patterns. Inexperienced users are often faced with difficulties in finding a good data selection or an accurate search query. If we consider a sketch-based search, analysts must be able to provide an accurate abstraction of the requested pattern to achieve good results. Small deviations between user sketch and target pattern can affect the final result. Moreover, the general procedure to sketch a pattern could be a challenge for many people too. Query interfaces may provide various drawing tools, e.g., freehand, line or shape drawing tools, that have different benefits to sketch queries. Finally, if the query sketch is not available in the view space the search process can quickly become to an inefficient exploration. The same applies to interactive exploration techniques. If a data selection includes poor data items, e.g., outliers, or miss important items may heavily influence the analysis results.

To explore large data sets more efficiently, visual analytics systems including overview visualization and detail-on-demand views can be used. However, in exploration tasks, analysts often have to go over large collections of views, compare individual views with each other and finally, decide which finding could explain a given phenomenon. A common problem here is that the exploration of high-dimensional data sets may include exhaustive comparisons between a multitude of views. Therefore, quality metrics can be used to filter and rank large exploration spaces, and provide a good starting point for exploration. However, most interestingness measures of quality metrics focus on global properties and do not consider the impact of local patterns, which can lead to novel insights. By considering local areas of interest the search space will additionally increase and makes the analysis task even more complicated. Consequently, a comparison of all subsets over the whole search space may by infeasible or unreasonably expensive to achieve. Visual guidance concepts could improve the interactive process of finding interesting data subsets by continuously monitoring user selections and make suggestions to improve data selection. Furthermore, recommender systems including quality metrics based on local patterns could help to reduce the large view space and facilitate the exploration plots containing significant local patterns.

In this chapter, we present novel guidance concepts to support sketch-based searches and interactive exploration for scatter plots. The sketch-based search from Chapter 3 is extended by a so-called shadow drawing component, which continuously executes similarity searches in the background while the user is sketching the query. Thus, potentially matching patterns can be overlaid as drawing templates on the user interface and help to sketch available patterns –like an auto-completion text retrieval. The second guidance function supports subset selection of the regression lens from Chapter 5. Since the regression lens can be used for fully interactive modeling, it may occur that users include outliers or exclude important records that influence the quality of local models. The selection guidance helps to optimize the selection area of the lens by indicating outliers and suggesting translation directions to improve the quality of regression models. Finally, we propose a recommender system based on the motif dictionary from Chapter 3 to suggest scatter plot views derived from *local* data properties. Inspired by the well-known tf×idf-approach, a quality measure is introduced that considers the frequency of local motifs in relation to the whole exploration space to recommend potentially interesting views for exploration.

We extended the previously presented techniques with the following guidance approaches:

- Visual drawing assistant for sketch-based search of scatter plot patterns.
- Data selection guidance of the interactive Regression Lens.
- Exploration suggestions of scatter plot views based on local motif dictionary.

The remainder of this chapter is structured as follows. We discuss related guidance approaches and recommendation techniques for scatter plots in Section 7.2. In Section 7.3, we introduce our visual guidance approaches that support users in sketching visual queries and selecting scatter plot patterns. In Section 7.4, we give an overview of the scatter plot recommender including technical details for computing interestingness measures based on local motifs. Next, in Section 7.5, we discuss limitations and a range of possible extensions. Finally, Section 7.6 concludes the paper.

7.2 Related Work

Many works support the exploration of large scatter plot data sets by means of quality and interest measures. In addition, there exist several guidelines and guidance approaches for the analysis of scatter plot. We next review a selection of related works in the context of our guidance approaches.

Quality Metrics for Scatter Plot Visualization

Automatic identification of interesting candidates within large sets of scatter plots has recently been an active field of research. The Scagnostics method [216] is a well-known feature-based approach, which proposes a set of graph-based measures for scatter plots, to describe the data properties. It can be used to analyze the structure of scatter plots, e.g., density, shape, coherence, and rank large view spaces according to the shape, trend, density, coherence and outliers. While the Scagnostics method does not require classified data, consistency measures [186] can further improve the identification of informative scatter plots by considering the class consistency of labeled points. In [195] ranking measures based on the image space of scatter plots are used to identify potentially relevant structures. They used a rotating variance measure (RVM) to find correlations in the data and a class density measure to measure the overlap between different classes and rank views based on well-separated classes. A recent work of Matute et al. [131] has shown that a skeleton-based metric outperforms RVM and Scagnostics in perceptually-based similarity. Lehmann et al. [119] introduced a multi-step analysis of large scatter plot matrices. The approach is based on visual quality measures, matrix reordering, and visual abstraction, and supports navigation and analysis in large scatter plot data. In [31], two-dimensional color-coding was applied to compare sets of scatter plots for topological relationships. Other works supported the comparison of sets of scatter plots by automatic and interactive approaches. Albuquerque et al. [2] introduced an importance-aware sorting algorithm to find good projections in scatter plot matrices. A recently tackled problem is the identification of interesting subspaces in high-dimensional data, using scatter plots of projected subspaces. In [3], a sampling approach was shown that identifies interesting subspace projections for high-dimensional data sets. In [196], a visual approach for the identification of interesting subspaces was proposed. It relies on a clustering-based subspace search method to compute the interestingness score from density and class-separation measures. Moreover, extensive surveys about quality metrics for scatter plot and other visualization techniques are given in [15, 21].

Guidance for Scatter Plots

In general, guidance is an important research topic that may be integrated in various visual analytics processes to support users in achieving their goals. VizAssist [30] is a good example of how guidance may help to find good visualization techniques. It suggests visualization tools based on the underlying data and users' analytical objectives, e.g., cluster analysis, outlier detection, finding correlations. There are plenty of other guidance concepts in the literature, that support analytical tasks in several ways. In [38], Ceneda et al. have differentiated the different guidance concepts in visual analytics and characterized five different types of guidance, i.e., type, domain, input, output and degree. Moreover, to provide in-depth reasoning of guidance they establish a model by extending van Wijk's model of visualization [203] with guidance components. With regard to scatter plot visualization, there exists also several guidance approaches. For instance, Lehmann et al. [116] propose a method to generate pictograms as visual guidance for communicating specific properties of data distributions bivariate and multivariate projection visualizations such as scatter plots and radial visualization. In [69], a multiform visualization technique, called Domino, is presented that allows users to connect scatter plot with other visualization such as parallel sets, parallel coordinates or matrices. It visualizes the relationships between connecting items and show visuals cues to indicate compatible views. Also statistical measures like the Scagnostics features can be used for organizing multivariate views and for guiding interactive exploration [217]. In [39], 2D scatter plots including sensitivity coefficients are used to visualize projection changes in multi-dimensional data. A smoothness ranking is provided as guided navigation that suggests variables for projection changes with the smoothest re-projection.

7.3 Visual Guidance for Interactive User Interfaces

In this chapter, we present guidance approaches to support the user in sketching visual search queries and analyzing scatter plot patterns with interactive lenses.

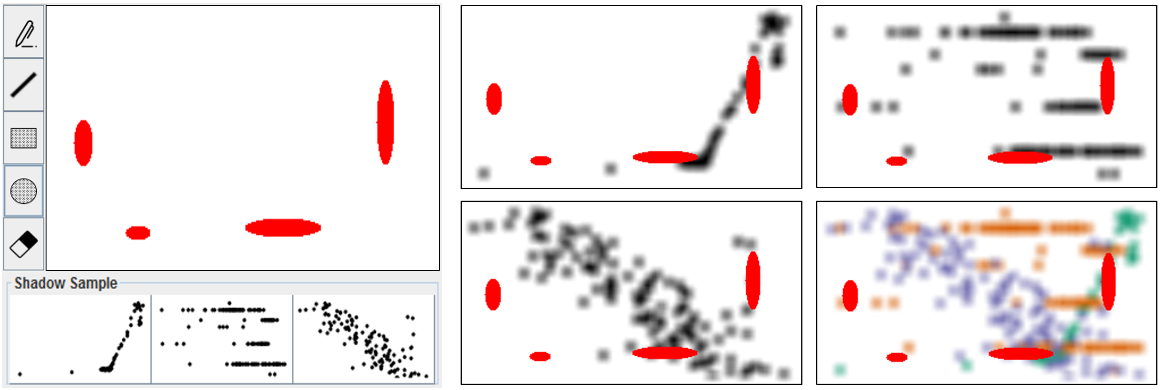
7.3.1 Guided Sketching for Scatter Plot Retrieval

In Chapter 3, we presented two visual search approaches including a sketch-based search. An important aspect of our visual search system is to find matching patterns in the data based on a rough sketch. However, in our retrieval experiments, we noticed that depending on the data set, it may be difficult to sketch precise queries for search, especially for inexperienced users.

For this reason, we extended the visual search interface with a shadow draw component to guide users in sketching a query. More specifically, we overlay contours of candidate scatter plot patterns in the background of the sketching panel so that the user can trace potentially matching patterns. This guidance feature supports real-time sketching feedback and provides the user with more accurate suggestions after each drawn stroke. This approach is inspired by [114], which supports users in freehand drawing of real-world objects. To suggest similar templates for a continuous sketch, our guidance approach performs a similarity search each time the user ends a given stroke during the query construction. To present a variety of sketching templates, we apply a k-means clustering on the n best matching results and present k patterns, one of each cluster as template, which are drawn in the background in a semi-transparent way. As shadow templates, we choose the cluster representative that is closest to the cluster's centroid. This ensures that we receive k rather different types of patterns, which are however still related to the current sketch, as they are computed from the current N best matches. Our guidance approach show possibly matching groups of scatter plot patterns, which can guide and inspire the subsequent sketching process. Consequently, the user may save time during the sketching process and may develop an understanding of the search space with regard to the given information –initial sketch. In order to provide less intrusive background images, we apply a blurring filter to obtain a more shadow like representations.

Figure 7.1 (a) shows an example of a user sketch with the resulting shadow patterns below. The suggested patterns can be used as sketching templates that can be overlayed in the background of the sketching interface Figure 7.1 (b). Users can select one of the suggested templates or display all templates with different color codings simultaneously. The aggregated view shows the similarity of the user sketch to all suggested templates and reveals areas with common point distributions. In this guidance demonstration, we chose as parameter setting $k = 3$ and $n = 200$.

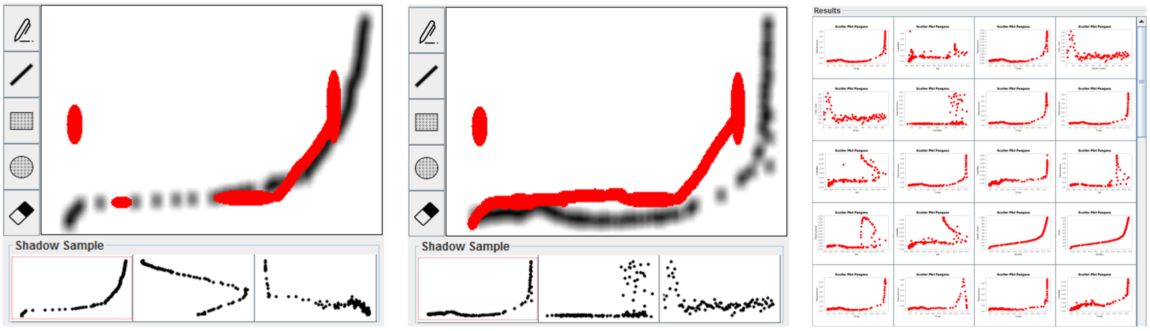
The basic idea of this guidance feature is to support users in creating visual search queries from start to finish. In Figure 7.1, we illustrate how a visual search may start by sketch distributed circles on the sketching interface. Thus, the guidance feature immediately shows potentially matching patterns in the data and gives a small overview of available patterns. Figure 7.2



(a) User sketch.

(b) Guided sketching with shadow templates.

Figure 7.1: Guided sketching for scatter plot search. User can start sketching a query and get instant feedback of available and similar patterns as sketching template (a). The guidance feature provides three different shadow templates that can be overlayed individually or all together in the background (b).



(a)

(b)

(c)

Figure 7.2: Demonstration of our guidance function by continuing the sketch from Figure 7.1. (a) shows the first extension of the sketch by connecting the two circles on the right. The shadow templates will be immediately updated after the sketch is modified and provide more accurate templates (b). (c) returns the search results of the query in (b).

demonstrates how this search process might be continued by using our guided sketching feature. In this case, the user chose the first suggested pattern as template (Figure 7.1 (b) upper left corner) and draw a connection between the two circles on the right hand side. Consequently, the guidance approach updates the suggested patterns and show more accurate patterns as templates, as shown in Figure 7.2 (a). By default, the guidance approach takes the best matching template as background image. Since it returns an interesting and good matching pattern, the user decided to complete the sketch by using the suggested template. The final user sketch is shown in Figure 7.2 (b) and the corresponding results in Figure 7.2 (c). Please note that the suggested patterns in Figure 7.2 (b) provides further opportunities and inspirations to modify the user sketch. This demonstration of our guidance feature shows how users may explore interesting patterns during the search process and finally retrieve a result set of similar patterns by using shadow templates.

To demonstrate the usability of our sketch guidance, we show further sketch suggestions in Figure 7.3. To this end, we use six frequently occurring patterns in scatter plot data and show the three pattern suggestions as shadows in the background. On can see that for each query suggestions are provided, which have similar but at the same time diverse characteristics to the drawn sketch. The purpose of this is to inspire users about available patterns with similarities to their current sketch, and thereby, support them at improving their query.

7.3.2 Quality Feedback for Interactive Lens Techniques

In Chapter 5, we presented a novel lens concept for visual-interactive regression analysis for scatter plots. Our approach offers users the possibility to freely define an area of interest on which the regression analysis may run. Mouse interactions for moving and resizing the lens selection facilitate an exploratory analysis procedure and creates the desired lens effect with real-time feedback of regression models. Thus, users will be able to exclude specific data points, such as clusters or points which belong to a particular class, which may have a negative influence on the statistical computation. However, finding a good selection area for the lens can be a challenging task, given the many possible positions and sizes of a regression lens.

During the development of our regression lens approach, we demonstrated its functionality to a smaller number of members of our research group, and invited them to informally test the system and specifically, comment on the interactivity of the lens operation. During these tests, we observed that often after a lens position was found, the researchers applied small local repositioning of the lens, to see if the chosen model would change noticeably or not. This observation inspired us to include an automatic guidance function that mimics this user behavior to improve the local regression model by small changes and offload the user from fine-grained selection tasks. Specifically, after a lens is dropped by the user, we apply tentative horizontal and vertical translations of 5% of plot width and height, respectively, and test if the regression quality is improving as measured via the in-sample error $e(f)$ (see Section 5.3) for any of these translations. Note that the selection of 5% is a parameter set heuristically and can be easily adapted to user requirements. We visually indicate the potential for improvement in model precision, thus guiding the user through the space of local regression models. To inform the user about improving directions, we provide a visual hint in terms of an arrow that points to the most significant direction of improvement (if given). This procedure optionally applies after each interactive adjustment of the lens, and thus creates an iterative feedback loop that helps users to find better fitting models.

Furthermore, we include an outlier detection to our guidance approach for indicating the points that may negatively influence the model computation. The used outlier detection is a distance-based approach that considers for each point selected by the given lens, the sum of

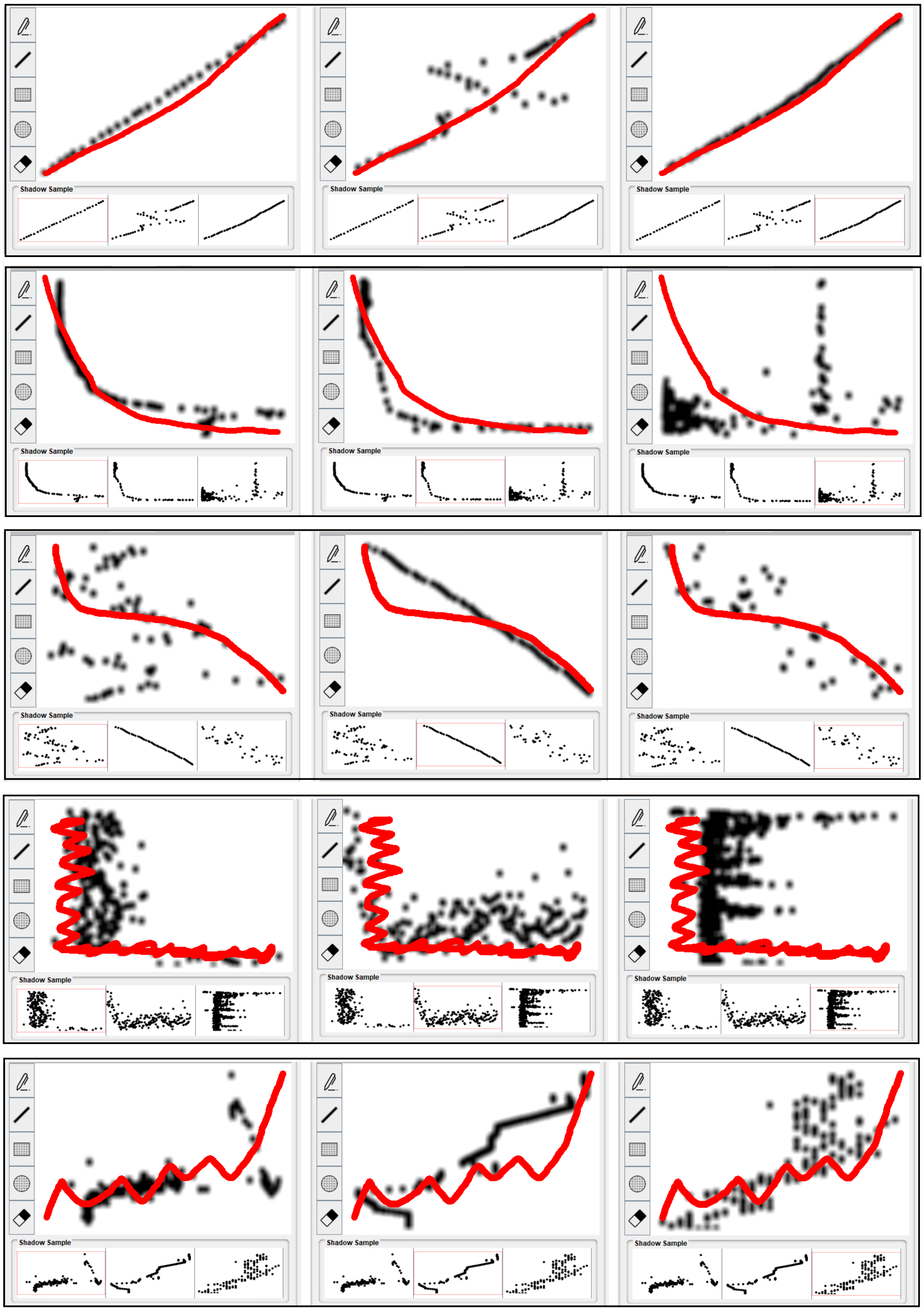
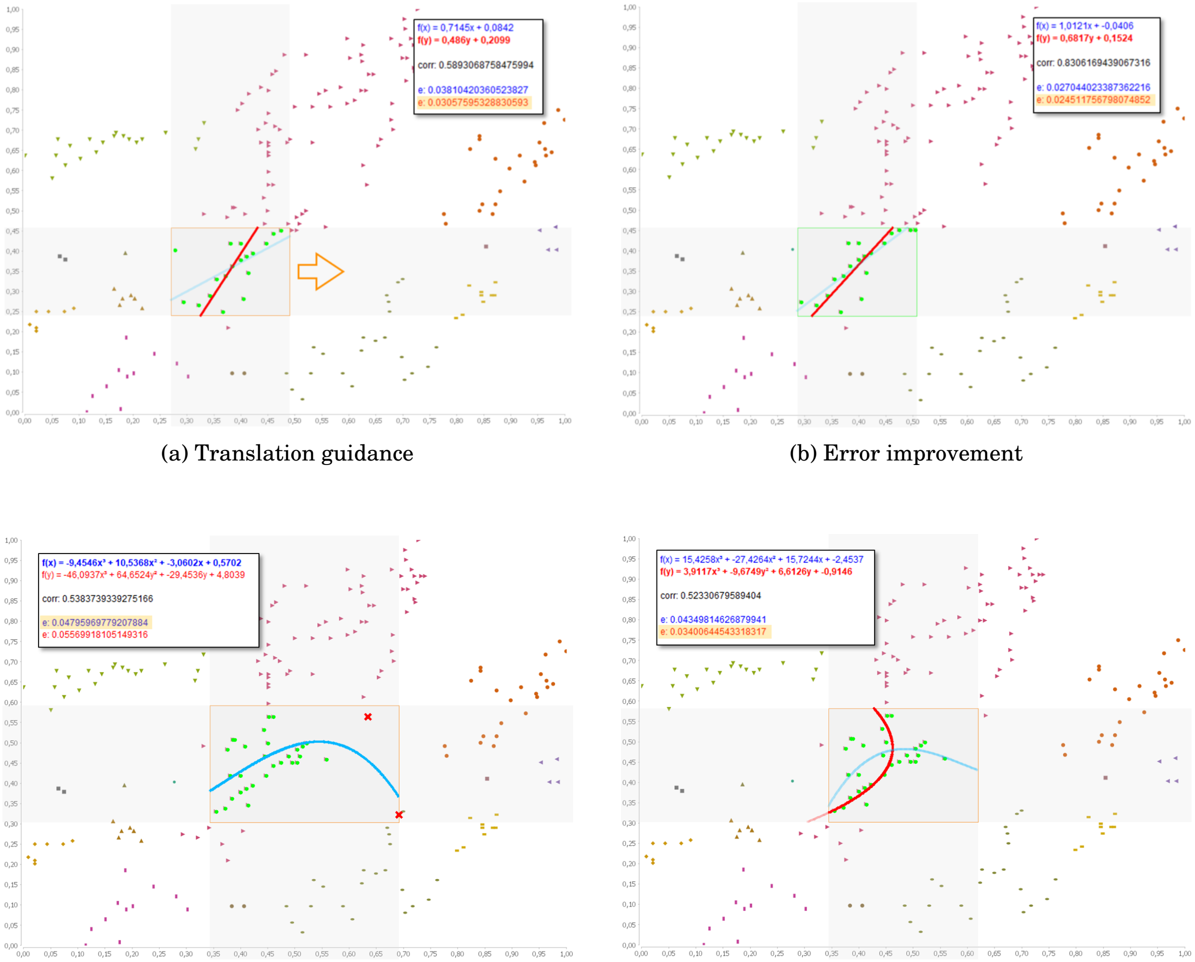


Figure 7.3: Further examples of our sketch guidance approach. For each sketch, three pattern suggestions are provided as shadow templates to inspire and support in sketching queries.



(a) Translation guidance

(b) Error improvement

(c) Outlier guidance

(d) Error improvement and model change

Figure 7.4: Two examples of selection guidance for our regression lens approach. In (a) the guidance approaches suggests to shift the lens selection to the right in order to improve the in-sample error $e(f)$. Through the shift the error score $e(f)$ has improved from 0.034 to 0.028, as shown in (b). In (c) the guidance approach has detected two outliers that negatively influence the error score. (d) shows that by excluding the outliers, the in-sample error will be decreased and the better fitting model changed from $f(x)$ to $f(y)$.

the distances from its k-nearest neighbors as outlier score [9]. Points are detected as outliers if their outlier score is larger than three times the average distance of all points to its k-nearest neighbors. For a fast and smooth computation, we automatically set k by using the heuristic $\sqrt{\frac{n}{2}}$ with n being the number of selected points.

A demonstration of this guidance concept is depicted in Figure 7.4. It shows two guidance approaches, translation suggestions (a)-(b) and outlier avoidance (c)-(d), that improve the model quality based on the in-sample error $e(f)$ (in-sample errors are highlighted in the information boxes). In the first example (a), our guidance approach has detected that the lens selection is not ideally positioned and therefore suggests to move the lens in the right direction. If one takes a closer look at the lens selection in (a), one can see that this selection involves an isolated point (indicated by a red circle) that impairs the linear regression. By moving the lens to the right, this isolated point will be excluded from the selection and in addition include two better fitting points according to linear regression. The results of this shift are shown in (b). Through a slight shift to the right, the in-sample error improved from $e(f) = 0.030$ to $e(f) = 0.024$. Furthermore, the correlation between the two models $f(x)$ $f(y)$ increased, which also indicates a better quality of the regression model. The second example (c), shows another bad selection including two outliers (indicated by red crosses). In this case, a shift to the left would exclude the outliers from the selection, but also include further points that will not improve the model quality. Thus, the guidance approach suggests to just exclude the outliers by downsizing the lens. By following the selection guidance, not only the in-sample error will improve but also the better fitting model –$f(x)$ vs. $f(y)$– will be changed, as shown in (d).

User Feedback

After initial tests and results, we again invited our participants from the first experiment to give our guidance features a trial. Since all participants were familiar with the system, we only gave a brief introduction to novel features and let them freely explore while thinking aloud. Afterward, a set of open-ended questions was asked regarding the lens guidance and the participant's impressions. Overall, all participants were pleased with the features and gave very positive feedback. The effectiveness of the lens translation was noted in particular - "it's a very quick way to find better selections", "it allows to quickly improve the model that might have taken longer by just manually adjusting the lens". The outlier detection was seen as useful - "I think the outlier detection is helpful, since I have included outlying points several times", "it helps to find unintended selections, which was made by the rectangular lens selection". All participants stated that the guidance features were not distracting and they saw the usability for improving local models, even if it was initially unclear what they are looking for.

7.4 Scatter Plot Recommendation based on Motifs Interestingness Scores

An alternative approach to find interesting views in the data is to use visual recommender systems that try to predict the preference of a user to a given view. In this section, we introduce a scatter plot recommender based on interest scores derived from local motifs.

The goal of this guidance approach is to guide the analyst through the exploration process, when facing a data set with a large number of individual scatter plots. Our main idea is to use the *motif dictionary* from Section 3.3 to find scatter plots containing interesting motifs. The dictionary contains prototype scatter plot segments (called motifs) that represent the different local scatter plot shapes occurring in a given data set. Based on an adapted $tf \times idf$-approach, we compute local and global interestingness scores derived from the frequency properties of motifs in the dictionary. Details about dictionary generation are given in Section 3.3.

We compute a measure of interestingness for each scatter plot. To this end, we rely on the notion of $tf \times idf$-analysis from information retrieval. Briefly, we consider each entry in the dictionary (motif) as a visual word. Intuitively, a scatter plot which contains one or several instances of a motif (high term frequency), which does not occur in many other scatter plots (low document frequency), is considered important. We use this intuition to define a measure for ranking the interestingness of scatter plots. Also, given that we have segmentation and dictionary, we can apply color-coding to visualize the distribution of motifs across many scatter plots for interactive exploration (see also Section 7.4.1).

Next, we will provide technical details of our implementation for detecting interesting scatter plot motifs and presents our aggregation scheme from local to global interestingness scores.

Dictionary-Based Interestingness Score

The dictionary contains information about the distribution and frequency of segments, and is used to determine the local interesting score. Therefore, the characteristics of the segments need to be described by a suitable feature vector. While many different visual features are possible candidates, we decided to use edge orientation and density features, as these have been shown to work robustly for global comparison of scatter plots [164].

We apply a k-means clustering on the feature vectors of all local segments to form the motif dictionary, as illustrated in the last step of Figure 3.1. An essential step here is the parameter setting k for the number of dictionary entries, since it influences the quality of the dictionary and consequently the local interestingness score. To determine an appropriate setting for k, we developed a visual exploration front end for experimental tests (see Section 3.5.2), which visualizes the motif dictionary for different settings of k.

The set of clusters is the basis for computing the local interestingness score and expresses the uniqueness of a motif, and how discriminant the motif is regarding the entire scatter plot space. This means that large motif clusters including many visually similar segments are weighted lower than small motif clusters with overall *rare* segments. Accordingly, scatter plots containing several motifs and especially rare ones are ranked higher, and suggested for investigation.

$$MU_{score}(q) = \frac{1}{|\{p \in Dict[q]\}|} \tag{7.1}$$

Equation 7.1 shows the proposed *Motif Uniqueness (MU)* score and how we measure the local interestingness for a given segment q. We divide one by the total number of segments p in the data set that belongs to the same motif q (i.e., the cluster size of the motif).

Global Interest Measure

The overall goal of our approach is to find interesting scatter plots for the exploration, containing discriminative local motifs. The global interest measure should reflect the interestingness of a given scatter plot based on the frequency of its local motifs in the entire scatter plot space. It is comparable to the text mining approach $tf \times idf$ [159], which uses the importance of a word to rank a document in a corpus. Instead of using the term frequency (*tf*), that computes the frequency of a term in a document, we use the *motif uniqueness* score MU. .It reflects how interesting and discriminant a motif is with respect to the corpus/scatter plot space. The basic idea of this local score is to weight frequent motifs (e.g., single dots or stripes) lower, and vice versa to weight discriminant motifs (e.g., complex patterns) higher.

The global interestingness measure is derived from these local factors in combination with an overall interestingness score. It corresponds to the inverse document frequency (*idf*) in text mining. The inverse document frequency is a measure to compute the overall importance of a term across all documents and follows the same idea as our second weighting factor that we call *inverse scatter plot frequency (ISPF)*. The difference to our approach is that we take the dictionary information and visual features into account and measure whether a motif is common or rare across all scatter plots. As shown in Equation 7.2, this score is obtained by dividing the total number of scatter plots N by the number of scatter plots sp containing one of the motifs in the dictionary cluster, and then taking the logarithm of that quotient. The substantial idea of this second weighting factor is to identify if a dictionary entry is based on many scatter plots containing such a motif, or e.g., just one scatter plot that contains many identical motifs.

$$ISPF_{score}(q) = \log \frac{N}{|\{sp \in Dict[q]\}|} \tag{7.2}$$

Algorithm 1: Computation of a global interest measure

```
Input: $motifDict, S$
Result: List of global interest measures
foreach $scatterplot$ in $S(s_1, ...s_n)$ do
    $localMotifs$ = get motifs of $scatterplot$
    foreach $m$ in $localMotifs$ do
        $dictIndex$ = get dict index of $m$
        $localScore = MU(dictIndex) \cdot ISPF(dictIndex)$ $globalScore$ += $localScore$
    $globalScore = globalScore/$size of $localMotifs$
    add $globalScore$ to $resultList$
return $resultList$
```

All local motif scores of a scatter plot are accumulated to produce the global interestingness score. Thus, scatter plots containing different and infrequent motifs achieve a higher score and are thereby considered as more interesting. Our proposed aggregation scheme for this interest measure is specified in Algorithm 1. For comparison reasons, we divide the aggregated global

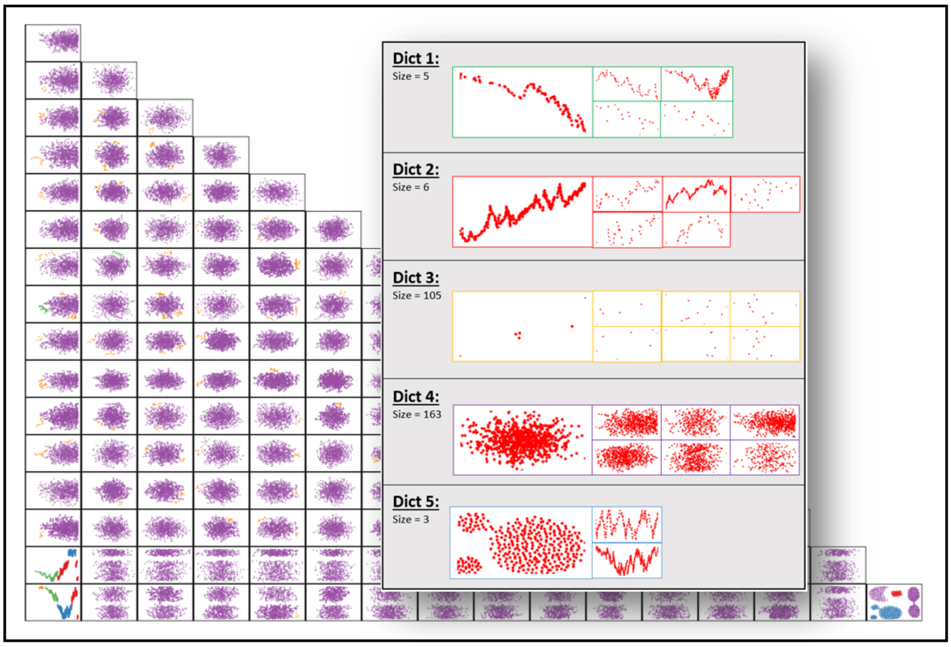


Figure 7.5: Scatter plot matrix overview of the synthetic data set and the resulting dictionary with five entries. By means of the displayed dictionary view and the corresponding color coding, analysts can easily determine a good dictionary size for a given data set.

scatter plot interest score by the number of local motifs. Alternatively, analysts can use a range factor to prioritize the number of desired motifs and can penalize scatter plots containing more or fewer motifs. By means of this interest measure approach, we are able to automatically extract interesting views for the exploration of large scatter plot spaces.

7.4.1 Use Cases

Next, we now demonstrate our recommender system including the scatter plot interest measure and show ranking results. First, we use a synthetic data set as a proof-of-concept to showcase our proposed approach. We then make use of the interest measure to investigate recommended views on a real-world data set.

Synthetic Data: Interestingness Measure

We created a synthetic data set by merging 15-dimensional Gaussian clusters with the two-dimensional *Aggregation* data set presented in [66]. Since the aggregation data set consists of a small sample size (788 records), we randomly created Gaussian clusters with the same size and merged the data, as illustrated in the background of Figure 7.5. The original scatter plot of the aggregation data set is located at the bottom right corner of the matrix. The experiment was designed to depict that motifs of the Gaussian dimensions (purple motif), which appear more often will also result in a low local and overall interestingness score. In contrast, scatter plots

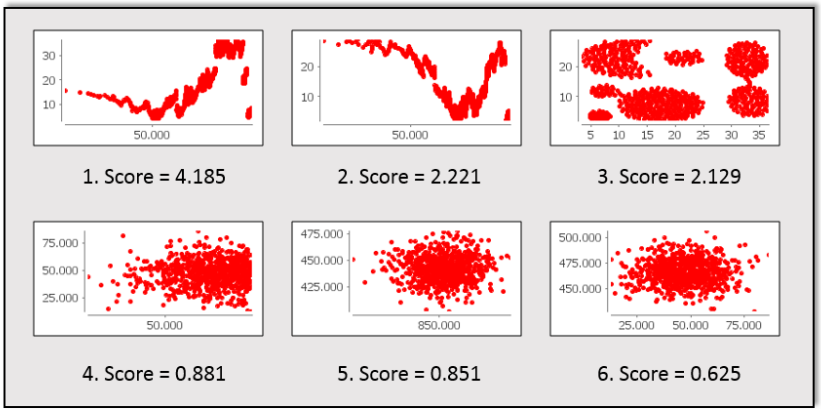


Figure 7.6: The six most interesting scatter plots of the synthetic data set for exploring local motifs. The global scores are obtained by aggregating all local motif scores ($MU \times ISPF$) of a given scatter plot.

that were merged with one of the aggregation data dimensions (last two rows) contain more complex and outstanding motifs, and will thus be rated more interesting.

The first step of our approach is to determine the interesting scatter plot segments by running our adapted MST approach (see Section 3.3). After the segmentation step, we have extracted 282 local segments from 136 scatter plots. When looking at the scatter plot matrix (Figure 7.5 - background), one can see that the data set contains only a few kinds of different motifs. In this case, we recommend choosing a small k (e.g., between three and five) to keep the quality of the dictionary high and clearly separate the different motif shapes. Choosing a too large dictionary size would lead to splitting up the homogeneous motifs of the Gaussian clusters into several dictionary entries and thus will distort the local interestingness score. On the contrary, a too small dictionary size will merge dissimilar motifs and also negatively influence the ranking.

In the experiment, depicted in Figure 7.5, we found a good dictionary setting by using a combined image descriptor, which takes the edge orientation and density of a motif into account, and chose a dictionary size of five. Thus, we received a dictionary with five well-separated clusters, containing a negative trend motif (green), positive trend motif (red), sparse point clouds (orange), dense point clouds (purple) and a motif cluster with wide-spread distributions (blue). The largest motif cluster is represented by the purple color with 163 similar segments, followed by the orange cluster with 105 segments. As one can see, all patterns are highly similar in the scatter plot space except those from the original aggregation data set and the two scatter plots in combination with the first dimension. Consequently, our interest measure ranks scatter plots with the purple and orange motifs less interesting than the other motif groups. A result overview of the six top-ranked scatter plots is shown in Figure 7.6. As expected, the three scatter plots containing deviant motifs achieve the highest global scores. The reason why the first three scatter plots received significantly higher scores is due to their higher ranked local motifs. The original aggregation scatter plot has been ranked third after the two line shaped scatter plots, because of the poor local score of the two purple motifs. Very unexpected was the incident that all other scatter plots derived from the two aggregation data dimensions are not in the top six results. This can be explained by the fact that scatter plots on rank four to six also contain higher ranked motifs from the orange and green dictionary entry.

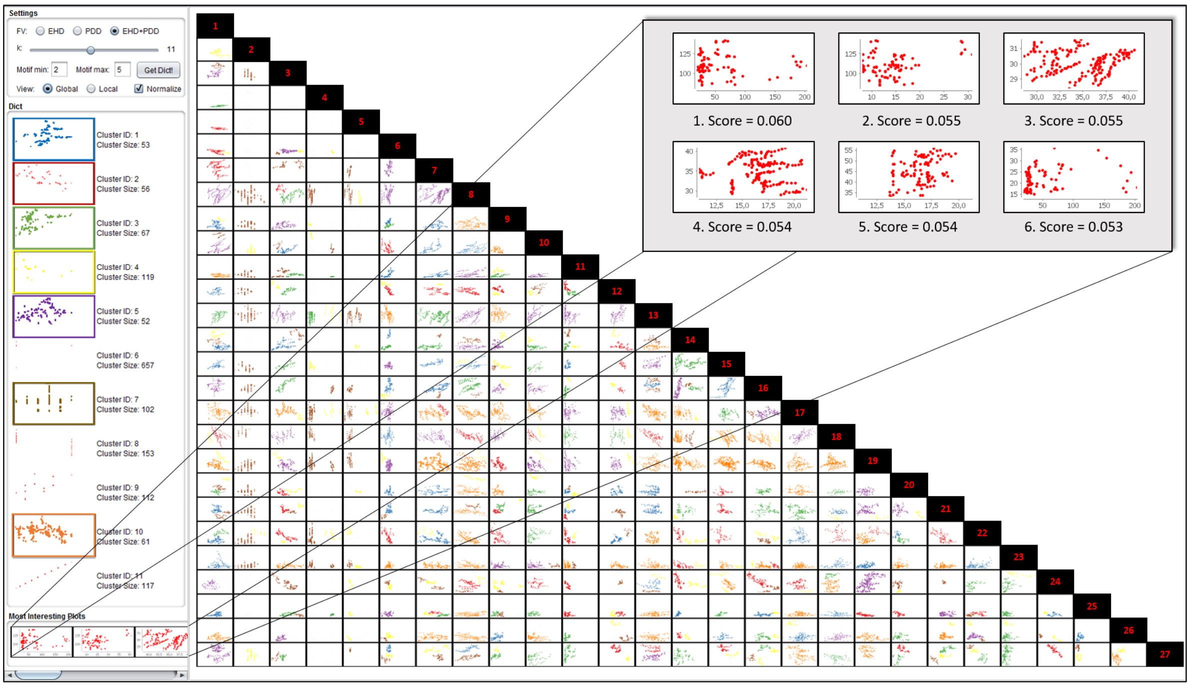
Real-World Data: Interestingness Analysis

The second evaluation data set is retrieved from the *eurostat*[1] data repository. The data repository provides in total 5500 data sets each containing information about a European related topic, such as economy, population and industry. We extracted a data subset containing 27 statistical attributes (e.g., population density, duration of working life, electricity consumption by households, etc.) from 28 EU countries that show temporal changes over the last decades. From these 27 dimensions, we created a scatter plot matrix (bottom half) resulting in 351 unique scatter plots ($\frac{27\times26}{2}$) in which each data instance (point) represents one country at a specific year. The corresponding scatter plot matrix is illustrated in Figure 7.7.

As in the previous example, we start the interestingness search process with segmenting the scatter plot space into local segments and select a good setting for the dictionary. Our segmentation approach returned 1549 local segments of the 351 scatter plots. We are using the combined image descriptor for characterizing the motifs in this experiment. As dictionary size, we found that appropriate results were achieved by using a size between 10 and 15. Then we iteratively highlight the most considerable motifs in the scatter plot matrix to identify the similarity of a dictionary entry and thus prove the quality of the settings. Finally, we decided to choose a dictionary size of 11 for further analysis. As Figure 7.7 depicts, one can clearly recognize the dependencies between similar motifs and the dimensions in the scatter plot matrix. For instance, if we consider the brown motif class (dictionary cluster ID 7), we are able to identify all the dictionary items in column two, four and five. The same applies to the orange motif class with sparse negative trend direction (dictionary cluster ID 10), which are mostly located in row 16 and 18. Finding such properties in the scatter plot matrix may lead to first insights into the local motif analysis.

The top ranked scatter plots of our chosen setting are outlined at the bottom left corner of our visual exploration tool. An enlarged excerpt form the best six rankings is also shown in Figure 7.7. On closer inspection, we can see that all suggested scatter plots contain significant motifs, which may be interesting to analyze. As an analysis example, we want to focus on the scatter plot ranked in the third place. The scatter plot shows separated motifs with several positive trend directions shifted on both axes. These motifs describe the relation between the mean duration of working life against the mean age of women at childbirth of the population in all EU countries. It becomes clear that the total work duration of women decreases when they become a mother

[1] Statistical Office of the European Union (`http://ec.europa.eu/eurostat`). Accessed 12/2019.



Settings

FV: ○ EHD ○ PDD ● EHD+PDD

k: 11

Motif min: 2 Motif max: 5 Get Dict!

View: ● Global ○ Local ☑ Normalize

Dict

Cluster ID: 1
Cluster Size: 53

Cluster ID: 2
Cluster Size: 56

Cluster ID: 3
Cluster Size: 67

Cluster ID: 4
Cluster Size: 119

Cluster ID: 5
Cluster Size: 52

Cluster ID: 6
Cluster Size: 657

Cluster ID: 7
Cluster Size: 102

Cluster ID: 8
Cluster Size: 153

Cluster ID: 9
Cluster Size: 112

Cluster ID: 10
Cluster Size: 61

Cluster ID: 11
Cluster Size: 117

Most Interesting Plots

1. Score = 0.060

2. Score = 0.055

3. Score = 0.055

4. Score = 0.054

5. Score = 0.054

6. Score = 0.053

Figure 7.7: Our visual exploration tool for global and local scatter plot analysis. By means of this tool, analysts can derive different motif-based dictionaries by adjusting the parameter settings and thus achieve various interesting scatter plots suggestions for exploration. The parameter settings, dictionary view and the resulting global interest ranking are located on the left hand side. Local motifs of a given setting can be highlighted in the scatter plot matrix to assess the dictionary quality.

earlier and for some reason, can no longer work. Additionally, it would be interesting to analyze these different groupings in relation to other non-numerical attributes, such as geolocation and see which countries share similar characteristics and how they change over time.

7.5 Discussion of Limitations and Extensions

The basic idea of our guidance approaches is to guide the user in exploring scatter plots based on the notion of local motifs. The concept of local pattern analysis is novel in that it extends beyond most feature-based scatter plot analysis methods, which consider global features. Our solutions are a first step to support the analysis for scatter plot patterns by using local features.

Guided Sketching

Our sketch guidance approach is implemented based on global scatter plot patterns. By using a pre-segmentation as introduced for the motif dictionary, it can also be adapted to support sketching local patterns. Regarding the proposed segmentation, many alternatives could be considered. First, data space segmentation approaches are possible, but also, image-based segmentation approaches could be considered. While our implemented approach is based on partitioning distributions in the data space, other data space segmentation approaches are possible. For example, a regression tree can be learned for finding a (non-)linear partitioning of the scatter plot space. Alternatively, a wide range of options from the image analysis community is available and not yet explored on scatter plots. We are planning to experiment with convex hull calculations on the rendered scatter plot images to detect local motifs. One advantage of this approach is that data space axis ranges are normalized by definition and no adjustments are needed for similarity comparisons. In our case, we normalize the images of each scatter plot segment based on the unit square. While this normalization supports easy extraction and comparison of features, it ignores different scales and aspect ratios of extracted patterns. All of this could be taken into account by extending the feature vector based on the application need.

As discussed in the previous Section 3.6, are performance evaluations in sketch-based search approaches very hard to achieve. The main objective of this guidance feature is not to provide the most accurate suggestions for unfinished sketches but rather to inspire users about available patterns in the data and support them in sketching potentially interesting patterns. Nevertheless, we experienced that the suggested shadow templates were quite appropriate for rough sketches with recognizable structures. Initial sketches without clear shapes often will return diverse results, which in turn provides varying shadow templates. As shown in the demonstration (Section 7.3.1), suggestions will be quickly adapted after each drawn stroke and become more precise the more a user draws.

Moreover, we are planning to integrate further motif descriptors into our system. While currently, an edge-descriptor and a density-based descriptor proved useful for our case studies, we expect that a greater variety of scatter plot motifs can be described with other feature descriptors. One option is to apply Hough analysis to detect presence of basic shapes in scatter plots such as lines or areas and form a descriptor from these. Other than that, regressional features, such as described in [163], could be integrated. One advantage of the latter descriptor is that it can be interpreted in terms of a regression model, whereas the density and edge orientation features used here, are low-level and cannot be easily interpreted by the user. These descriptors will bring an advanced semantic level in the motif detection, which needs to be understood and researched more extensively.

Regression Lens Feedback

A well-known problem in data analysis is the issue of dealing with outliers. With the guidance feature for our interactive lens, we address this issue on a local level and support the interactive selections of areas of interest in scatter plot views. The guidance approach including an outlier detection to indicate selected points that may negatively influence the model computation. It should be taken into account that this guidance feature is performed on a local level and may indicate outliers, which may not be an outlier on a global level. Nevertheless, it is reasonable to indicate such data points and inform the user about computational improvements of local models. It should also be pointed out that our guidance features for the lens, only inform users about possible model improvements and do not show up if user selections are chosen well. More advanced or domain-specific outlier detection methods can also be thought of as an extension. Furthermore, our guidance approach uses a translation model with a predefined shift size to suggest local repositioning of the lens. As one participant correctly observed, unintended points may be selected by the rectangular lens selection, which leads to poorer results. More sophisticated guidance may involve more exhaustive search over translation or transformations of the lens selection. For instance, lens selections may be rotated or a convex hull could also be used here to improve the lens selection.

Motif-based Recommendations

The key aspect of our scatter plot recommender is the definition of interestingness based on motif distributions. Our proposed score is basically representing the notion of outlyingness (or sparsity). However, many other notions of interestingness could be defined, based on the motif distribution. For instance, a distribution where specific combinations of motifs in a scatter plot occurring frequently could be valuable. We exploit the motif analysis and ranking in an interactive scatter plot matrix representation, which allows users to compare and overview the different motifs by color-coding. The visual exploration tool could be enhanced in several ways. An overlay of cluster members with semi-transparent drawings may be effective to this end. So far, we draw each local pattern in its original shape, showing the cluster (motif) membership by color coding. An alternative that could scale better for large scatter plot spaces, might be a shape abstraction of each motif and replace the occurrence of a motif in a scatter plot by its simplified version. The recently proposed so-called Visual Guidance Pictograms [116] could be a starting point to this end.

Beyond these ideas for technical extensions, we realize that our approach should also be evaluated together with domain data analysts. Local scatter plot analysis is potentially useful in all domains where several patterns need to be distinguished and related in respective scatter plots. Examples may include the business domain, where a customer segmentation and relationship analysis is needed. Another domain is the analysis of earth observation data [163, 164], where e.g., different natural phenomena could mix in a single data set, giving rise to locally relevant patterns. Future work should elaborate on such application cases and assess the effectiveness of local scatter plot analysis.

7.6 Conclusion

We extended our previously presented interaction techniques with guidance features to support users in operating the tools. More specifically, we introduced a sketch guidance approach that helps users in specifying visual queries by showing template patterns in the background. Thus, users can explore available patterns during the sketching process and take usable templates to improve their sketches. For our regression lens tool, we included a guidance concept that supports the interactive process of finding well distributed selections based on the in-sample error of slight translations. Furthermore, we introduced a novel workflow in which we analyze the interestingness of automatically extracted local motifs to guide the exploration in scatter plot data. To assess the overall interestingness, we adapted the $tf \times idf$ scheme from information retrieval to the domain of scatter plot motifs. We derive the interestingness of local scatter plot motifs based on its occurrence among and within the scatter plot space. We developed interactive visual exploration tools with brushing and linking that supports analysts to find appropriate motif dictionaries and suggests interesting scatter plots for exploration. Finally, we applied the workflow on a synthetic and real-world data set to demonstrate how it can efficiently lead to interesting discoveries of local motifs. Our approach is only a first step into the direction of local analysis in large scatter plot spaces, and we have discussed a range of extensions to be done in future work.

CHAPTER 8

Guided Exploration for Scatter Plots based on Pattern Recommendation via Eye Tracking

Contents

The general idea of guidance systems is to use computer-assisted processes to actively resolve knowledge gaps during an interactive VA session [38]. In Chapter 7, I presented different guidance approaches that support the exploration and retrieval of scatter plots by using quantitative measures. These measurements are computed based on the users' input, e.g., drawn sketch, lens selection and dictionary configuration, to provide suggestions. However, one remaining issue in the exploration of large scatter plot spaces, is that typically not all views are potentially relevant to a given user or analysis task. For example, if one considers a scatter plot matrix (SPLOM), similar patterns often occur across the dimensions –rows and columns– and may interfere with the investigation for unexplored views. To assist and guide exploration for unexplored views, currently a user has to manually annotate explored views as relevant or irrelevant, which can be tedious for large view spaces.

In this chapter, I will go one step further and consider novel sensor technologies like eye tracking for the guidance process. I introduce a novel concept and prototype implementation for an interactive recommender system supporting the exploration of large SPLOMs based on indirectly obtained user feedback from user eye tracking. The system tracks the patterns that are currently under exploration based on gaze times, recommending areas of the SPLOM containing potentially new, unseen patterns for successive exploration. I use an image-based dissimilarity measure to recommend patterns that are visually dissimilar to previously seen ones, to guide the exploration in large SPLOMs. The dynamic exploration process is visualized by an analysis provenance heatmap, which captures the duration of explored and recommended SPLOM areas. To demonstrate the usability, I showcase a user experiment, showing the indirectly controlled recommender system achieves higher pattern recall as compared to fully interactive navigation using mouse operations.

This chapter is based on:

> [180] **Visual Exploration of Large Scatter Plot Matrices by Pattern Recommendation Based on Eye Tracking,** L. Shao, N. Silva, E. Eggeling and T. Schreck. *ACM Workshop on Exploratory Search and Interactive Data Analytics (ESIDA '17), 2017.*

8.1 Introduction

A current problem of data analysts is the exploration of large amounts of data in a short period of time. The data may contain a high number of dimensions, which increases the number of potentially interesting views. One way to explore high-dimensional data more efficiently is to use Scatter Plot Matrices (SPLOMs), which visualize all pairwise combinations of dimension views in tabular form. However, the exploration in large SPLOMs is a challenging task, as the view space can be very large and interesting views may be overlooked in exploration. The resulting question is how to explore SPLOMs more efficiently to investigate a large variety of interesting views in less time? Quality metrics [15, 21] can be used, which help with the visual exploration of patterns in large spaces of alternative visualizations, like projection views. A drawback of these approaches is that these methods are objective numerical measures of qualities based on specific tasks, e.g., clustering or outlier detection and thus, may fail to reflect the given user interest. To cater for the specific needs of the user, recommender systems including classification may be used to improve search and filter tasks. However, most classification methods require explicit user interactions to work.

Recently, research in eye movement analysis has made advances, in the technology to track eye movement, as well as in the analysis and visualization of eye movement data. Eye tracking so far has been mainly used to evaluate user gazing in conjunction with user interfaces, or to do user interaction. It is a promising research direction to include eye tracking for Visual Analytics approaches as an indirect means to monitor the user and to adapt the interactive analysis process. We believe that the integration of eye tracking into Visual Analytics approaches is reasonable and opens up new possibilities.

We present a novel concept to recommend interesting scatter plot views based on user attention measured by an eye tracker. This idea is inspired by information retrieval approaches and recommender systems, which help to find previously unseen information by explicit user relevance feedback. Therefore, we use an area of interest (AOI)-based metric that identifies scatter plots on the stimulus as AOI, and capture the transitions between AOIs and the gaze times for each AOI. A classifier learns the visual characteristics from previously seen plots and uses a dissimilarity measure to recommend the most visually dissimilar scatter plots for further exploration. Moreover, we conduct a user experiment to demonstrate the usability of our system and show efficiency increase during explorations.

8.2 Related Work

In many areas, eye tracking devices are used for analyzing user behavior, e.g., in market research, human-computer interaction and visualization research. Our approach combines user attention analysis with interactive learning systems to support exploratory analysis tasks.

8.2.1 Eye Tracking in Visual Analytics

In Human-Computer Interaction, eye movement tracking is commonly used to study usability issues [94]. An introduction to the basics of eye movement is given by Pool and Ball [153]. They report about key aspects of practical guidance in usability-evaluation studies, and give several statistical metrics that can be derived from eye tracking data and their possible interpretations. Attention heatmaps [27] are often used to show the distribution of the users' attention over the display space when performing tasks. These heatmaps are useful to visualize fixation counts and gaze times on a point-based level. Our work follows an AOI-based metric, where each scatter plot is an AOI. The usage of eye tracking in typical analysis tasks like detecting clusters or correlations is addressed by Etemadpour et al. [58]. Holmqvist et al. [85] have investigated different methods and measures for analyzing user attention patterns. An overview of visualization techniques and methods for analyzing eye movement data is given in [4, 26]. The use of eye movement as an input mechanism to steer a system is considered in [93, 153]. The choice of using eye movement tracking instead of point-and-click interaction allows a non-invasive and continuous tracking of the users. By using eye movement analysis, a fast and continuous tracking of the user interests in real-time is possible. This allows, for example, the detection of moments of confusion, indecision and high interest regions [67]. A previous study [75] discusses important links between cognitive processes and eye movements. Here, the potential of using eye movements as a performance measure is debated, and several examples of possible inferences that can be made using eye tracking are given. Finally, there are guidelines for adjusting the user interface design to improve the accuracy of eye tracking.

In our work, we use eye movement tracking to identify the user's interests in exploratory analysis tasks by extracting the eye gaze path on the user interface.

8.2.2 Active Learning and Recommending

In information retrieval, learning methods are often used to classify text documents automatically and have shown to be effective [167]. These methods derive models from a set of pre-classified documents and make use of the characteristics of the categories to label previously unseen data. For instance, in [29], classifiers are used for filtering and monitoring text data streams from Twitter. Heimerl et al. [78] compared three different approaches to interactively train an SVM text classifier – basic learning method, visual method and user-driven method. Furthermore, Visual Analytics tools are available, which help to train improved classifiers. Höferlin et al. [82] extended their active learning system with a Visual Analytics process to define filters by ad-hoc training classifiers in the domain of Video Visual Analytics. In [32], visual summaries of misclassified documents are presented to improve the feature ideation and creation of the classifier. Besides text documents, learning systems can also be applied to other research areas such as content-based recommender systems. Behrisch et al. [16] presented a recommender system for scatter plot exploration, which uses a classifier based on Scagnostics [216] features. Moreover, surveys of existing recommender systems and possible extensions concerning recommendation capabilities are given in [1, 128]. However, user interactions like labeling, selecting or rating a test data set are required to train the classifier. Our recommendation system makes use of the advantage of eye movement analysis and transfers the information about user attention directly to a k-nearest neighbor recommender.

8.3 Overview of our Approach

In this section, we present our recommendation concept and indicate how we indirectly integrate guided visual dissimilarity search to support the exploration in large SPLOMs.

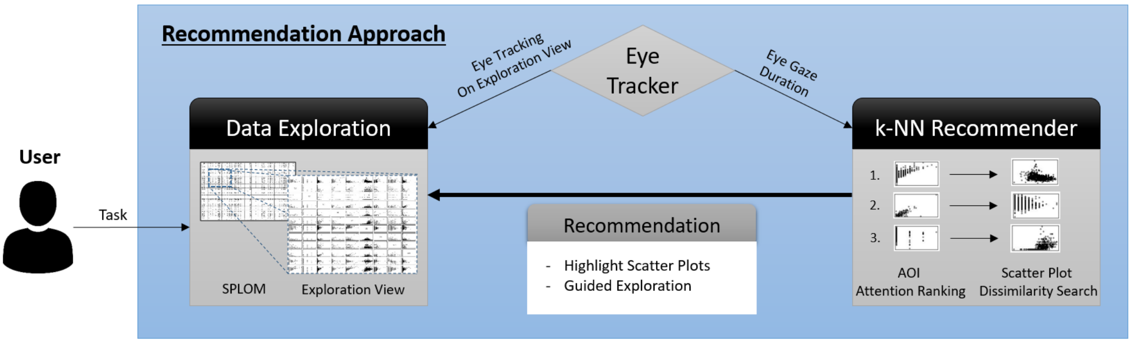


Figure 8.1: We use an eye tracker as independent intermediary to connect the k-NN recommender with the data exploration interface. Thus, users can focus on the data exploration task while the recommender automatically receives the information from the eye tracker for supporting the exploration.

8.3.1 Recommendation Concept

Our approach aims to support exploratory analysis tasks in SPLOMs by suggesting interesting and previously unseen scatter plots. We define interestingness in terms of unseen and visually dissimilar to previously explored patterns. The basic idea is to help the user explore large view spaces as efficiently as possible, avoiding to get lost in details and miss relevant patterns. Thus, the user may focus on underexplored areas and explore the SPLOM in a different way. To accomplish this, we use an eye tracker that automatically records user attentions during the exploration and supply a k-nearest neighbor (k-NN) recommender with the characteristics of explored scatter plots. In our concept, the eye tracker acts as a bridge between data exploration and k-NN recommender, as depicted in Figure 8.1.

An advantage over previous recommender systems is that we use an eye tracker as an intermediary to supply the k-NN recommender with the users' interests. Hence, users can fully concentrate on the exploration task and do not have to provide any kind of relevance feedback to the system. The eye tracker captures the actual users' interest in terms of total gaze duration and number of fixations per scatter plot. Consequently, we determine which scatter plots have been explored for the longest time and were fixated the most. These serve as a basis to determine the most dissimilar plots, for recommending. For instance, a scatter plot with a high number of fixations could indicate that it was used as a reference for comparison. We also assume that scatter plots, which attracted the user's attention for a longer time, were of more interest to the user.

Based on a history of previously explored views, we perform a dissimilarity search to the top explored scatter plots –most fixations or longest gaze times– and recommend the most dissimilar ones for exploration. To provide a larger variety of different scatter plots as recommendations, we perform multiple queries from the scatter plots, which called the user's attention the most. Since we are recommending visually dissimilar and unseen scatter plots, our recommendation approach still works for exploratory analysis tasks even if scatter plots with high durations were considered as uninteresting. Our first priority is to support the exploration óf undiscovered patterns.

Finally, the k-NN recommender returns the recommendations in terms of visual feedback or by guiding the exploration to the suggested areas. It is a unidirectional connection between k-NN recommender and data exploration interface, in which the recommender supplies computed information to the exploration interface in real-time. Thus, the exploration interface has to wait for input information and transmit the recommendations to the user. To inspect the recommended scatter plots, users can manually change the SPLOM viewport to certain areas by zooming and panning interactions. Alternatively, users can let the system fully control the viewport by using the *Guided Exploration* function. By using a single keystroke, the viewport is changed to the SPLOM area with the most recommendations. The major advantage of this function is that users do not have to decide about when and where to navigate within the matrix. The system automatically detects the areas in the SPLOM with the highest computed interest to the user. Optionally, the system may be set to decide to change the current viewport, once the main patterns visible in a current viewport have been attended by the user (see section below). In this manner, we create a recommender system including navigation guidance, which constantly learns from the user's eye movement and recommends undiscovered scatter plots.

8.3.2 Visual Cluster Analysis

In SPLOMs, zooming and panning functions are needed to investigate individual scatter plots in larger view spaces. However, one problem of these interaction functions is that the user may get lost in the detail of large view spaces. Another drawback is that these zoomed-in sections, depending on the data characteristics or matrix ordering employed, may contain visually similar scatter plot patterns. An example is shown in Figure 8.2 (a) - row 4 and 5. Such areas containing similar plots may expend user attention, which is especially important during early exploration stages. At this point, questions arise concerning when to leave the current exploration viewport and follow the recommendations provided by the system.

To this end, we integrate visual cluster analysis into our system to estimate the number of different scatter plot classes given in a SPLOM, and provide recommendations and navigation guidance based on the exploration degree of the current viewport. In other words, we notify the user to explore further sections as soon as the current viewport is fully explored, i.e., at least one scatter plot of each class is explored. To do this, we cluster scatter plots by their visual similarity and determine the number of various patterns in the user's viewport.

Since image-based descriptors performed quite well for detecting visual similarities in scatter plots [164, 178], we rely on image features based on the density of points and the distribution of edge orientations [220] as implemented in the edge histogram descriptor. Due to performance reasons, we extract the image feature vectors of all scatter plots and apply the DBSCAN clustering algorithm [57] in advance to achieve all classes of similar scatter plot patterns. Finally, we compute the number of distinct scatter plot classes of the current SPLOM viewport and send notifications when all different classes were explored for at least a minimum exploration time of 200 ms. The limit of 200 ms is an important threshold in human visual attention as it is the time needed to initiate an eye movement and therefore to begin a serial attentive mechanism. Usually, the responses obtained in less than 200 ms are considered as preattentive perceptions [76].

8.4 Exploration Recommendations via Eye Tracker

Now, we provide implementation details of our prototype system and show how the major components interact together, as shown in Figure 8.1. Users start their exploratory analysis task on the *Data Exploration* interface, while an eye tracker is recording users' attention and feeding the k-NN recommender.

8.4.1 Eye Tracking Integration

We have incorporated a remote eye tracker[1] in our recommender system through the integration of an open-source message broker called ActiveMQ[2]. The message broker allows us to exchange messages between more than one client or server application. In this way, all the messages coming from the eye tracker are queued and available for multiple client applications. Both, the k-NN recommender and the user interface are able to consume eye tracking messages in real-time.

The eye gaze coordinates are calculated with respect to the screen that the person is looking at, and are represented by a pair of x and y coordinates given on the screen coordinate system. When the system is calibrated, the eye tracker calculates the user's eye gaze coordinates with average accuracy. Assuming the user sits approximately 60 cm away from the screen and tracker, this accuracy corresponds to an on-screen average error of 0.5 to 1 cm. The EyeTribe SDK returns both raw and smoothed data coordinates. The smoothed data set contains the coordinates of the estimated on-screen gaze position. To remove high variability in the raw data, the SDK incorporates a gaze data validation algorithm that maintains and analyzes a frame history of gaze frames at run time. Since users typically do not remain stationary for a long period of time and a high number of fast eye gaze jumps can occur, smoothing updates are necessary. In this way, unstable gaze coordinates can be converted to more stable and accurate gaze coordinates. In this work, we use smoothed gaze coordinates.

We use the eye tracker to collect gaze information on the entire screen space and also to detect off screen times. However, we implemented a series of AOI-hit checks that allow us to record the exploration times for when a user is exploring specific scatter plots in the SPLOM. Each scatter

[1] The Eye Tribe: `https://theeyetribe.com/`.

[2] Apache Software Foundation: `http://activemq.apache.org/`.

plot corresponds to an AOI-hit region that will start a stopwatch timer and add up the fixation counter when it is activated. With this information, we are able to distinguish the different AOIs and to calculate statistics on all explored AOIs.

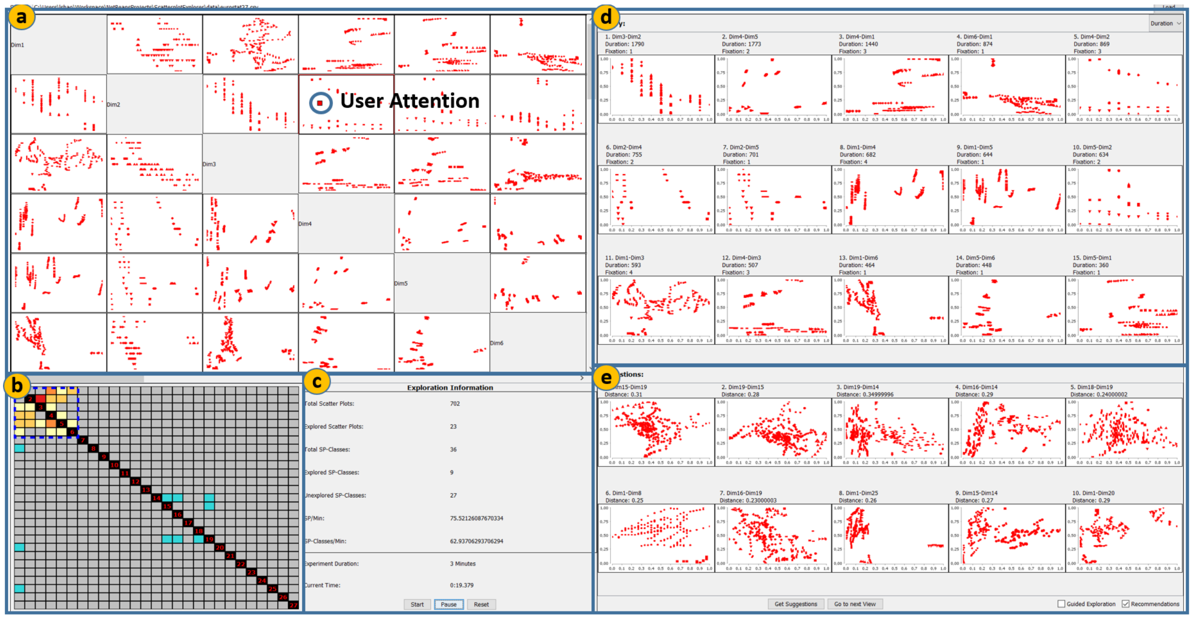
8.4.2 K-Nearest Neighbor Recommender

Our recommender includes a message listener that receives information from the eye tracker in real-time. Consequently, we continuously sum up the gaze duration and fixation counts for each scatter plot (AOI), and create a ranked scatter plot list by these values. By default, we rank the scatter plots by their total gaze duration and initiate the recommendation process after a minimum threshold of scatter plots and scatter plot classes (see Section *Visual Cluster Analysis*) have been investigated. Since our recommendation approach is based on the top N explored scatter plots, these minimum thresholds are required to guarantee that the subsequent recommendation process performs well. Firstly, the threshold for the number of minimum scatter plots (τ_{SP}) guarantees that there are at least N queries to perform, and secondly, the other threshold for scatter plot classes (τ_{SPC}) is needed to include class distinction within the queries. For instance, if the first N explored scatter plots are from the same class, we do not initiate the recommendation process, since it would perform equal queries and result in similar recommendations. This is an iterative recommendation process where $N = \tau_{SP}$ and τ_{SP} is for initiating the recommendation process. To not overwhelm the users by giving them too many recommendations, the threshold τ_{SPC} must be fulfilled anew for each iteration to execute the recommendations. In our configuration, we set $\tau_{SP} = 10$ and $\tau_{SPC} = 5$. In this case, our recommendation system starts at the earliest after 10 scatter plots and 5 scatter plot classes have been explored. It repeats the recommendation process and will recommend every time unseen scatter plots from various classes when the user explored 10 scatter plots from 5 different classes. Thereby, we create a recommendation loop that constantly recommends visually dissimilar scatter plots and learns about explored scatter plot features.

To recommend dissimilar scatter plots regarding previously seen ones, we use image descriptors based on gradient and density features. We extract edge orientation [220] and density features of the top N explored scatter plots and use these characteristics as feature vector for the dissimilarity measure. For computing density and edge features, we adapt state-of-the-art techniques from previous work [178]. Finally, we use the Euclidean distance to determine the distance between the top N explored scatter plots to the unseen scatter plots and choose scatter plots with the highest distance for each query as recommendation.

8.4.3 Recommendations and Guided Exploration

We provide the users with two different ways for inspecting the recommendations, either by manual or automatic navigation to suggested areas in the SPLOM.



Exploration Information

Total Scatter Plots: 702
Explored Scatter Plots: 23
Total SP-Classes: 36
Explored SP-Classes: 9
Unexplored SP-Classes: 27
SP/Min: 75.52126087670334
SP-Classes/Min: 62.93706293706294
Experiment Duration: 3 Minutes
Current Time: 0:19.379

y:

1. Dim3-Dim2 Duration: 1790 Fixation: 1
2. Dim4-Dim5 Duration: 1773 Fixation: 2
3. Dim4-Dim1 Duration: 1440 Fixation: 3
4. Dim6-Dim1 Duration: 874 Fixation: 1
5. Dim4-Dim2 Duration: 869 Fixation: 3
6. Dim2-Dim4 Duration: 755 Fixation: 2
7. Dim2-Dim5 Duration: 701 Fixation: 1
8. Dim1-Dim4 Duration: 682 Fixation: 4
9. Dim1-Dim5 Duration: 644 Fixation: 1
10. Dim5-Dim2 Duration: 634 Fixation: 2
11. Dim1-Dim3 Duration: 593 Fixation: 4
12. Dim4-Dim3 Duration: 507 Fixation: 3
13. Dim1-Dim6 Duration: 464 Fixation: 1
14. Dim5-Dim6 Duration: 448 Fixation: 1
15. Dim5-Dim1 Duration: 360 Fixation: 1

stions:

im15-Dim19 Distance: 0.31
2. Dim19-Dim15 Distance: 0.28
3. Dim19-Dim14 Distance: 0.34999996
4. Dim16-Dim14 Distance: 0.29
5. Dim18-Dim19 Distance: 0.24000002
6. Dim1-Dim8 Distance: 0.25
7. Dim16-Dim19 Distance: 0.23000003
8. Dim1-Dim25 Distance: 0.26
9. Dim15-Dim14 Distance: 0.27
10. Dim1-Dim20 Distance: 0.29

Figure 8.2: Our scatter plot recommender system consists of (a) the main exploration view (a given viewport of a larger SPLOM), which is monitored by the eye tracker; (b) a overview panel that shows the entire SPLOM including already explored and recommended unseen scatter plots; (c) an exploration detail view with statistics about the user analysis process; (d) ranked history view of explored scatter plots; and (e) recommendation view with suggested new scatter plots for further investigation.

Manual Exploration: The system indicates recommendations by highlighting the suggested scatter plots in the SPLOM and by displaying the plots in a ranked view (see Section *System Design*). The exploration interface is waiting for new recommendations and will highlight certain SPLOM indices in the overview representation (*analysis provenance heatmap*) and update the ranked view respectively. By this way of recommendation, we leave the users unrestricted freedom of choice in the further exploration process and just notify them about possibly interesting scatter plots. Thus, users have the option to follow the recommendation system or ignore some suggestions and explore the SPLOM by themselves. A typical mode of operation is that the user may compare the patterns in the current SPLOM viewport with the ranked list of recommended scatter plots, and decide when or if a navigation away from the current viewport is desired. Whatever the user decides, the eye tracker is still recording the exploration process and the system will continue suggesting unseen scatter plots based on the actual user attention.

Guided Exploration: If the user activates the *Guided Exploration* function, the system will automatically guide the user to the most interesting areas of the SPLOM that contains recommended scatter plots. We first compute the degree of interest of the current viewport of the user by taking the visual similarity of scatter plot patterns into account. If the current viewport is fully explored, i.e., all different scatter plot patterns are investigated, the system will guide the user to unexplored areas of the SPLOM. The only requirement is that the threshold τ_{SPC} for new recommendations must be adjusted to the grid of the user's viewport, to guarantee a continuous cycle of recommendations. We use a viewport with a 6×6 grid and a τ_{SPC} threshold of 5. To detect the most interesting area within the SPLOM, we iterate a search over the entire matrix using a sliding window approach to find an interesting viewport, which includes the most suggested AOIs. One suggested AOI increases the interestingness of a viewport to 10, whereas explored areas decrease the interest by 1. Thereby, the navigation decisions will be balanced between a high number of recommended AOIs and less explored AOIs. Consequently, users do not have to care about navigating within the SPLOM, i.e., zooming, panning and selecting interesting areas, and have the highest probability to explore undiscovered scatter plot patterns.

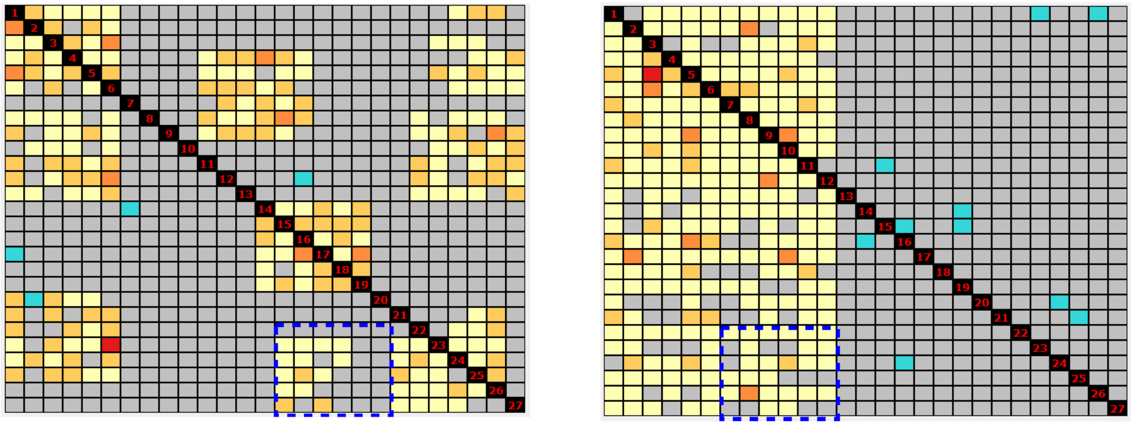
8.5 System Design

In this section, we introduce the design of our system and explain the usabilities of the individual components. Figure 8.2 shows the interface of the recommender system.

To identify the sequence of explored scatter plots, we make use of the eye tracker that measures the user's eye gazes on the exploration view - Figure 8.2 (a). The exploration view displays an enlarged subset of the SPLOM (6 × 6), which is linked to an AOI-based configuration of the eye tracker. This allows us to investigate plots in detail, e.g., for local trends or patterns, and improves the tracking mechanism of explored scatter plots.

To control the exploration view, users can either use the keyboard, apply scroll functions or directly jump to a certain area by using the navigation view - Figure 8.2 (b). Moreover, the navigation view provides a general overview of the SPLOM by showing the current viewport and visualizes already explored scatter plots and highlights recommendations. The current viewport is visualized by a blue bounding box with dashed lines and is synchronized with the exploration view. A heatmap representation with a sequential color scale shows the history of explored scatter plots, whereas darker colors indicate longer gaze durations and lighter colors indicate shorter durations (*analysis provenance heatmap*). New recommendations for exploration are emphasized as turquoise rectangles –a complementary color to the heatmap– and allow the user efficient navigation to the interesting areas of the SPLOM. By this overview representation, we additionally provide a new way of creating data provenance for SPLOM explorations. Specifically, it summarizes all user attentions during the exploration period in one raster image and can be ideally used as a reference to other exploration results. By means of the color-coding, users are able to quickly identify areas with the highest user attention and relate these across the SPLOM. Furthermore, it generates visual exploration patterns of how users explored the projection space, as shown in Figure 8.3. One can clearly observe different exploration approaches –guided exploration vs. sequential exploration– by comparing the navigation views.

The exploration detail view, shown in Figure 8.2 (c), displays additional metadata about the



(a) Guided exploration.

(b) Sequential exploration.

Figure 8.3: Illustration of two different visual patterns on how users explored a SPLOM. (a) the user explored the SPLOM by using our recommender system, one can clearly see the block patterns, which results by jumping to the suggested areas. In contrast, (b) the user explored the SPLOM sequentially with the top-down principle and from left to right.

exploration and is also the basis to evaluate our user experiments. More precisely, it shows how many individual scatter plots and scatter plot classes were explored and presents calculated mean values based on the exploration time. By these values, we evaluate the performance of the users and compare the performance under different conditions – with recommendations, without recommendations and with guided explorations.

In our approach, the history data of explored scatter plots plays an important role in creating recommendations. In contrast to feedback-driven recommendation systems, we use quantitative information directly extracted from the eye tracker, such as exploration duration or fixation count of a scatter plot, instead of using binary decisions like yes or no. Hence, we can individually adapt the recommendations and create weighted queries based on the actual user attention. For this reason, we provide a history view that shows all previously explored scatter plots and simultaneously serves as data provenance of our recommendations - Figure 8.2 (d). Just like the navigation view, all exploration information will be updated in real-time. By default, the history view is sorted by exploration duration, and we use the top ranked scatter plots as reference for searching dissimilar recommendations. Based on the exploration task, the ranking priority can be switched to e.g., fixation counts, and thus, changes the references for recommendations.

Finally, after computing the recommendations, all results will be displayed in the recommendation view - Figure 8.2 (e). This view is automatically updated when the constraints are fulfilled (see Section *Recommendation Concept*) or on user demand (via button click). This view is also linked to both the navigation view and the exploration view. By clicking on a recommended plot, the bounding box of the navigation view as well as the viewport will update to the appropriate place.

For comprehensibility and monitoring reasons, an eye cursor can be enabled, which is displayed as a small red rectangle, as shown in Figure 8.2 (a) - row 4 and 5. For instance, users can enable the eye cursor for verifying the eye tracking calibration and disabling the cursor if they feel that it would distract them from the exploration task. Another important usage is to let third parties (e.g., analysts) follow eye movements of the user and analyze their exploration processes. However, one major feature of the eye cursor is that it gives exploration information about the current viewport and notifies the user if all current scatter plot classes were discovered. Immediately after all classes have been explored, the color of the eye cursor will change from red to green and thus informs the user to explore further areas of the SPLOM.

8.6 Experimental Evaluation

In our preliminary evaluation, we test if our system allows users to discover and explore scatter plot patterns faster (more efficient) than fully interactive exploration. Therefore, we did an experimental comparison of user performance with and without recommendations.

Data Set: For our experiments, we use a data set from the *Eurostat* data repository[3] that contains approximately 5500 data sets each with information about an EU-related statistic on topics including economy, population and industry. We tested our approach on a subset containing 27 statistical attributes including population density, duration of work, electricity consumption, etc. from 28 EU countries that show temporal changes over the last decades. From these 27 dimensions, we created a SPLOM resulting in 702 scatter plots ($n \times (n-1)$) in which each data instance (point) represents one EU country at a specific year.

As a ground truth to the experiment, we clustered all 702 plots into a number of classes (see Section *Visual Cluster Analysis*), representing visually similar plots. The set of clusters represents the data set and our analysis goal is to overview as many clusters as possible in a given amount of time. The clustering procedure returned acceptable results, nevertheless we manually adjusted the clusters by their visual similarities (i.e., similar pattern, similar trend, similar density, etc.). Finally, we obtained 36 different scatter plot classes as ground truth for our experiment. Figure 8.4 shows a subset of the scatter plot clusters thereof. On average, a cluster consists of 20 scatter plots, while the largest cluster consists of 43 similar scatter plots (Figure 8.4 - bottom left) and the smallest of 2 (Figure 8.4 - top right). This ensures that the experimental task is challenging and not too easy to accomplish.

Definition of Experimental Analysis Tasks: One common problem in exploratory analysis tasks is that users have to explore a large number of plots as well as similar plots in order to find interesting patterns in the data. To show how our recommender system can help, we had users perform the following task: *Discover as many different global scatter plot patterns as possible within a given amount of time*. Each participant had three trials:

[3] European Commission: `http://ec.europa.eu/eurostat`.

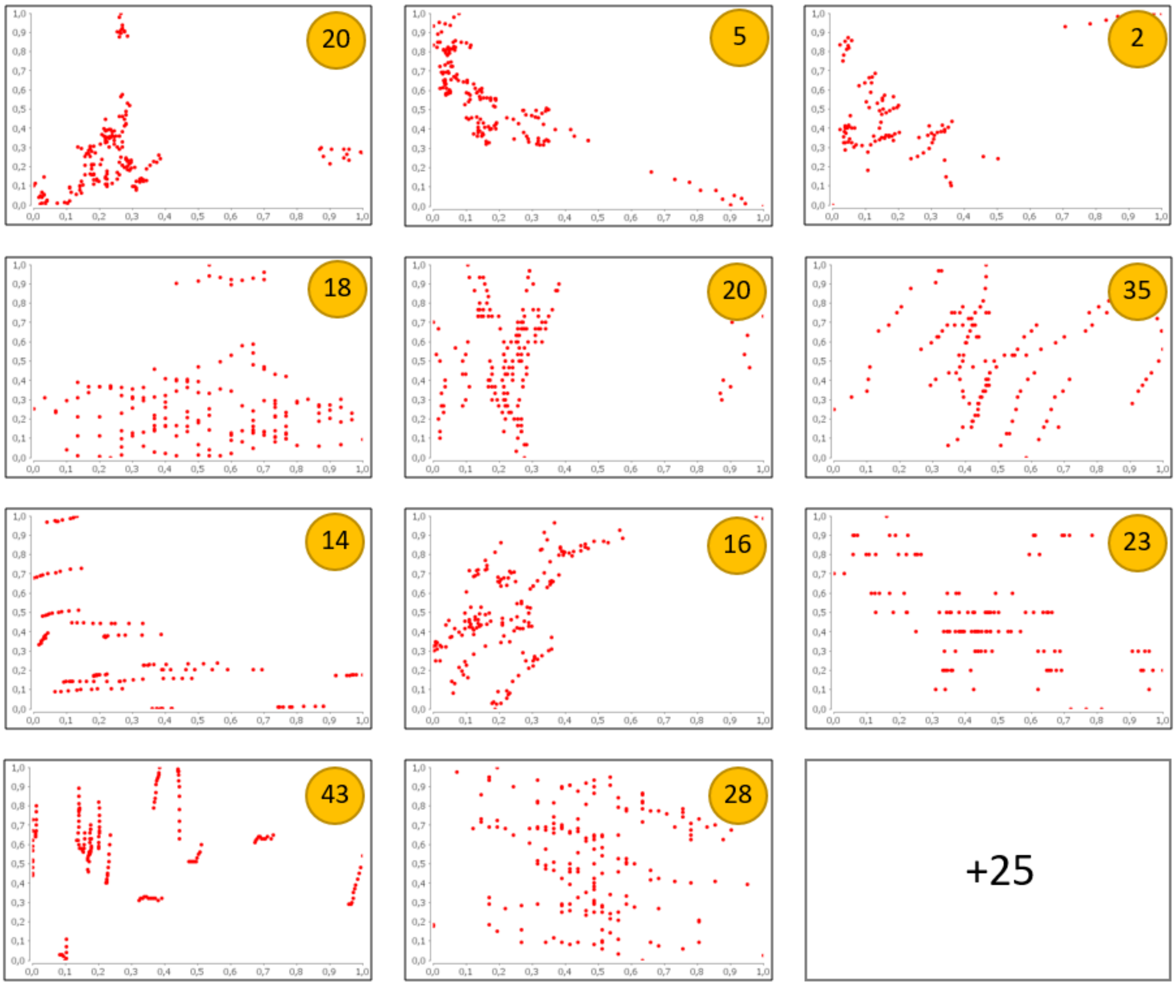


Figure 8.4: A subset of different scatter plot classes and their frequency of occurrence in the data. In total, we defined 36 scatter plot classes from 702 scatter plot views.

(1) Fully interactive SPLOM navigation; (2) Highlighting recommendations, but no navigation guidance; (3) Recommendations and navigation guidance.

We set the allowed exploration time to 2 minutes. We define as the success rate the number of unique clusters observed by the user during the exploration, and compare this across the three different system setups. For analysis of the exploration process, we captured screenshots with activated eye cursor and created a log file including mouse and keyboard events. Users were given a brief introduction to the system design and interaction techniques, and then allowed to try out the system for a first dry run. This introduction was followed by an interview session to gather initial feedback and to clarify possible issues. Afterward, the users performed the actual experiment task, while their response time and exploration success rate were recorded. Finally, a questionnaire survey on system usability was conducted.

Results: The study included 12 participants (8 male, 4 female) all of whom were graduate

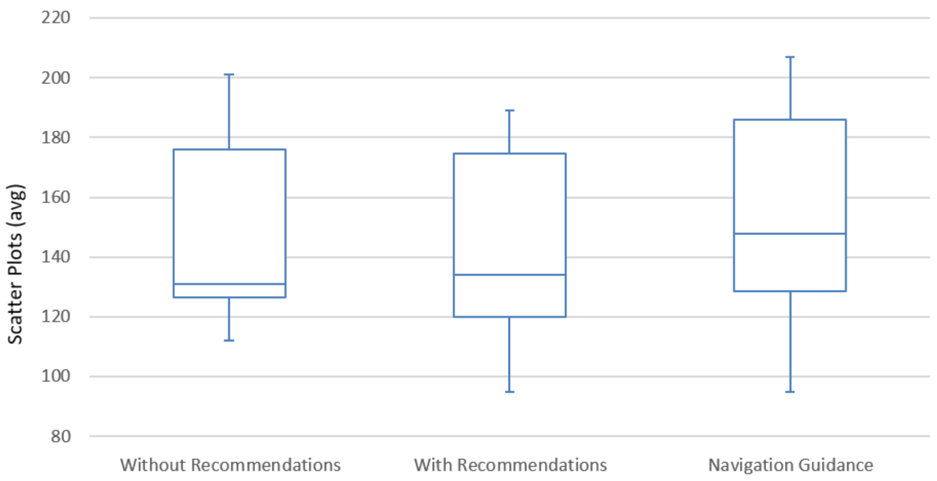


Figure 8.5: The box plot shows that participants explored less scatter plots in the 2. experiment (with recommendations) compared to the 1. experiment (without recommendations). On average the highest score was achieved in the 3. experiment.

students and had casual to moderate expertise in exploratory data analysis. To assess our approach, we observed for the three different experiment trials both the total number of explored scatter plots, as well as the number of unique scatter plot clusters. We count a scatter plot as explored, if it has been fixated for a minimum amount of time. We count a cluster to be explored if one of its members has been fixated for 200 ms. The experiment results are shown in Figure 8.5 and 8.6. Figure 8.5 shows the average number of explored scatter plots including deviation for each experiment in a box plot. On average, the participants explored 147 scatter plots in the 1. experiment, 143 in the 2. experiment and 153 scatter plots in the 3. experiment. One can see from the box plot that the participants explored less scatter plots in the 2. experiment as compared to the 1. experiment, and achieved the highest score in the 3. experiment. We observed in the 2. experiment that some participants were somewhat distracted by the recommendations and lost time while looking at the navigation view. Our questionnaire survey indicated that some participants had difficulties in navigating and finding the suggested plots in the SPLOM. Based on our observations during the experiments and the survey results, we draw the conclusion that users may need more time to process the information given by the recommender, and consequently, lost performance in exploring scatter plots and scatter plot classes respectively. This is confirmed by the fact that users explored more scatter plots by using our guided exploration function (3. experiment) where they do not have to take care of the navigation in the SPLOM.

However, if we now consider the results of explored scatter plot classes in Figure 8.6, one can clearly see the performance improvements of our participants in the second and third experiment, which include the proposed recommender approach. It is interesting to note that when considering the number of unique clusters explored, also the performance in the 2. experiment improved, although the total number of scatter plots explored was low in that experiment. In this case, the

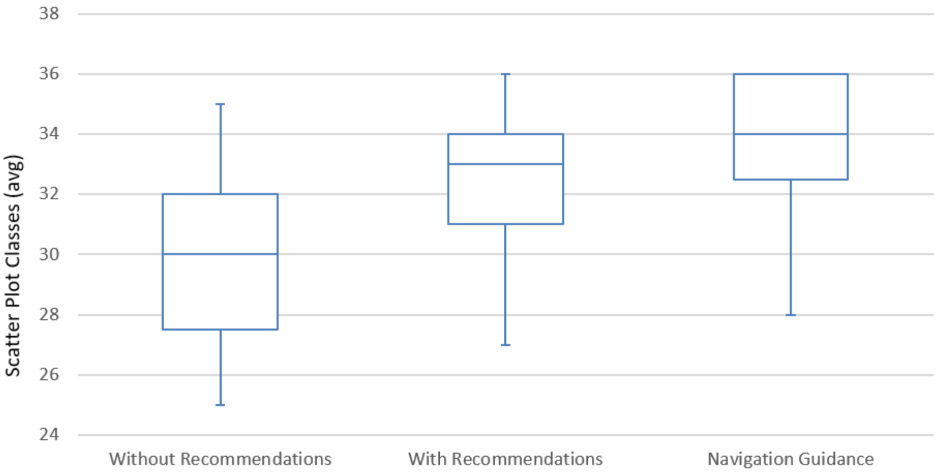


Figure 8.6: The results clearly show that the performance of the exploration of different scatter plot classes improved in the 2. and 3. experiment, which included the proposed visual recommender system.

participants explored on average 30 scatter plot classes in the 1. experiment, 32 classes in the 2. experiment and 34 classes in the 3. experiment. Furthermore, it shows that the positive trend remains for all three boxes (median, upper and lower quartiles) and their whiskers (minimum and maximum values). By comparing only the top results of the first two experiments, one can see that the best participant explored 35 scatter plot classes in the 1. experiment and 36 classes in the 2. experiment. Even better were the results in the 3. experiment where the upper quartile –splits off the highest 25% of data from the lowest 75%– achieved the best experiment results of 36 explored scatter plot classes.

The study also showed that our approach supports the exploration of various patterns in the data and helps to obtain a better overview in SPLOMs. We were able to improve the exploration rate of scatter plot classes in both experiment trials using our visual recommender system, and achieved the highest scores by using our guided exploration function.

8.7 Discussion and Extensions

We also made several observations during the study and from the questionnaire survey. It turned out that most participants preferred to freely navigate within the SPLOM, but getting visual recommendations from the system (2. experiment) and following these interactively. One participant said he enjoyed following the recommendations within the SPLOM to see the visual changes of scatter plot views along the dimensions. However, some users stated they felt lost when the guided exploration function automatically changed their exploration view. This might be improved by motion animation to show the navigation direction within the SPLOM space, helping keep up a mental map of the overall SPLOM. Also, we observed different exploration strategies users followed in the first and second experiments. This was enabled by our heatmap view. We discovered different approaches like random exploration, column-wise, row-wise or clockwise navigation. While it would be interesting to relate these navigation patterns with the exploration success rates, here we could not find sufficient evidence in our study to do such correlation analysis. However, we believe that future experiments can be done relating exploration success with eye gaze paths.

One way to improve the precision and quality of the recommendations might be to upgrade the hardware of the system, e.g., by using eye tracking glasses or larger displays. We used an affordable eye tracker with a sampling rate of 30Hz to 60Hz and ran our system on a 25" screen. To achieve good calibrations and precise eye tracking results, we created a self-made chin rest to stabilize the participants' heads. We saw difficulties to calibrate two of our initially 14 invited participants and had to cancel their experiments due to poor calibration results, probably based on reflections given by glasses.

Our approach is a first concept and implementation for eye-tracking based SPLOM exploration and can be extended in many directions. User profiles could be generated to improve the recommendations based on individual user preferences and tasks, respectively. Thus, users could give explicit interest feedback to the system and train the k-NN recommender according to their preferences. As a result, the system could especially provide weighted recommendations based on different sets of geometric or graph-theoretic features, e.g., connectivity, density or outlying. Another interesting extension would be the integration of local scatter plot pattern analysis. Thus, the system could identify locally relevant areas of scatter plots, e.g., clusters or multiple correlations, and provide recommendations based on these.

Finally, we note that the user experiment is a first step to assess the effectiveness of our novel recommendation concept. Additional measurements can be defined, including qualitative user measurements and other exploration tasks might be tested. It remains a difficult problem to measure the value of analytical insight gained from interactive systems, and evaluation approaches for eye-tracking based analytics systems can be researched in future work.

8.8 Conclusion

We introduced a novel recommendation concept for exploratory analysis tasks, in which we utilize eye tracking information to suggest interesting and unseen views. We extracted image features of previously explored samples and used them to retrieve and recommend visually dissimilar views. Thus, users can focus on their exploration task while the system automatically captures explored features and constantly supports the user with exploration notifications and guidance. We implemented a prototype system for exploration, which also includes visualization of exploration provenance data. In a user experiment, we have compared the number of scatter plots and scatter plot classes that can be explored in a given amount of time, with and without our recommendation approach. The results show that our recommender system can increase the number of scatter plot patterns during exploration, which can improve the overall analysis success. This is a first concept and we believe eye tracking has much potential to indirectly capture user information on an ongoing analysis process which can be useful to steer the analysis.

CHAPTER 9

Conclusions and Perspectives

Contents

This thesis presented novel approaches and techniques towards supporting the analysis of data patterns in high-dimensional data. In particular, I investigated methods to support visual search processes, developed interactive exploration tools and proposed novel guidance concepts. This final chapter of the dissertation summarizes all presented contributions concerning the defined research questions and outlines future research directions.

9.1 Summary of Contributions

The general objective of this dissertation was to investigate novel techniques and concepts to foster the analysis of patterns in large and complex data sets. Therefore, I have elaborated three main research goals, which were covered by the three core parts of this dissertation.

In Part I, I investigated visual search techniques to discover local patterns in high-dimensional data sets. More precisely, I presented visual search techniques for scatter plot (Chapter 3) and movement data (Chapter 4). To find interesting local patterns in scatter plot data, I proposed two approaches, a sketch-based and an example-based search, which allow defining queries more easily. Furthermore, I discussed how a minimum spanning tree clustering algorithm (MST) can be adapted to automatically extract local patterns in scatter plot data. The experiments revealed that image-based descriptors work well to find matches between user sketches and several different patterns. Moreover, it has shown that a motif dictionary can be used to create scatter plot queries and find related views that contain sets of similar patterns. In the following chapter, this visual search concept was then adapted to another data domain. I demonstrated how sketch-based search methods can also be used to find similar movement patterns in soccer games. Together with a domain expert from the German soccer club FC Bayern München, we defined domain specific analysis tasks and conducted a case study to show the usefulness of this approach. The case study has shown that our visual search tool may be extremely helpful for domain experts, e.g., couches or scouts, to retrieve interesting game situations and search for certain behavior of players.

Next, in Part II, I showed alternative exploration techniques in the case when interesting patterns are not known in advance for defining search queries. Therefore, I propose to use lens selection techniques, since they are efficient exploration tools that allow users to analyze local data properties in interactive time. In Chapter 5, I introduced the Regression Lens, a visual-interactive regression analysis tool that supports both global and local regression modeling in scatter plot visualizations. This lens concept allows users to interactively select a portion of data, on which regression analysis can be applied in interactive time. The conducted experiments showed that the Regression Lens is an effective tool for user-driven regression modeling and supports model understanding. Furthermore, I evaluated important analysis issues, such as global vs. local regression analysis, underfitted vs. overfitted models and independent vs. dependent variable, for our regression lens concept, and demonstrated the benefits of the Regression Lens with use cases. An extended version of the Regression Lens for analyzing high-dimensional data structures in subspaces is presented in Chapter 6. I presented a novel projection-based visualization approach that integrates the projection paths obtained from subspace configurations, and included lens interactions for local pattern analysis in various subspaces. This technique can be used to complement projection paths, allowing an analyst to focus on specific subsets, discover common structures and verify clusters when changing subspace configurations –variable-to-variable analysis. Analysis use cases showed that this approach helps to clarify misinterpretations like false neighbors, missing neighbors and constant structures in HD data spaces, and supports the analysis of clusters in different subspace configurations.

Finally, Part III discussed novel guidance concepts to support users in finding and exploring interesting scatter plot patterns. Visual guidance extensions for sketch-based search and Regression lens are presented in Chapter 7. I demonstrated how a shadow draw technique can be used to support the sketching process and how lens selection can be improved by recomputing automatic repositioning. Furthermore, I introduced an interestingness measure based on an adapted tf×idf-approach that incorporates local patterns and showed how interesting views can be found. In Chapter 8, I investigated whether eye tracking can be integrated into visual analytics systems to support data exploration. A user experiment has shown that it can be used to indirectly control a recommender module to suggest unseen views and thus improves the exploration of large scatter plot matrices. Furthermore, the results has shown that a recommender system including eye tracking can increase the number of explored patterns, which can improve the overall analysis success.

9.2 Future Perspectives

Each chapter in this dissertation is concluded with a comprehensive discussion about the proposed techniques, possible challenges, and future research directions. In this last chapter, I want to see my contributions as a whole and discuss general perspective of this work.

In most of my works, I have applied the presented techniques to high-dimensional data visualized by scatter plots. However, also other application domains may benefit from visual search and exploration techniques that take local patterns into account. For instance, local patterns may also occur in network data that emerge from similar arrangements or other characterizing features. In [176], I worked together with colleagues from geography and human environment department, and implemented a visual exploration tool to discover varying local patterns in street networks based on different planning approaches. Based on the identified patterns we were able to distinguish between historic development and modern planning approaches of urban networks. Street networks are an important research field that benefit strongly from local pattern analysis. Example applications include the analysis of traffic jams, urban planning and pedestrian friendliness. With respect to visual search, I also investigated the use of visual search techniques for classifying similar figures in research papers [174]. In this work, we introduced a novel approach to identify visualization types in research documents and showed first image retrieval results of our initial prototype. By integrating advanced text analysis methods automatic labeling of visualization could be further improved. In the future, we plan to develop an active learning model that should learn the essential features of local areas in visualizations (e.g., large number of edge crossings for parallel coordinates; a wider range of colors for real world images).

I also believe that visual guidance is a promising research direction for visual analytics and has considerable potential to improve analysis workflows. Although visual analytics techniques are constantly evolving and novel approaches are being published, users may still need help to find the insights or archive the goal faster. Novel approach could address the lack of human-in-the-loop processes and provide guidance approach on different VA phases, e.g., data transformation, model building, user interactions, parameter refinement. Just as there are specific visualization types for different data types, e.g., high-dimensional data, geospatial data, hierarchical data), guidance approach could also be tailored for specific applications. Additionally, multimodal interaction and sensor technologies may be integrated into visual analytics pipelines to provide guidance. The integrating eye tracking into a recommender system was a first step into that direction. Furthermore, a novel research direction in visual analytics is the integration of virtual reality (VR) and augmented reality (AR) technologies. Especially in this research area, guidance may be a useful means to guide users in a new and unusual environment.

9.3 Concluding Remarks

During my doctoral program, I noticed that most visual analytics approaches focus on detecting and analyzing global data properties. I believe that also important information may be hidden in local properties of a data set and appropriate visual analytics methods are needed to discover these insights. Thus, this dissertation focuses on the integration of novel visual search, exploration techniques and guidance approaches for local patterns into visual analytics processes. The major research questions of:

1. How can we find local patterns in high-dimensional data sets?
2. How can we help the user to efficiently explore and search for local patterns?
3. How can we support the user in finding interesting local patterns?

has been addressed in the previous chapters. This dissertation provides exemplary solutions to these research questions and proposes directions for further investigations. I hope that these perspectives and conducted experiments will guide and inspire other researchers from all different research communities to conduct further research in this important area.

List of Figures

FIGURE **Page**

LIST OF TABLES

TABLE **Page**

BIBLIOGRAPHY

[1] G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, June 2005.

[2] G. Albuquerque, M. Eisemann, D. J. Lehmann, H. Theisel, and M. Magnor, “Quality-based visualization matrices,” in *Proc. Vision, Modeling and Visualization (VMV)*. Eurographics, Nov 2009, pp. 341–349.

[3] A. Anand, L. Wilkinson, and T. N. Dang, “Visual pattern discovery using random projections,” in *IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2012, pp. 43–52.

[4] G. L. Andrienko, N. V. Andrienko, M. Burch, and D. Weiskopf, “Visual Analytics Methodology for Eye Movement Studies,” *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2889–2898, 2012.

[5] G. L. Andrienko, N. V. Andrienko, C. Hurter, S. Rinzivillo, and S. Wrobel, “From movement tracks through events to places: Extracting and characterizing significant places from mobility data,” in *IEEE Conference on Visual Analytics Science and Technology (VAST)*, Oct 2011, pp. 161–170.

[6] G. L. Andrienko, N. V. Andrienko, P. Bak, D. A. Keim, and S. Wrobel, *Visual Analytics of Movement*. Springer Science & Business Media, 2013. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-37583-5

[7] N. V. Andrienko and G. L. Andrienko, *Exploratory analysis of spatial and temporal data - a systematic approach*. Springer Science & Business Media, 2006. [Online]. Available: http://dx.doi.org/10.1007/3-540-31190-4

[8] N. V. Andrienko and G. L. Andrienko, “Visual analytics of movement: An overview of methods, tools and procedures,” *Information Visualization*, vol. 12, no. 1, pp. 3–24, 2013. [Online]. Available: https://doi.org/10.1177/1473871612457601

[9] F. Angiulli and C. Pizzuti, “Fast Outlier Detection in High Dimensional Spaces,” in *Proc. of the 6th European Conference on Principles of Data Mining and Knowledge Discovery*, 2002, pp. 15–26.

[10] M. Ankerst, S. Berchtold, and D. A. Keim, “Similarity clustering of dimensions for an enhanced visualization of multidimensional data,” in *Proceedings IEEE Symposium on Information Visualization*, Oct 1998, pp. 52–60.

[11] F. J. Anscombe, “Graphs in Statistical Analysis,” *The American Statistician*, vol. 27, no. 1, pp. 17–21, 1973.

[12] D. Asimov, “The Grand Tour: A Tool for Viewing Multidimensional Data,” *SIAM J. Sci. Stat. Comput.*, vol. 6, no. 1, pp. 128–143, Jan. 1985. [Online]. Available: http://dx.doi.org/10.1137/0906011

[13] R. A. Becker and W. S. Cleveland, “Brushing Scatterplots,” *Technometrics*, vol. 29, no. 2, p. 127–142, May 1987. [Online]. Available: https://doi.org/10.2307/1269768

[14] M. Behrisch, B. Bach, M. Hund, M. Delz, L. Von Rüden, J. Fekete, and T. Schreck, “Magnostics: Image-Based Search of Interesting Matrix Views for Guided Network Exploration,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 31–40, Jan 2017.

[15] M. Behrisch, M. Blumenschein, N. W. Kim, L. Shao, M. El-Assady, J. Fuchs, D. Seebacher, A. Diehl, U. Brandes, H. Pfister, T. Schreck, D. Weiskopf, and D. A. Keim, “Quality Metrics for Information Visualization,” *Computer Graphics Forum*, vol. 37, no. 3, pp. 625–662, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13446

[16] M. Behrisch, F. Korkmaz, L. Shao, and T. Schreck, “Feedback-driven interactive exploration of large multidimensional data supported by visual classifier,” in *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)*, Oct 2014, pp. 43–52.

[17] M. Behrisch, D. Streeb, F. Stoffel, D. Seebacher, B. Matejek, S. H. Weber, S. Mittelstädt, H. Pfister, and D. Keim, “Commercial Visual Analytics Systems–Advances in the Big Data Analytics Field,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 10, pp. 3011–3031, 2019.

[18] M. Behrisch, B. Bach, N. Henry Riche, T. Schreck, and J.-D. Fekete, “Matrix Reordering Methods for Table and Network Visualization,” *Computer Graphics Forum*, vol. 35, no. 3, pp. 693–716, Jun. 2016. [Online]. Available: https://doi.org/10.1111/cgf.12935

[19] J. Bernard, J. Brase, D. Fellner, O. Koepler, J. Kohlhammer, T. Ruppert, T. Schreck, and I. Sens, “A visual digital library approach for time-oriented scientific primary data,” *International Journal on Digital Libraries*, vol. 11, no. 2, pp. 111–123, Jun 2010. [Online]. Available: https://doi.org/10.1007/s00799-011-0072-x

[20] J. Bertin, *Graphics and Graphic Information Processing*. De Gruyter, 1981. [Online]. Available: https://www.degruyter.com/view/title/10300

[21] E. Bertini, A. Tatu, and D. A. Keim, “Quality Metrics in High-Dimensional Data Visualization: An Overview and Systematization,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2203–2212, Dec 2011.

[22] E. Bertini and D. Lalanne, “Surveying the Complementary Role of Automatic Data Analysis and Visualization in Knowledge Discovery,” in *Proceedings of the ACM SIGKDD Workshop on Visual Analytics and Knowledge Discovery*. ACM, 2009, pp. 12–20. [Online]. Available: http://doi.acm.org/10.1145/1562849.1562851

[23] E. Bertini, M. Rigamonti, and D. Lalanne, “Extended Excentric Labeling,” *Computer Graphics Forum*, vol. 28, no. 3, pp. 927–934, 2009. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2009.01456.x

[24] A. Bialkowski, P. Lucey, P. Carr, Y. Yue, S. Sridharan, and I. Matthews, “Large-Scale Analysis of Soccer Matches Using Spatiotemporal Tracking Data,” in *Data Mining (ICDM), 2014 IEEE International Conference on*, Dec 2014, pp. 725–730.

[25] E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose, “Toolglass and Magic Lenses: The See-through Interface,” in *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH ’93. ACM, 1993, pp. 73–80. [Online]. Available: http://doi.acm.org/10.1145/166117.166126

[26] T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl, “State-of-the-Art of Visualization for Eye Tracking Data,” in *EuroVis - STARs*, R. Borgo, R. Maciejewski, and I. Viola, Eds. The Eurographics Association, 2014.

[27] A. Bojko, “Informative or Misleading? Heatmaps Deconstructed,” in *Proceedings of the 13th International Conference on Human-Computer Interaction*, 2009, p. 30–39. [Online]. Available: https://doi.org/10.1007/978-3-642-02574-7_4

[28] T. Boren and J. Ramey, “Thinking aloud: reconciling theory and practice,” *IEEE Transactions on Professional Communication*, vol. 43, no. 3, pp. 261–278, 2000.

[29] H. Bosch, D. Thom, F. Heimerl, E. Püttmann, S. Koch, R. Krüger, M. Wörner, and T. Ertl, “ScatterBlogs2: Real-Time Monitoring of Microblog Messages Through User-Guided

Filtering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2022–2031, 2013.

[30] F. Bouali, A. Guettala, and G. Venturini, "VizAssist: An Interactive User Assistant for Visual Data Mining," *Vis. Comput.*, vol. 32, no. 11, pp. 1447–1463, Nov. 2016. [Online]. Available: http://dx.doi.org/10.1007/s00371-015-1132-9

[31] S. Bremm, T. von Landesberger, J. Bernard, and T. Schreck, "Assisted Descriptor Selection Based on Visual Comparative Data Analysis," *Computer Graphics Forum*, vol. 30, no. 3, pp. 891–900, 2011. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2011.01938.x

[32] M. Brooks, S. Amershi, B. Lee, S. M. Drucker, A. Kapoor, and P. Simard, "FeatureInsight: Visual support for error-driven feature ideation in text classification," in *IEEE Conference on Visual Analytics Science and Technology*, 2015, pp. 105–112.

[33] J. Browne, B. Lee, S. Carpendale, N. Riche, and T. Sherwood, "Data Analysis on Interactive Whiteboards through Sketch-Based Interaction," in *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. Association for Computing Machinery, 2011, p. 154–157. [Online]. Available: https://doi.org/10.1145/2076354.2076383

[34] S. Butscher, S. Hubenschmid, J. Müller, J. Fuchs, and H. Reiterer, "Clusters, Trends, and Outliers: How Immersive Technologies Can Facilitate the Collaborative Analysis of Multidimensional Data," in *Proceedings of the Conference on Human Factors in Computing Systems*, ser. CHI '18. ACM, 2018, pp. 90:1–90:12. [Online]. Available: http://doi.acm.org/10.1145/3173574.3173664

[35] Y. Cao, H. Wang, C. Wang, Z. Li, L. Zhang, and L. Zhang, "MindFinder: Interactive Sketch-based Image Search on Millions of Images," in *Proceedings of the International Conference on Multimedia*. ACM, 2010, pp. 1605–1608.

[36] D. B. Carr, R. J. Littlefield, and W. L. Nichloson, "Scatterplot matrix techniques for large n," *Proceedings of the Seventeenth Symposium on the Interface of Computer Sciences and Statistics on Computer Science and Statistics*, pp. 297–306, 1986.

[37] M. Cavallo and c. Demiralp, *A Visual Interaction Framework for Dimensionality Reduction Based Data Exploration*. Association for Computing Machinery, 2018, p. 1–13. [Online]. Available: https://doi.org/10.1145/3173574.3174209

[38] D. Ceneda, T. Gschwandtner, T. May, S. Miksch, H.-J. Schulz, M. Streit, and C. Tominski, "Characterizing Guidance in Visual Analytics," *IEEE Transactions on Visualization*

and Computer Graphics, vol. 23, no. 1, pp. 111–120, Jan. 2017. [Online]. Available: https://doi.org/10.1109/TVCG.2016.2598468

[39] Y. Chan, C. D. Correa, and K. Ma, "Flow-based scatterplots for sensitivity analysis," in *2010 IEEE Symposium on Visual Analytics Science and Technology*, Oct 2010, pp. 43–50.

[40] H. Chen, W. Chen, H. Mei, Z. Liu, K. Zhou, W. Chen, W. Gu, and K. L. Ma, "Visual Abstraction and Exploration of Multi-class Scatterplots," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 1683–1692, Dec 2014.

[41] W. S. Cleveland, *Visualizing Data*. Hobart Press, 1993.

[42] W. S. Cleveland, M. E. McGill, and R. McGill, "The Shape Parameter of a Two-Variable Graph," *Journal of the American Statistical Association*, vol. 83, no. 402, pp. 289–300, 1988. [Online]. Available: http://www.jstor.org/stable/2288843

[43] D. Cook, A. Buja, J. Cabreta, and C. Hurley, "Grand tour and projection pursuit," *Journal of Computational and Statistical Computing*, vol. 4, no. 3, pp. 155–172, 1995.

[44] K. A. Cook and J. J. Thomas, "Illuminating the Path: The Research and Development Agenda for Visual Analytics," 5 2005.

[45] M. A. A. Cox and T. F. Cox, *Multidimensional Scaling*. Springer Berlin Heidelberg, 2008, pp. 315–347. [Online]. Available: https://doi.org/10.1007/978-3-540-33037-0_14

[46] H. Cramér and H. Wold, "Some theorems on distribution functions," *Journal of the London Mathematical Society*, no. 4, 1936.

[47] G. C. De Silva, T. Yamasaki, and K. Aizawa, "Sketch-Based Spatial Queries for Retrieving Human Locomotion Patterns From Continuously Archived GPS Data," *IEEE Transactions on Multimedia*, vol. 11, no. 7, pp. 1240–1253, Nov 2009.

[48] M. Dowling, J. E. Wenskovitch, J. T. Fry, S. Leman, L. House, and C. North, "SIRIUS: Dual, Symmetric, Interactive Dimension Reductions," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, pp. 172–182, 2018.

[49] M. Dry, D. Navarro, and M. Lee, "The perceptual organization of point constellations," 01 2009.

[50] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "Sketch-Based Image Retrieval: Benchmark and Bag-of-Features Descriptors," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 17, no. 11, pp. 1624–1636, Nov 2011.

[51] M. Eitz, R. Richter, T. Boubekeur, K. Hildebrand, and M. Alexa, "Sketch-Based Shape Retrieval," *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 31, no. 4, pp. 31:1–31:10, 2012.

[52] G. Ellis, E. Bertini, and A. Dix, "The Sampling Lens: Making Sense of Saturated Visualisations," in *CHI '05 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2005, pp. 1351–1354.

[53] G. Ellis and A. Dix, "The Plot, the Clutter, the Sampling and Its Lens: Occlusion Measures for Automatic Clutter Reduction," in *Proceedings of the Working Conference on Advanced Visual Interfaces*, 2006.

[54] N. Elmqvist, P. Dragicevic, and J.-D. Fekete, "Rolling the Dice: Multidimensional Visual Exploration using Scatterplot Matrix Navigation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1141–1148, 2008.

[55] M. J. Enenhofer, "Spatial-Query-by-Sketch," in *Proceedings 1996 IEEE Symposium on Visual Languages*, Sep. 1996, pp. 60–67.

[56] R. Engels, "Planning Tasks for Knowledge Discovery in Databases; Performing Task-oriented User-guidance," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96. AAAI Press, 1996, pp. 170–175. [Online]. Available: http://dl.acm.org/citation.cfm?id=3001460.3001496

[57] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise." *Proc. of the Second International Conference on Knowledge Discovery and Data Mining*, 1996.

[58] R. Etemadpour, B. Olk, and L. Linsen, "Eye-tracking investigation during visual analysis of projected multidimensional data with 2D scatterplots," in *International Conference on Information Visualization Theory and Applications*, 2014, pp. 233–246.

[59] M. Fink, J.-H. Haunert, J. Spoerhase, and A. Wolff, "Selecting the aspect ratio of a scatter plot based on its delaunay triangulation." *IEEE transactions on visualization and computer graphics*, vol. 19, no. 12, pp. 2326–35, Dec. 2013.

[60] D. Freedman, *Statistical Models: Theory and Practice*. Cambridge University Press, 2005.

[61] J. H. Friedman and J. W. Tukey, "A Projection Pursuit Algorithm for Exploratory Data Analysis," *IEEE Trans. Comput.*, vol. 23, 1974.

[62] J. H. Friedman, "Exploratory projection pursuit," *Journal of the American Statistical Association*, vol. 82, no. 397, pp. 249–266, 1987.

[63] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs, "A Search Engine for 3D Models," *ACM Trans. Graph.*, vol. 22, no. 1, pp. 83–105, Jan. 2003. [Online]. Available: http://doi.acm.org/10.1145/588272.588279

[64] L. A. García-Escudero, A. Gordaliza, C. Matrán, and A. Mayo-Iscar, “A Robust Maximal F-Ratio Statistic to Detect Clusters Structure,” *Communications in Statistics - Theory and Methods*, vol. 38, no. 5, pp. 682–694, 2009. [Online]. Available: https://doi.org/10.1080/03610920802287297

[65] R. Gasteiger, M. Neugebauer, O. Beuing, and B. Preim, “The FLOWLENS: A Focus-and-Context Visualization Approach for Exploration of Blood Flow in Cerebral Aneurysms,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, pp. 2183–2192, 2011.

[66] A. Gionis, H. Mannila, and P. Tsaparas, “Clustering aggregation,” *ACM Transactions on Knowledge Discovery from Data*, vol. 1, no. 1, pp. 4–es, Mar. 2007.

[67] J. H. Goldberg and A. M. Wichansky, “Eye Tracking in Usability Evaluation: A Practitioner’s Guide,” in *The Mind’s Eye*, 2003, pp. 493 – 516.

[68] K. Goldsberry, “Courtvision: New visual and spatial analytics for the nba,” in *2012 MIT Sloan Sports Analytics Conference*, 2012.

[69] S. Gratzl, N. Gehlenborg, A. Lex, H. Pfister, and M. Streit, “Domino: Extracting, Comparing, and Manipulating Subsets Across Multiple Tabular Datasets,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 2023–2032, Dec 2014.

[70] J. Gudmundsson and T. Wolle, “Towards automated football analysis: Algorithms and data structures,” in *Proc. 10th Australasian Conf. on Mathematics and Computers in Sport*. Citeseer, 2010.

[71] J. Gudmundsson and T. Wolle, “Football analysis using spatio-temporal tools,” *Computers, Environment and Urban Systems*, no. 47, pp. 16–27, 2014.

[72] Z. Guo, M. O. Ward, and E. A. Rundensteiner, “Model space visualization for multivariate linear trend discovery,” in *2009 IEEE Symposium on Visual Analytics Science and Technology*, 2009, pp. 75–82.

[73] Z. Guo, M. O. Ward, E. A. Rundensteiner, and C. Ruiz, “Pointwise local pattern exploration for sensitivity analysis,” in *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2011, pp. 131–140.

[74] M. Harrower and C. A. Brewer, “ColorBrewer.org: An Online Tool for Selecting Colour Schemes for Maps,” *The Cartographic Journal*, vol. 40, no. 1, pp. 27–37, 2003.

[75] M. M. Hayhoe, “Advances in Relating Eye Movements and Cognition,” *Infancy*, vol. 6, no. 2, pp. 267–274, 2004.

[76] C. Healey and J. Enns, “Attention and Visual Memory in Visualization and Computer Graphics,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 7, 2012.

[77] P. Heim, S. Lohmann, D. Tsendragchaa, and T. Ertl, “SemLens: Visual Analysis of Semantic Data with Scatter Plots and Semantic Lenses,” in *Proceedings of the 7th International Conference on Semantic Systems*. ACM, 2011, pp. 175–178.

[78] F. Heimerl, S. Koch, H. Bosch, and T. Ertl, “Visual Classifier Training for Text Document Retrieval,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, 2012.

[79] N. Heulot, M. Aupetit, and J.-D. Fekete, “ProxiLens: Interactive Exploration of High-Dimensional Data using Projections,” in *EuroVis Workshop on Visual Analytics using Multidimensional Projections*. The Eurographics Association, 2013.

[80] H. Hochheiser and B. Shneiderman, “Visual Specification of Queries for Finding Patterns in Time-Series Data,” 05 2001.

[81] H. Hochheiser and B. Shneiderman, “Dynamic Query Tools for Time Series Data Sets: Timebox Widgets for Interactive Exploration,” *Information Visualization*, vol. 3, no. 1, pp. 1–18, 2004. [Online]. Available: https://doi.org/10.1057/palgrave.ivs.9500061

[82] B. Höferlin, R. Netzel, M. Höferlin, D. Weiskopf, and G. Heidemann, “Inter-Active Learning of Ad-Hoc Classifiers for Video Visual Analytics,” in *IEEE Conference on Visual Analytics Science and Technology*, 2012, pp. 23–32.

[83] P. Hoffman and G. Grinstein, *A Survey of Visualizations for High-Dimensional Data Mining*. Morgan Kaufmann Publishers Inc., 2001, p. 47–82.

[84] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley, “DNA Visual and Analytic Data Mining,” in *Proceedings of the 8th Conference on Visualization ’97*. IEEE Computer Society Press, 1997, pp. 437–ff.

[85] K. Holmqvist, M. Nystrom, R. Andersson, R. Dewhurst, H. Jarodzka, and J. van de Weijer, “Eye Tracking. A comprehensive guide to methods and measures,” 2011.

[86] C. Holz and S. Feiner, “Relaxed Selection Techniques for Querying Time-series Graphs,” in *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology*, ser. UIST ’09. ACM, 2009, pp. 213–222. [Online]. Available: http://doi.acm.org/10.1145/1622176.1622217

[87] R. Hu and J. Collomosse, “A Performance Evaluation of Gradient Field HOG Descriptor for Sketch Based Image Retrieval,” *Comput. Vis. Image Underst.*, vol. 117, no. 7, pp. 790–806, Jul. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.cviu.2013.02.005

[88] S. Hubenschmid, J. Zagermann, S. Butscher, and H. Reiterer, "Employing Tangible Visualisations in Augmented Reality with Mobile Devices," in *MultimodalVis '18 Workshop at AVI 2018*, 2018.

[89] P. J. Huber, "Projection pursuit," *Ann. Statist.*, vol. 13, no. 2, pp. 435–475, 06 1985.

[90] C. Hurter, A. Telea, and O. Ersoy, "MoleView: An Attribute and Structure-Based Semantic Lens for Large Element-Based Plots," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2600–2609, Dec 2011.

[91] P. Indyk and R. Motwani, "Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality," in *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, ser. STOC '98. ACM, 1998, pp. 604–613. [Online]. Available: http://doi.acm.org/10.1145/276698.276876

[92] S. G. J. Staib and S. Gumhold, "Enhancing Scatterplots with Multi-Dimensional Focal Blur," *Computer Graphics Forum*, 2016.

[93] R. J. Jacob, "The Use of Eye Movements in Human-computer Interaction Techniques: What You Look at is What You Get," *ACM Trans. Inf. Syst.*, vol. 9, no. 2, pp. 152–169, 1991.

[94] R. J. Jacob and K. S. Karn, "Eye Tracking in Human-Computer Interaction and Usability Research: Ready to Deliver the Promises," in *The Mind's Eye*, 2003, pp. 573 – 605.

[95] S. Jan, J. Wang, A. Schwering, and M. Chipofya, "Ordering: A reliable qualitative information for the alignment of sketch and metric maps," in *2013 IEEE 12th International Conference on Cognitive Informatics and Cognitive Computing*, July 2013, pp. 203–211.

[96] P. Jana and A. Naik, "An efficient minimum spanning tree based clustering algorithm," in *Proc. Int. Conference on Methods and Models in Computer Science*, 2009.

[97] H. Janetzko, D. Sacha, M. Stein, T. Schreck, D. A. Keim, and O. Deussen, "Feature-Driven Visual Analytics of Soccer Data," in *Proceedings of the 2014 IEEE Symposium on Visual Analytics Science and Technology (VAST '14)*, 2014.

[98] D. H. Jeong, C. Ziemkiewicz, B. Fisher, W. Ribarsky, and R. Chang, "iPCA: An Interactive System for PCA-based Visual Analytics," *Computer Graphics Forum*, vol. 28, no. 3, pp. 767–774, 2009.

[99] Jing Yang, Wei Peng, M. O. Ward, and E. A. Rundensteiner, "Interactive hierarchical dimension ordering, spacing and filtering for exploration of high dimensional datasets," in *IEEE Symposium on Information Visualization 2003 (IEEE Cat. No.03TH8714)*, Oct 2003, pp. 105–112.

[100] I. Jolliffe, *Principal component analysis*, ser. Springer series in statistics. Springer-Verlag, 1986.

[101] I. Jusufi, Y. Dingjie, and A. Kerren, "The Network Lens: Interactive Exploration of Multivariate Networks Using Visual Filtering," in *14th International Conference Information Visualisation*, 2010, pp. 35–42.

[102] D. Jäckle, M. Hund, M. Behrisch, D. A. Keim, and T. Schreck, "Pattern trails: Visual analysis of pattern transitions in subspaces," in *IEEE Conference on Visual Analytics Science and Technology*, 2017.

[103] E. Kandogan, "Star Coordinates: A Multi-dimensional Visualization Technique with Uniform Treatment of Dimensions," in *In Proceedings of the IEEE Information Visualization Symposium, Late Breaking Hot Topics*, 2000, pp. 9–12.

[104] E. Kandogan, "Visualizing Multi-dimensional Clusters, Trends, and Outliers Using Star Coordinates," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '01. ACM, 2001, pp. 107–116. [Online]. Available: http://doi.acm.org/10.1145/502512.502530

[105] J. Kehrer and H. Hauser, "Visualization and Visual Analysis of Multi-faceted Scientific Data: A Survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 3, pp. 495–513, Mar. 2013, spotlight paper of the March issue of TVCG. [Online]. Available: /research/publications/2013/Kehrer-2013-STAR/

[106] D. A. Keim, "Information visualization and visual data mining," *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, no. 1, pp. 1–8, Jan 2002.

[107] D. A. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, *Visual Analytics: Definition, Process, and Challenges*. Springer Berlin Heidelberg, 2008, pp. 154–175. [Online]. Available: https://doi.org/10.1007/978-3-540-70956-5_7

[108] D. A. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann, Eds., *Mastering the information age : solving problems with visual analytics*. Goslar : Eurographics Association, 2010. [Online]. Available: https://diglib.eg.org/handle/10.2312/14803

[109] D. A. Keim, F. Mansmann, J. Schneidewind, J. Thomas, and H. Ziegler, *Visual Analytics: Scope and Challenges*. Springer Berlin Heidelberg, 2008, pp. 76–90. [Online]. Available: https://doi.org/10.1007/978-3-540-71080-6_6

[110] M. Kreuseler, N. Lopez, and H. Schumann, "A scalable framework for information visualization," in *Proceedings of IEEE Symposium on Information Visualization*, Oct 2000, pp. 27–36.

[111] R. Krüger, D. Thom, M. Wörner, H. Bosch, and T. Ertl, “TrajectoryLenses - a Set-based Filtering and Exploration Technique for Long-term Trajectory Data,” in *Proceedings of the 15th Eurographics Conference on Visualization*, ser. EuroVis ’13, 2013, pp. 451–460.

[112] E. LaMar, B. Hamann, and K. I. Joy, “A magnification lens for interactive volume visualization,” in *Proceedings Ninth Pacific Conference on Computer Graphics and Applications*, 2001, pp. 223–232.

[113] J. Lee and T. Funkhouser, “Sketch-based search and composition of 3D models,” in *EUROGRAPHICS Workshop on Sketch-Based Interfaces and Modeling*, Jun. 2008.

[114] Y. J. Lee, C. L. Zitnick, and M. F. Cohen, “ShadowDraw: Real-time User Guidance for Freehand Drawing,” in *ACM SIGGRAPH 2011 Papers*. ACM, 2011, pp. 27:1–27:10.

[115] P. A. Legg, D. H. S. Chung, M. L. Parry, R. Bown, M. W. Jones, I. W. Griffiths, and M. Chen, “Transformation of an Uncertain Video Search Pipeline to a Sketch-Based Visual Analytics Loop,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2109–2118, 2013.

[116] D. J. Lehmann, F. Kemmler, T. Zhyhalava, M. Kirschke, and H. Theisel, “Visualnostics: Visual Guidance Pictograms for Analyzing Projections of High-dimensional Data,” *Computer Graphics Forum*, vol. 34, no. 3, 2015.

[117] D. J. Lehmann and H. Theisel, “Orthographic Star Coordinates,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2615–2624, 2013.

[118] D. J. Lehmann and H. Theisel, “General Projective Maps for Multidimensional Data Projection,” *Computer Graphics Forum*, vol. 35, no. 2, 2016.

[119] D. J. Lehmann, G. Albuquerque, M. Eisemann, M. Magnor, and H. Theisel, “Selecting Coherent and Relevant Plots in Large Scatterplot Matrices,” *Computer Graphics Forum*, vol. 31, no. 6, pp. 1895–1908, Apr. 2012.

[120] Y. K. Leung and M. D. Apperley, “A Review and Taxonomy of Distortion-oriented Presentation Techniques,” *ACM Trans. Comput.-Hum. Interact.*, vol. 1, no. 2, pp. 126–160, Jun. 1994.

[121] V. I. Levenshtein, “Binary codes capable of correcting deletions, insertions, and reversals,” in *Soviet physics doklady*, vol. 10, no. 8, 1966, pp. 707–710.

[122] J. Li, J.-B. Martens, and J. J. van Wijk, “Judging Correlation from Scatterplots and Parallel Coordinate Plots,” *Information Visualization*, vol. 9, no. 1, pp. 13–30, Mar. 2010.

[123] M. Lichman, “UCI machine learning repository,” 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[124] S. Liu, D. Maljovec, B. Wang, P. Bremer, and V. Pascucci, "Visualizing High-Dimensional Data: Advances in the Past Decade," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 3, pp. 1249–1268, 2017.

[125] S. Liu, B. Wang, J. J. Thiagarajan, P.-T. Bremer, and V. Pascucci, "Visual Exploration of High-Dimensional Data through Subspace Analysis and Dynamic Projections," *Computer Graphics Forum*, vol. 34, no. 3, pp. 271–280, 2015.

[126] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of Internal Clustering Validation Measures," in *Proceedings of the IEEE International Conference on Data Mining*, ser. ICDM '10. IEEE Computer Society, 2010, pp. 911–916.

[127] J. Loffler, "Content-based retrieval of 3d models in distributed web databases by visual shape information," in *IEEE Conference on Information Visualization. An International Conference on Computer Visualization and Graphics*, July 2000, pp. 82–87.

[128] P. Lops, M. de Gemmis, and G. Semeraro, *Content-based Recommender Systems: State of the Art and Trends*, 2011.

[129] P. Lucey, A. Bialkowski, M. Monfort, P. Carr, and I. Matthews, ""Quality vs Quantity": Improved Shot Prediction in Soccer using Strategic Features from Spatiotemporal Data." MIT Sloan Sports Analytics Conference, 2014.

[130] J. Mackinlay, P. Hanrahan, and C. Stolte, "Show Me: Automatic Presentation for Visual Analysis," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1137–1144, Nov 2007.

[131] J. Matute, A. C. Telea, and L. Linsen, "Skeleton-Based Scagnostics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 542–552, Jan 2018.

[132] A. Mayorga and M. Gleicher, "Splatterplots: Overcoming Overdraw in Scatter Plots," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 9, pp. 1526–1538, Sep. 2013.

[133] C. Mehrotra, N. Chitransh, and A. Singh, "Scope and challenges of visual analytics: A survey," in *2017 International Conference on Computing, Communication and Automation (ICCCA)*, 2017, pp. 1229–1234.

[134] D. Mladenić, "Automated Model Selection," in *Proceedings of the Knowledge Level Modelling and Machine Learning Workshop, Crete*, 1995.

[135] R. Motta, R. Minghim, A. de Andrade Lopes, and M. C. F. Oliveira, "Graph-based measures to assist user assessment of multidimensional projections," *Neurocomputing*, vol. 150, Part B, no. 0, pp. 583 – 598, 2015.

[136] T. Mühlbacher and H. Piringer, “A Partition-Based Framework for Building and Validating Regression Models,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 1962–1971, Dec 2013.

[137] P. K. Muthumanickam, K. Vrotsou, M. Cooper, and J. Johansson, “Shape grammar extraction for efficient query-by-sketch pattern matching in long time series,” in *IEEE Conference on Visual Analytics Science and Technology (VAST)*, Oct 2016, pp. 121–130.

[138] J. E. Nam and K. Mueller, “TripAdvisor^{N-D}: A Tourism-Inspired High-Dimensional Space Exploration Framework with Overview and Detail,” *IEEE Transactions on Visualization and Computer Graphics*, 2013.

[139] L. G. Nonato and M. Aupetit, “Multidimensional Projection for Visual Analytics: Linking Techniques with Distortions, Tasks, and Layout Enrichment,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 8, pp. 2650–2673, 2019.

[140] C. North, T. Dwyer, B. Lee, D. Fisher, P. Isenberg, G. Robertson, and K. Inkpen, “Understanding multi-touch manipulation for surface computing,” in *Human-Computer Interaction – INTERACT 2009*, T. Gross, J. Gulliksen, P. Kotzé, L. Oestreicher, P. Palanque, R. O. Prates, and M. Winckler, Eds. Springer Berlin Heidelberg, 2009, pp. 236–249.

[141] L. Nováková and O. Štěpánková, “Visualization of trends using radviz,” *Journal of Intelligent Information Systems*, vol. 37, no. 3, p. 355, Apr 2011. [Online]. Available: https://doi.org/10.1007/s10844-011-0157-4

[142] L. Nováková and O. Štepánková, “Multidimensional clusters in RadViz,” in *Proceedings of the 6th WSEAS International Conference on Simulation, Modelling and Optimization*, 2006, pp. 470–475.

[143] R. Ong, F. Pinelli, R. Trasarti, M. Nanni, C. Renso, S. Rinzivillo, and F. Giannotti, “Traffic Jams Detection Using Flock Mining,” in *Machine Learning and Knowledge Discovery in Databases*, D. Gunopulos, T. Hofmann, D. Malerba, and M. Vazirgiannis, Eds. Springer Berlin Heidelberg, 2011, pp. 650–653.

[144] Pak Chung Wong and J. Thomas, “Visual Analytics,” *IEEE Computer Graphics and Applications*, vol. 24, no. 5, pp. 20–21, Sep. 2004.

[145] A. V. Pandey, J. Krause, C. Felix, J. Boy, and E. Bertini, “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*, ser. CHI ’16. Association for Computing Machinery, 2016. [Online]. Available: https://doi.org/10.1145/2858036.2858155

[146] D. K. Park, Y. S. Jeon, and C. S. Won, “Efficient use of local edge histogram descriptor,” in *Proceedings of the 2000 ACM workshops on Multimedia*. ACM, 2000, pp. 51–54.

[147] F. V. Paulovich, M. C. F. Oliveira, and R. Minghim, “The Projection Explorer: A Flexible Tool for Projection-based Multidimensional Visualization,” in *XX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2007)*, 2007, pp. 27–36.

[148] C. Perin, R. Vuillemot, and J.-D. Fekete, “SoccerStories: A Kick-off for Visual Soccer Analysis,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2506–2515, Dec 2013.

[149] J. Petit, “Experiments on the Minimum Linear Arrangement Problem,” *J. Exp. Algorithmics*, vol. 8, Dec. 2003. [Online]. Available: http://doi.acm.org/10.1145/996546.996554

[150] K. Pfeuffer, J. Alexander, M. K. Chong, and H. Gellersen, “Gaze-touch: Combining Gaze with Multi-touch for Interaction on the Same Surface,” in *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST ’14. ACM, 2014, pp. 509–518. [Online]. Available: http://doi.acm.org/10.1145/2642918.2647397

[151] C. Pindat, E. Pietriga, O. Chapuis, and C. Puech, “Drilling into Complex 3D Models with Gimlenses,” in *Proceedings of the 19th ACM Symposium on Virtual Reality Software and Technology*, 2013.

[152] T. Polk, J. Yang, Y. Hu, and Y. Zhao, “TenniVis: Visualization for Tennis Match Analysis,” *Visualization and Computer Graphics, IEEE Transactions on*, vol. 20, no. 12, pp. 2339–2348, Dec 2014.

[153] A. Poole and L. J. Ball, “Eye Tracking in Human-Computer Interaction and Usability Research: Current Status and Future Prospects,” in *C. Ghaoui (Ed.): Encyclopedia of Human-Computer Interaction.*, 2005.

[154] A. Ram and L. Hunter, “The use of explicit goals for knowledge to guide inference and learning,” *Applied Intelligence*, vol. 2, no. 1, pp. 47–73, Jul 1992. [Online]. Available: https://doi.org/10.1007/BF00058575

[155] R. Rao and S. K. Card, “The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus + Context Visualization for Tabular Information,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI ’94. ACM, 1994, pp. 318–322. [Online]. Available: http://doi.acm.org/10.1145/191666.191776

[156] M. G. I. Rathod and M. D. A. Nikam, “Review on Event Retrieval in Soccer Video,” *International Journal of Computer Science & Information Technologies*, vol. 5, no. 4, 2014.

[157] M. Rubio-Sánchez, L. Raya, F. Díaz, and A. Sanchez, "A comparative study between RadViz and Star Coordinates," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, pp. 619–628, 2016.

[158] D. Sacha, A. Stoffel, F. Stoffel, B. C. Kwon, G. Ellis, and D. A. Keim, "Knowledge Generation Model for Visual Analytics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 1604–1613, Dec 2014.

[159] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., 1986.

[160] K. Samp and S. Decker, "Supporting menu design with radial layouts," in *Proceedings of the International Conference on Advanced Visual Interfaces*. ACM, 2010, pp. 155–162.

[161] A. Sarikaya and M. Gleicher, "Scatterplots: Tasks, Data, and Designs," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 402–412, 2018.

[162] R. Scheepens, N. Willems, H. van de Wetering, and J. J. van Wijk, "Interactive visualization of multivariate trajectory data with density maps," in *2011 IEEE Pacific Visualization Symposium*, March 2011, pp. 147–154.

[163] M. Scherer, J. Bernard, and T. Schreck, "Retrieval and Exploratory Search in Multivariate Research Data Repositories Using Regressional Features," in *Proc. of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries*, 2011, pp. 363–372.

[164] M. Scherer, T. von Landesberger, and T. Schreck, "A Benchmark for Content-Based Retrieval in Bivariate Data Collections," in *Proc. Int. Conference on Theory and Practice of Digital Libraries*, 2012.

[165] T. Schreck, T. von Landesberger, and S. Bremm, "Techniques for Precision-Based Visual Analysis of Projected Data," *Information Visualization*, vol. 9, no. 3, pp. 181–193, 2010.

[166] H.-J. Schulz, M. Streit, T. May, and C. Tominski, "Towards a Characterization of Guidance in Visualization," in *IEEE Conference on Information Visualization (InfoVis) - Poster Paper*, 2013.

[167] F. Sebastiani, "Machine Learning in Automated Text Categorization," *ACM Comput. Surv.*, vol. 34, no. 1, pp. 1–47, 2002.

[168] M. Sedlmair, A. Tatu, T. Munzner, and M. Tory, "A taxonomy of visual cluster separation factors," *Computer Graphics Forum (Proc. EuroVis 2012)*, vol. 31(3), pp. 1335–1344, 2012.

[169] J. Z. Self, M. Dowling, J. Wenskovitch, I. Crandell, M. Wang, L. House, S. Leman, and C. North, "Observation-Level and Parametric Interaction for High-Dimensional Data Analysis," *ACM Trans. Interact. Intell. Syst.*, vol. 8, no. 2, pp. 15:1–15:36, Jun. 2018.

[170] S. Shang, R. Ding, B. Yuan, K. Xie, K. Zheng, and P. Kalnis, "User Oriented Trajectory Search for Trip Recommendation," in *Proceedings of the 15th International Conference on Extending Database Technology*, ser. EDBT '12. ACM, 2012, pp. 156–167.

[171] L. Shao, A. Mahajan, T. Schreck, and D. J. Lehmann, "Interactive Regression Lens for Exploring Scatter Plots," *Computer Graphics Forum*, vol. 36, no. 3, pp. 157–166, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13176

[172] L. Shao, *Skizzen-basierte Suche für bivariate Daten mittels Bild-basierten Deskriptoren*. Konstanz, Univ., Masterarb.„ 2003.

[173] L. Shao, M. Behrisch, T. Schreck, T. von Landesberger, M. Scherer, S. Bremm, and D. A. Keim, "Guided Sketching for Visual Search and Exploration in Large Scatter Plot Spaces," in *EuroVis Workshop on Visual Analytics (EuroVA)*, M. Pohl and J. Roberts, Eds. The Eurographics Association, 2014.

[174] L. Shao, M. Glatz, E. Gergely, M. Müller, D. Munter, S. Papst, and T. Schreck, "Extending Document Exploration with Image Retrieval: Concept and First Results," in *EuroVis 2018 - Poster Paper*, A. Puig and R. Raidou, Eds. The Eurographics Association, 2018.

[175] L. Shao, S. Kloiber, M. Chegini, K. Andrews, T. Schreck, and D. J. Lehmann, "Integrated Projection Paths for the Discovery of Patterns in High-Dimensional Data and its Embedded Subspaces," *Submitted to IEEE Transactions on Visualization and Computer Graphics*, 2020.

[176] L. Shao, S. Mittelstädt, R. Goldblatt, I. Omer, P. Bak, and T. Schreck, "Analysis and Comparison of Feature-Based Patterns in Urban Street Networks," in *Computer Vision, Imaging and Computer Graphics Theory and Applications*. Springer International Publishing, 2017, pp. 287–309.

[177] L. Shao, D. Sacha, B. Neldner, M. Stein, and T. Schreck, "Visual-Interactive Search for Soccer Trajectories to Identify Interesting Game Situations," *Electronic Imaging Conference on Visualization and Data Analysis*, vol. 2016, no. 1, pp. 1–10, 2016. [Online]. Available: https://www.ingentaconnect.com/content/ist/ei/2016/00002016/00000001/art00033

[178] L. Shao, T. Schleicher, M. Behrisch, T. Schreck, I. Sipiran, and D. A. Keim, "Guiding the exploration of scatter plot data using motif-based interest measures," *Journal*

of Visual Languages & Computing, vol. 36, pp. 1 – 12, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1045926X16300441

[179] L. Shao, T. Schleicher, and T. Schreck, "Query by Visual Words: Visual Search for Scatter Plot Visualizations," in *Proceedings of the Eurographics / IEEE VGTC Conference on Visualization (EuroVis) - Poster Paper*, ser. EuroVis '16. Eurographics Association, 2016, pp. 41–43. [Online]. Available: https://doi.org/10.2312/eurp.20161137

[180] L. Shao, N. Silva, E. Eggeling, and T. Schreck, "Visual Exploration of Large Scatter Plot Matrices by Pattern Recommendation Based on Eye Tracking," in *Proceedings of the 2017 ACM Workshop on Exploratory Search and Interactive Data Analytics*, ser. ESIDA '17. ACM, 2017, pp. 9–16. [Online]. Available: http://doi.acm.org/10.1145/3038462.3038463

[181] J. Sharko, G. Grinstein, and K. A. Marx, "Vectorized Radviz and Its Application to Multiple Cluster Datasets," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, 2008.

[182] D. B. Shepard, A. R. Kuhns, M. J. Dreslik, and C. A. Phillips, "Roads as barriers to animal movement in fragmented landscapes," *Animal Conservation*, vol. 11, no. 4, pp. 288–296, 2008.

[183] B. Shneiderman, "The eyes have it: a task by data type taxonomy for information visualizations," in *Proceedings 1996 IEEE Symposium on Visual Languages*, Sep. 1996, pp. 336–343.

[184] N. Silva, T. Schreck, E. Veas, V. Sabol, E. Eggeling, and D. W. Fellner, "Leveraging Eye-gaze and Time-series Features to Predict User Interests and Build a Recommendation Model for Visual Analysis," in *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*, ser. ETRA '18. ACM, 2018, pp. 13:1–13:9. [Online]. Available: http://doi.acm.org/10.1145/3204493.3204546

[185] N. Silva, L. Shao, T. Schreck, E. Eggeling, and D. W. Fellner, "Visual Exploration of Hierarchical Data Using Degree-of-Interest Controlled by Eye-Tracking," in *FMT*, 2016.

[186] M. Sips, B. Neubert, J. P. Lewis, and P. Hanrahan, "Selecting good views of high-dimensional data using class consistency," in *Computer Graphics Forum*, vol. 28, no. 3. Wiley Online Library, 2009, pp. 831–838.

[187] D. Spretke, H. Janetzko, F. Mansmann, P. Bak, B. Kranstauber, S. Davidson, and M. Mueller, "Exploration through Enrichment: A Visual Analytics Approach for Animal Movement,"

in *Proceedings of the 19th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ser. GIS '11. ACM, 2011, pp. 421–424.

[188] A. Srinivasan and J. Stasko, "Orko: Facilitating Multimodal Interaction for Visual Exploration and Analysis of Networks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 511–521, Jan 2018.

[189] J. Stahnke, M. Dörk, B. Müller, and A. Thom, "Probing Projections: Interaction Techniques for Interpreting Arrangements and Errors of Dimensionality Reductions," *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 1, pp. 629–638, 2016.

[190] C. Stolte, D. Tang, and P. Hanrahan, "Polaris: A System for Query, Analysis, and Visualization of Multidimensional Databases," *Commun. ACM*, vol. 51, no. 11, pp. 75–84, Nov. 2008. [Online]. Available: http://doi.acm.org/10.1145/1400214.1400234

[191] G.-D. Sun, Y.-C. Wu, R.-H. Liang, and S.-X. Liu, "A Survey of Visual Analytics Techniques and Applications: State-of-the-Art Research and Future Challenges," *Journal of Computer Science and Technology*, vol. 28, no. 5, pp. 852–867, Sep 2013. [Online]. Available: https://doi.org/10.1007/s11390-013-1383-8

[192] D. F. Swayne and D. Cook, "Xgobi: A Dynamic Graphics Program Implemented in X With a Link to S," in *Computing Science and Statistics*, C. Page and R. LePage, Eds. Springer New York, 1992, pp. 544–547.

[193] J. Talbot, J. Gerth, and P. Hanrahan, "Arc length-based aspect ratio selection." *IEEE transactions on visualization and computer graphics*, vol. 17, no. 12, pp. 2276–82, Dec. 2011.

[194] A. Tatu, G. Albuquerque, M. Eisemann, P. Bak, H. Theisel, M. Magnor, and D. A. Keim, "Automated Analytical Methods to Support Visual Exploration of High-Dimensional Data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 5, pp. 584–597, May 2011.

[195] A. Tatu, G. Albuquerque, M. Eisemann, J. Schneidewind, H. Theisel, M. Magnork, and D. A. Keim, "Combining automated analysis and visualization techniques for effective exploration of high-dimensional data," in *2009 IEEE Symposium on Visual Analytics Science and Technology*, Oct 2009, pp. 59–66.

[196] A. Tatu, F. Maaß, I. Färber, E. Bertini, T. Schreck, T. Seidl, and D. A. Keim, "Subspace search and visualization to make sense of alternative clusterings in high-dimensional data," in *IEEE Conference on Visual Analytics Science and Technology*, 2012, pp. 63–72.

[197] C. Tominski, S. Gladisch, U. Kister, R. Dachselt, and H. Schumann, "Interactive Lenses for Visualization: An Extended Survey," *Computer Graphics Forum*, vol. 36, no. 6, pp. 173–200, 2017.

[198] C. Tominski, H. Schumann, G. Andrienko, and N. Andrienko, "Stacking-Based Visualization of Trajectory Attribute Data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2565–2574, Dec 2012.

[199] C. Tominski, S. Gladisch, U. Kister, R. Dachselt, and H. Schumann, "A Survey on Interactive Lenses in Visualization," in *EuroVis - STARs*, R. Borgo, R. Maciejewski, and I. Viola, Eds. The Eurographics Association, 2014.

[200] E. R. Tufte, *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, CT, 1983.

[201] J. W. Tukey, *Exploratory Data Analysis*. Addison-Wesley, 1977.

[202] P. van der Corput and J. J. van Wijk, "Exploring Items and Features with IF, FI-Tables," *Computer Graphics Forum*, vol. 35, no. 3, 2016.

[203] J. J. van Wijk, "Views on Visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 421–432, July 2006.

[204] S. Voida, M. Tobiasz, J. Stromer, P. Isenberg, and S. Carpendale, "Getting Practical with Interactive Tabletop Displays: Designing for Dense Data, "Fat Fingers," Diverse Interactions, and Face-to-face Collaboration," in *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ser. ITS '09. ACM, 2009, pp. 109–116. [Online]. Available: http://doi.acm.org/10.1145/1731903.1731926

[205] T. von Landesberger, T. Schreck, D. W. Fellner, and J. Kohlhammer, *Visual Search and Analysis in Complex Information Spaces—Approaches and Research Challenges*. Springer London, 2012, pp. 45–67. [Online]. Available: https://doi.org/10.1007/978-1-4471-2804-5_4

[206] T. von Landesberger, S. Bremm, J. Bernard, and T. Schreck, "Smart query definition for content-based search in large sets of graphs," in *Proc. Int. Symposium on Visual Analytics Science and Technology*. Eurographics Association, 2010, pp. 7–12.

[207] A. K. Wagner, S. B. Soumerai, F. Zhang, and D. Ross-Degnan, "Segmented regression analysis of interrupted time series studies in medication use research," *Journal of Clinical Pharmacy and Therapeutics*, vol. 27, no. 4, pp. 299–309, 2002.

[208] J. Walny, B. Lee, P. Johns, N. Henry Riche, and S. Carpendale, "Understanding Pen and Touch Interaction for Data Exploration on Interactive Whiteboards," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2779–2788, Dec 2012.

[209] B. Wang, P. Ruchikachorn, and K. Mueller, "SketchPadN-D: WYDIWYG Sculpting and Editing in High-Dimensional Space," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 19, no. 12, pp. 2060–2069, Dec 2013.

[210] L. Wang, Y. Zhao, K. Mueller, and A. Kaufman, "The magic volume lens: an interactive focus+context technique for volume rendering," in *IEEE Visualization*, Oct 2005, pp. 367–374.

[211] Z. Wang, M. Lu, X. Yuan, J. Zhang, and H. van de Wetering, "Visual Traffic Jam Analysis Based on Trajectory Data," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 19, no. 12, pp. 2159–2168, 2013.

[212] M. O. Ward, "Xmdvtool: integrating multiple methods for visualizing multivariate data," in *Proceedings Visualization '94*, Oct 1994, pp. 326–333.

[213] M. O. Ward and J. Yang, "Interaction Spaces in Data and Information Visualization," in *Proceedings of the 6th Joint Eurographics - IEEE TCVG Conference on Visualization*. Eurographics Association, 2004, pp. 137–146.

[214] M. O. Ward, G. Grinstein, and D. A. Keim, *Interactive Data Visualization: Foundations, Techniques, and Applications*. A. K. Peters, Ltd., 2010.

[215] M. Wattenberg, "Sketching a Graph to Query a Time-series Database," in *CHI '01 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2001, pp. 381–382. [Online]. Available: http://doi.acm.org/10.1145/634067.634292

[216] L. Wilkinson, A. Anand, and R. Grossman, "Graph-Theoretic Scagnostics," in *Proceedings of the 2005 IEEE Symposium on Information Visualization*, 2005.

[217] L. Wilkinson, A. Anand, and R. Grossman, "High-Dimensional Visual Analytics: Interactive Exploration Guided by Pairwise Views of Point Distributions," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 6, pp. 1363–1372, Nov 2006.

[218] N. Willems, H. van de Wetering, and J. J. van Wijk, "Visualization of Vessel Movements," in *Proceedings of the 11th Eurographics / IEEE - VGTC Conference on Visualization*, 2009, pp. 959–966.

[219] W. Willett, Q. Lan, and P. Isenberg, "Eliciting Multi-touch Selection Gestures for Interactive Data Graphics," in *Short-Paper Proceedings of the European Conference on Visualization (EuroVis)*. Eurographics, 2014.

[220] C. S. Won, D. K. Park, and S.-J. Park, "Efficient use of MPEG-7 edge histogram descriptor," *Etri Journal*, vol. 24, no. 1, 2002.

[221] Z. Xiong, R. Radhakrishnan, A. Divakaran, and T. S. Huang, "Audio Events Detection Based Highlights Extraction from Baseball, Golf and Soccer Games in a Unified Framework," in *Acoustics, Speech, and Signal Processing. IEEE International Conference on*, vol. 5. IEEE, 2003, pp. V–632.

[222] K. Zhao, M. O. Ward, E. A. Rundensteiner, and H. N. Higgins, "LoVis: Local Pattern Visualization for Model Refinement," in *Proceedings of the 16th Eurographics Conference on Visualization*, ser. EuroVis '14. Eurographics Association, 2014, p. 331–340.

[223] G. Zhu, Q. Huang, C. Xu, Y. Rui, S. Jiang, W. Gao, and H. Yao, "Trajectory Based Event Tactics Analysis in Broadcast Sports Video," in *Proceedings of the 15th International Conference on Multimedia*. ACM, 2007, pp. 58–67.

[224] G. Zhu, C. Xu, Q. Huang, W. Gao, and L. Xing, "Player action recognition in broadcast tennis video with applications to semantic analysis of sports game," in *Proceedings of the 14th annual ACM international conference on Multimedia*. ACM, 2006, pp. 431–440.