Interactive Visualization Lenses

Natural Magic Lens Interaction for Graph Visualization

Ulrike Kister

June 2018

Technische Universität Dresden

Fakultät Informatik
Institut für Software- und Multimediatechnik
Interactive Media Lab Dresden

Interactive Visualization Lenses

Natural Magic Lens Interaction for Graph Visualization

Dissertation

zur Erlangungen des akademischen Grades
Doktoringenieur (Dr.-Ing.)

Ulrike Kister
Master of Science in Computational Visualisitics
geboren am 18.03.1987 in Brandenburg (Havel)

Erstgutachter und Betreuer	Prof. Dr.-Ing. Raimund Dachselt Professur für Multimedia-Technologie Technische Universität Dresden
Zweitgutachterin	Petra Isenberg, Ph.D. Chargé de Recherche – AVIZ research group Institut national de recherche en informatique et en automatique (Inria)
Kommissionsvorsitz	Prof. Dr.-Ing. Wolfgang Lehner
Fachreferent	Prof. Dr. rer. nat. habil. Gerhard Weber
Weiteres Mitglied	Prof. Dr. Markus Krötzsch

Tag der Verteidigung: 08. Mai 2018

Dresden im Juni 2018

Ulrike Kister
Interactive Visualization Lenses : Natural Magic Lens Interaction for Graph Visualization
Dissertation, June 2018
Reviewers: Prof. Dr.-Ing. Raimund Dachselt and Petra Isenberg, Ph.D.
Supervisor: Prof. Dr.-Ing. Raimund Dachselt

Technische Universität Dresden
Interactive Media Lab Dresden
Institute of Software and Multimedia Technology
Faculty of Computer Science
01062 Dresden

Abstract

Information visualization is an important research field concerned with making sense and inferring knowledge from data collections. Graph visualizations are specific techniques for data representation relevant in diverse application domains among them biology, software-engineering, and business finance. These data visualizations benefit from the display space provided by novel interactive large display environments. However, these environments also cause new challenges and result in new requirements regarding the need for interaction beyond the desktop and according redesign of analysis tools. This thesis focuses on interactive magic lenses, specialized locally applied tools that temporarily manipulate the visualization. These may include magnification of focus regions but also more graph-specific functions such as pulling in neighboring nodes or locally reducing edge clutter. Up to now, these lenses have mostly been used as single-user, single-purpose tools operated by mouse and keyboard.

This dissertation presents the extension of magic lenses both in terms of function as well as interaction for large vertical displays. In particular, this thesis contributes several natural interaction designs with magic lenses for the exploration of graph data in node-link visualizations using diverse interaction modalities. This development incorporates flexible switches between lens functions, adjustment of individual lens properties and function parameters, as well as the combination of lenses. It proposes interaction techniques for fluent multi-touch manipulation of lenses, controlling lenses using mobile devices in front of large displays, and a novel concept of body-controlled magic lenses. Functional extensions in addition to these interaction techniques convert the lenses to user-configurable, personal territories with use of alternative interaction styles. To create the foundation for this extension, the dissertation incorporates a comprehensive design space of magic lenses, their function, parameters, and interactions. Additionally, it provides a discussion on increased embodiment in tool and controller design, contributing insights into user position and movement in front of large vertical displays as a result of empirical investigations and evaluations.

Zusammenfassung

Informationsvisualisierung ist ein wichtiges Forschungsfeld, das das Analysieren von Daten unterstützt. Graph-Visualisierungen sind dabei eine spezielle Variante der Datenrepräsentation, deren Nutzen in vielerlei Anwendungsfällen zum Einsatz kommt, u.a. in der Biologie, Softwareentwicklung und Finanzwirtschaft. Diese Datendarstellungen profitieren besonders von großen Displays in neuen Displayumgebungen. Jedoch bringen diese Umgebungen auch neue Herausforderungen mit sich und stellen Anforderungen an Nutzerschnittstellen jenseits der traditionellen Ansätze, die dadurch auch Anpassungen von Analysewerkzeugen erfordern. Diese Dissertation befasst sich mit interaktiven „Magischen Linsen", spezielle lokal-angewandte Werkzeuge, die temporär die Visualisierung zur Analyse manipulieren. Dabei existieren zum Beispiel Vergrößerungslinsen, aber auch Graph-spezifische Manipulationen, wie das Anziehen von Nachbarknoten oder das Reduzieren von Kantenüberlappungen im lokalen Bereich. Bisher wurden diese Linsen vor allem als Werkzeug für einzelne Nutzer mit sehr spezialisiertem Effekt eingesetzt und per Maus und Tastatur bedient.

Die vorliegende Doktorarbeit präsentiert die Erweiterung dieser magischen Linsen, sowohl in Bezug auf die Funktionalität als auch für die Interaktion an großen, vertikalen Displays. Insbesondere trägt diese Dissertation dazu bei, die Exploration von Graphen mit magischen Linsen durch natürliche Interaktion mit unterschiedlichen Modalitäten zu unterstützen. Dabei werden flexible Änderungen der Linsenfunktion, Anpassungen von individuellen Linseneigenschaften und Funktionsparametern, sowie die Kombination unterschiedlicher Linsen ermöglicht. Es werden Interaktionstechniken für die natürliche Manipulation der Linsen durch Multitouch-Interaktion, sowie das Kontrollieren von Linsen durch Mobilgeräte vor einer Displaywand vorgestellt. Außerdem wurde ein neuartiges Konzept körpergesteuerter magischer Linsen entwickelt. Funktionale Erweiterungen in Kombination mit diesen Interaktionskonzepten machen die Linse zu einem vom Nutzer einstellbaren, persönlichen Arbeitsbereich, der zudem alternative Interaktionsstile erlaubt. Als Grundlage für diese Erweiterungen stellt die Dissertation eine umfangreiche analytische Kategorisierung bisheriger Forschungsarbeiten zu magischen Linsen vor, in der Funktionen, Parameter und Interaktion mit Linsen eingeordnet werden. Zusätzlich macht die Arbeit Vor- und Nachteile körpernaher Interaktion für Werkzeuge bzw. ihre Steuerung zum Thema und diskutiert dabei Nutzerposition und -bewegung an großen Displaywänden belegt durch empirische Nutzerstudien.

Acknowledgement

This thesis would not have been possible without the guidance, support, and friendship of many amazing people around me. I am sincerely grateful to everybody who encouraged and helped me along the way.

First and foremost, my uttermost thanks go out to my supervisor *Raimund Dachselt* for his guidance, advisement, and support throughout this entire endeavor. Starting with his encouragement for me to pursue a PhD and his inspiring research ambitions, I'm especially grateful for his continuous confidence. Much of what I've accomplished in the last few years can be directly traced back to his support and I very much appreciate his general concern and interest in my personal progress. I would also like to take this opportunity to thank *Petra Isenberg* for taking the time and kindly agreeing to review my thesis as well as for her inspiring work in our research fields.

Beside his input on design, this LaTeX template, and his gesture graphics, I am especially grateful to *Ricardo Langner* for his always constructive and worthwhile discussions, his open mind and ear for any of my problems, and his contagious enthusiasm for information visualization including our conversations of whether and why it is actually needed. I am also very thankful to *Wolfgang Büschel* for his support and witty remarks in so many situations as well as his ability for critical thinking that inspired many discussions and reflections on all of our work. I am equally grateful for his love for movies which helped get us out of the lab every now and then. I further want to thank *Annett Mitschick* for her ever-friendly advise and productive attitude, *Anke Lehmann* for her input on our BodyLenses work and the discussions that helped restructuring it, as well as *Ulrich von Zadow* for his support and maintenance of libavg which became the basis for many of the prototypes in this thesis.

In addition, I would like to thank *Patrick Reipschläger* for his mind for good, modular, and stable programming and his dedication to our joint work especially, but not limited, to our prototypes for MultiLens and BodyLenses. I am also very grateful to *Konstantin Klamka* and *Tom Horak* for our always joyful collaborations regarding either GraSp or anything graph-related (SAP, VDT, GEMS 2.0, ...), respectively. I thank both of you for the often quite necessary coffee breaks and the resulting inspiring discussions. Special thanks go to *Udo Wähner* for his all-in-all friendly and helpful attitude when anything was to build or set up and to *Ramona Behling* for solving the many not so little organizational issues of contracts, travel grants, and student requests on top of her willingness to listen and give advice to any of my personal problems and dreams.

I very much appreciate all the supportive colleagues I had the opportunity to meet during my time at the Interactive Media Lab Dresden: *Sophie Stellmach*, *Axel Berndt*, *Martin Spindler*, *André Viergutz*, and *Fabrice Matulic*. I further especially owe my thanks to all of my co-authors but am particularly grateful to *Christian Tominski*, *Heidrun Schumann*, and *Stefan Gladisch* from Rostock for their input on visualization aspects in general, their expert opinions and constructive criticisms, our shared adventure of ruling magic lenses, and the friendly beers we enjoyed together at EuroVis both in Wales and Barcelona.

Last but not least, I want to thank *my family* for their unconditional support, encouragement, and love. Especially, I am very grateful to my mum for taking the time and, with no interest in programming herself, helping her eleven year old girl go through the programming manual for BASIC – yeah, why not? I also very much appreciate and thank my dad and my brother who, despite naturally annoying me to no end, always believed in me and encouraged me to go my own way. Finally, my deepest gratitude goes to *Mathias Frisch* for sticking with me, despite our 320 km distance, and always providing the necessary tea/coffee/cake/love to get me through this, especially in the last couple of months. Thank you.

Publications

Parts of this research were previously published in the following publications. Where applicable, reused material is indicated at the beginning of each chapter.

Journal and Full Paper Publications (Peer-Reviewed)

- **Ulrike Kister**, Konstantin Klamka, Christian Tominski, Raimund Dachselt. 2017. GraSp: Combining Spatialy-aware Mobile Devices and a Display Wall for Graph Visualization and Interaction. In *Computer Graphics Forum (CGF), Vol. 36, No. 3*. 503-514, ISSN: 1467-8659.
- Christian Tominski, Stefan Gladisch, **Ulrike Kister**, Raimund Dachselt, and Heidrun Schumann. 2017. Interactive Lenses for Visualization: An Extended Survey. In *Computer Graphics Forum (CGF), Vol. 36, No. 6*. 173–200, ISSN: 1467-8659.
- **Ulrike Kister**, Patrick Reipschläger, and Raimund Dachselt. 2016. MultiLens: Fluent Interaction with Multi-Functional Multi-Touch Lenses for Information Visualization. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces* (ISS '16'). ACM, New York, NY, USA, 139-148.
- **Ulrike Kister**, Patrick Reipschläger, Fabrice Matulic, and Raimund Dachselt. 2015. BodyLenses: Embodied Magic Lenses and Personal Territories for Wall Displays. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces* (ITS '15). ACM, New York, NY, USA, 117-126.
- Juan David Hincapié-Ramos, Sophie Roscher, Wolfgang Büschel, **Ulrike Kister**, Raimund Dachselt, and Pourang Irani. 2014. tPad: designing transparent-display mobile interactions. In *Proceedings of the 2014 conference on Designing interactive systems* (DIS '14). ACM, New York, NY, USA, 161-170.
- Juan David Hincapié-Ramos, Sophie Roscher, Wolfgang Büschel, **Ulrike Kister**, Raimund Dachselt, and Pourang Irani. 2014. cAR: Contact Augmented Reality with Transparent-Display Mobile Devices. In *Proceedings of The International Symposium on Pervasive Displays* (PerDis '14). ACM, New York, NY, USA, 80-86.
- Wolfgang Büschel, **Ulrike Kister**, Mathias Frisch, and Raimund Dachselt. 2014. T4 - transparent and translucent tangibles on tabletops. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces* (AVI '14). ACM, New York, NY, USA, 81-88.

State-of-the-Art Report (Peer-Reviewed)

- Christian Tominski, Stefan Gladisch, **Ulrike Kister**, Raimund Dachselt, and Heidrun Schumann. 2014. A Survey on Interactive Lenses in Visualization. In *Eurographics Conference on Visualization (EuroVis) State-of-the-Art Reports*. Eurographics Association, Swansea, Wales, UK. The Eurographics Association, 2014.

Poster and Demo Publications (Peer-Reviewed)

- Tom Horak, **Ulrike Kister**, and Raimund Dachselt. 2017. Improving Value Driver Trees to Enhance Business Data Analysis. In *Poster Program of the 2017 IEEE Conference on Information Visualization* (InfoVis '17). IEEE, Phoenix, USA.
- Tom Horak, **Ulrike Kister**, and Raimund Dachselt. 2016. Presenting Business Data: Challenges during Board Meetings in Multi-Display Environments. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces* (ISS '16'). ACM, New York, NY, USA, 319-324.
- **Ulrike Kister**, Konstantin Klamka, and Raimund Dachselt. 2016. Supporting Graph Exploration Tasks on Display Walls using Spatially-aware Mobile Devices. In *Poster Program of the 2016 IEEE Conference on Information Visualization* (InfoVis '16). IEEE, Baltimore, MD, USA.
- Ricardo Langner, **Ulrike Kister**, Christin Engel, Tom Horak, and Raimund Dachselt. 2015. Towards a Fluid Interaction Concept Bringing Star Plots to Interactive Displays. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces* (ITS '15). ACM, New York, NY, USA, 349-354.
- Stefan Gladisch, **Ulrike Kister**, Christian Tominski, Raimund Dachselt, and Heidrun Schumann. 2015. Mapping Tasks to Interactions for Graph Exploration and Editing. In *Poster Program of the 2015 IEEE Conference on Information Visualization* (InfoVis). InfoVis'15, Chicago, USA. IEEE, 2015.
- **Ulrike Kister**, Patrick Reipschläger, and Raimund Dachselt. 2014. Multi-Touch Manipulation of Magic Lenses for Information Visualization. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces* (ITS '14). ACM, New York, NY, USA, 431-434.
- Robert Morawa, Tom Horak, **Ulrike Kister**, Annett Mitschick, and Raimund Dachselt. 2014. Combining Timeline and Graph Visualization. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces* (ITS '14). ACM, New York, NY, USA, 345-350.

Technical Reports

- Stefan Gladisch, **Ulrike Kister**, Christian Tominski, Raimund Dachselt, and Heidrun Schumann. 2015. Mapping Tasks to Interactions for Graph Exploration and Editing on Interactive Surfaces. In *arXiv:1504.07844* (Technical Report), 21 pages, 2015.

Workshop Contributions and Doctoral Symposium

- Tom Horak, **Ulrike Kister**, Konstantin Klamka, Ricardo Langner, Raimund Dachselt. 2016. Logging in Visualizations: Challenges of Interaction Techniques Beyond Mouse and Keyboard. In *Workshop on Logging Interactive Visualizations & Visualizing Interaction Logs (LIVVIL)*. (VIS '16), Baltimore, MD, USA.
- **Ulrike Kister**. 2015. Supporting Interactive Graph Analysis Using Adjustable Magic Lenses. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces* (ITS '15). ACM, New York, NY, USA, 469-474.
- Mathias Frisch, **Ulrike Kister**, Wolfgang Büschel, and Raimund Dachselt. 2013. Translucent Tangibles on Tabletops: Exploring the Design Space. In *CHI 2013 Workshop on Blended Interaction - Envisioning Future Collaborative Interactive Spaces.* (CHI '13). Paris, France.

Of the 24 student theses that I co-supervised during the time of this PhD work, content of the following works contributed in part to this dissertation:

- Marc Satkowski. *Presentation and Interaction with Multiple Coordinated Views at Wall-sized Displays* (Student research project). 2017.
- Norman Lorenz. *Magic Lenses on Display Walls: Control and Manipulation with Mobile Devices* (Diploma thesis). 2016/17.
- Marc Satkowski. *Analysis of Multivariate Data on Maps Using BodyLenses* (Student research project). 2016/17.
- Stephanie Sara Groß. *Evaluation of a Multi-touch Menu for Magic Lenses in Comparison to Classic Menus* (Bachelor's thesis). 2016.
- Konstantin Klamka. *Using Mobile Devices as Lenses for Graph Exploration and Manipulation* (Diploma thesis). 2015/16.
- Maximilian Gräf. *Collaborative Mind-Map Creation Using Personal Lenses on Interactive Display Walls* (Bachelor's thesis). 2015.
- Christoph Plagge. *Combining Physical Navigation & Touch Interaction for Manipulating Graph Lenses on Interactive Display Walls* (Bachelor's thesis). 2015.
- Martin Wegner. *Interaction with Tangible Graph Lenses on Interactive Tabletops* (Diploma thesis). 2013/14.
- Patrick Reipschläger. *Combining Magic Lenses for Graph Exploration* (Master's thesis). 2013/14.

Contents

1 Introduction

The increasing digitalization process in the modern world results in large amounts of data both in context of individual users, e.g., personal health data, jogging routes, or social networks, as well as in professional contexts, e.g., product lifecycle management or performance simulations. Graph data are an essential data structure relevant in a variety of domains including but not limited to biology, software-engineering, or business finance. Handling these data and even more so making sense of them to infer knowledge requires data analysis [Fek+08; VPF06; Rob+14]. While some of these analytical processes can and will be handled through automatic analysis or machine learning, visual analysis by exploring visualization and thereby actively exploring data is still an essential part of seeing behind individual numbers and grasping their meaning. As such, the visualized data is perceived and consumed by individual users taking advantage of their perception, their visual capacity, and individual insights that allow us to see patterns and connections that are otherwise undetectable by an automatic system [Fek+08; Spe14]. While some of these data sets may initially seem un-graspable due to their size and complexity, visual analysis can be useful to get an understanding and feel for the data.

Visualization, Graphs, and Magic Lenses

Visualization as a tool for data analysis and visual analytics aids the process of understanding by providing a visual representation of the data and thereby enabling an abstraction from individual data recordings to seeing the data as a whole [Fek+08]. Our visual system grasps the highest bars in a bar chart more easily than comparing a table of numbers. It is also able to quickly recognize similar properties like color [Ber83]. Depending on the data types and the interest to the user, various information visualization techniques have been introduced to present data. Some of these are well-known and used in every-day products such as bar charts and scatter plots, while others have been applied for more advanced data analysis and more complex representation of data, e.g., star/radar plots, parallel coordinate plots, or bubble charts.

The term *graph* refers to relational data that can be represented as a set of nodes and edges [Die05] including their associated attributes and properties. This data can be visually represented in graph visualizations, specifically node-link diagrams (see Figure 1.1), which are a specific technique of information visualization. While the solutions provided and investigated in this thesis may be equally applicable to other

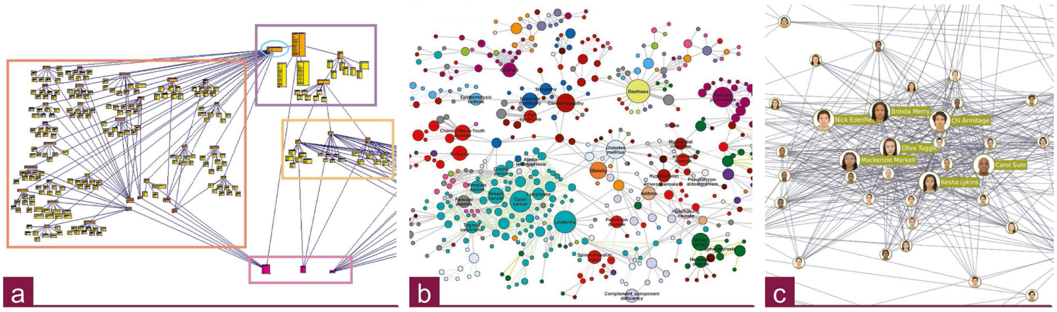


Fig. 1.1.: Node-link visualizations present graph data as nodes and edges to visualize, for example, cancer staging decisions (a, [Cyp+17]), human diseases (b, [Goh+07]), or social networks (c).

visualization techniques and information representations, graph visualization has been selected as a main focus of this work for their diverse application scenarios.

Aiming at supporting the exploration, analysis, and understanding of graph data sets in visualization requires knowledge concerning the necessary tasks and interactions which are crucial for a successful analysis. To understand and cover these exploration tasks, research has already focused on extracting and categorizing goals and interaction steps used to investigate data [Yi+07; BM13] and graphs specifically [Lee+06; Wyb+14; PPS14]. For instance, this includes tasks such as filtering nodes of specific type, degree, or attribute values and visually removing them or finding the adjacent nodes of an interesting cluster. While these basic tasks are understood, the questions remains of *how* these tasks can be supported best by means of user interface design for an improved exploration process. This requires tools that flexibly adapt to the current goals of the user and cover support for a wide range, if not all, of the necessary interaction tasks.

Magic lenses are such a tool equipped for data analysis in visualization. They have been introduced as general user interface tools by Bier et al. [Bie+93] in 1993. Originating from the idea of a physical lens moved above an area of interest to increase visibility of details, the concept has been extended in the digital world. Magic lenses (in the following also 'lenses') locally manipulate the view to enrich or alter it, showing further details or removing selected parts of the content (see Figure 1.2). Lenses have been shown to support a wide range of exploration tasks [Tom+17]. Many researchers have focused on developing appropriate lens functions for specific problems in individual data sets and selected interaction tasks (e.g., [WCG03; EDF11; Hei+11]). However, in prior work the lens has often been applied as a single-purpose tool. Even though the same general principle has been used for different visualization techniques and tasks, it has rarely been applied as a flexible, manipulatable many-purpose toolbox. Furthermore, existing literature focused on the functionality of the tool and did not or rarely discussed the interactive

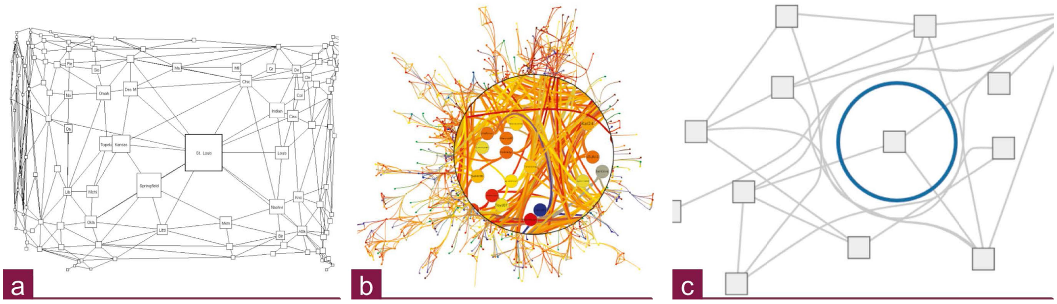


Fig. 1.2.: Local manipulations of the view support graph exploration by creating space for relevant nodes (a, fisheye distortion [SB92]), resizing a region of interest (b, magnification lens [LAM10]), or removing clutter (c, PushLens [Sch+10b]).

aspects of lens usage. As a result, there was no end-user configuration of the tool and smooth integration into an actual workflow of varying tasks and goals.

Beside these general aspects of *how* visual analysis and its appropriate tools can be supported, the entire process of how users interact with the virtual, digital world has increasingly developed in a more fast-paced work environment. Novel interactive spaces have been created introducing various new interaction modalities and situations, from 'business on the go' using mobile devices to large multiple display spaces and highly digitized meeting rooms.

Natural User Interfaces and Novel Display Environments

Large display sizes and vertical display walls have been shown to benefit interaction tasks in various use cases including sensemaking [AEN10], way-finding [NBC06], and exploratory visual analysis [Red+15]. Their size and resolution alone are key factors for improved usage and investigations into data [Raj+15]. Size is especially relevant for more complex tasks [Liu+14]. Within these novel display setups, interaction modalities beyond mouse and keyboard are required. For instance, touch interaction is used to enable a direct contact with the content on the display (see Figure 1.3). Further modalities such as pen, tangible objects moved in space, or speech input are just a few of the possibilities. These *natural user interfaces* aim to make interactions fluent [Elm+11] and more closely coupled to the user's natural abilities [Jac+08]. Additionally, the ability to move in front of the screen to examine details when close as well as getting an overview from afar increases performance as it aids the user's spatial sense, orientation, and navigation [BNB07; AN13].

Furthermore, these novel display setups empower collaboration and influence the number of users involved. Collaboration and discussion between analysts as well as discussion in inter-disciplinary groups have been shown to benefit the analysis process [CLM12]. Traditional single desktop setups however are unsuitable when multiple users gather together since only a single user interacts and others can only direct interaction or peer over another's shoulder. With the development

Fig. 1.3.: Novel display technologies support touch interaction in desktop settings (a). Multi-device environments further provide large screen space for visualization (b) for single-user and multi-user scenarios (c).

of multi-display environments and the possibility of additional personal mobile devices, multiple users can be supported enabling more active collaboration. These collaborative interactions incorporate both phases of actively discussing and working on a common problem as well as working in parallel on the same context [Ise+12; JH14; Liu+16]. When aiming at enabling data analysis processes, systems require support of both styles of interaction and scenarios.

The next generation of analysis tools have to embrace these novel display setups. Again, lenses seem to be a worthwhile tool to investigate: Beside their flexibility for tasks and data types, their interactive character and local effect that restricts the consequences of manipulations to a specific area or display space could be beneficial. Specifically in large display spaces and/or collaborative scenarios this property could enable co-located parallel work on the same context visualization without interrupting other users or influencing data in areas used by other participating users (cf. territoriality [SCI04]).

1.1 Research Questions and Goals

Focusing on the powerful tool of magic lenses, the goal of this thesis is to investigate how natural user interfaces may support and even enhance data exploration in front of a large vertical display. The work builds on the existing understanding of required interaction tasks to explore design spaces for lenses in natural user interfaces and develop concrete solutions for the application of magic lenses for graph exploration. In particular, this thesis addresses the following research questions.

How can magic lenses be transformed into a flexible tool for graph analysis which can be configured and adjusted according to fit user-specific needs?

Up to now research on magic lenses was mostly limited to lenses with a single effect (e.g., [WCG03; PA08; EDF11]). Exploration workflows however are often described as a fluid movement between diverse interaction tasks [Yi+07; PPS14]. To support these kind of workflows the existing visualization lenses require a **functional extension** to support easy adjustment and adaption to the user's current goal and requirements. This includes a switch of lens functions within a single tool, the combination of effects, and the manipulation of function-dependent parameters to create a new level of magic lenses.

How can lens interaction enable graph exploration at large display spaces?

Wall-sized displays have the benefit of increased visualization and interaction space for data exploration. However, they also add a range of challenges regarding the need for enhanced interaction modalities. This includes the support of interactions directly on the vertical display but also support for physical navigation [BNB07] in front of the display which is required to explore the entire information space. As a result, multi-modal **interaction design** with novel interaction modalities is needed to develop magic lens systems that support the wide-range of interaction tasks for graph exploration in fluid ways and to easily transition between modalities.

How can natural interactions improve the fluid and flexible manipulation and configuration of lenses in large display environments?

Natural user interfaces have the potential to make interactions more direct and support rich workflows [WW11; PD15]. To use these benefits for user-specific configuration of lens properties, effect, and individual function parameters requires adapting suitable existing interaction concepts as well as designing novel **interaction techniques**. Focus lies specifically in enabling this configuration within the users' workflows and thereby creating a flexible, adaptable tool. In particular, alternative **interaction styles** have to be adapted to the preferences, experiences, and current workflow of the users.

How can magic lenses support individual work in multi-user scenarios?

Collaboration of multiple analysts and even interdisciplinary discussions are widely considered helpful to improve quality and quantity of insights (e.g., [CLM12; Ise+12]). Due to their locally-restricted effect, magic lenses could foster parallel work in multi-user scenarios. This however imposes various requirements on interaction design, e.g., lens manipulations to be in-place per user and/or per lens. At the same time, the lenses will have to be regarded in terms of personalized tools to enable the use by individuals with diverse styles of interaction. It is one goal of this thesis to investigate the suitability of lenses as user-specific tools for multi-user scenarios requiring an understanding of user behavior for data exploration through **empirical investigations**.

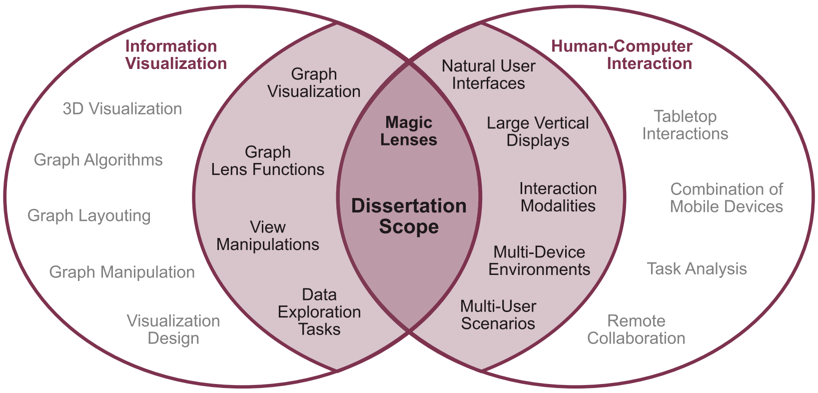


Fig. 1.4.: The scope of this thesis lies within the intersection of information visualization and human-computer interaction. Focusing on magic lenses, it considers a range of visualization and interaction aspects (highlighted areas). Though related, the principles in gray are not part of this thesis' scope.

Scope of this Research Work

Taking into account the goals of this thesis regarding both the functional extension of magic lenses as well as the application of natural user interfaces to this specific tool, the scope of this thesis lies within the intersection of human-computer interaction (HCI) and information visualization (InfoVis) (see Figure 1.4). Within the research field of information visualization which concerns the visual representation of data and the interaction with visualization for data analysis, this thesis focuses specifically on supporting exploration tasks for 2D graph visualization. Focusing on human-computer interaction, this research targets the systematic exploration of various interaction modalities and interaction designs in the field of natural user interfaces and their suitability for magic lens applications. Therefore, this thesis addresses magic lenses in regard to the following aspects:

2D Graph Visualization Graph visualizations and specifically node-link diagrams are the main example visualization technique for which this thesis explores the potential of magic lenses. In particular, this work specifically focuses on 2D data visualizations as opposed to 3D graph visualization. As a result, it will also focus exclusively on 2-dimensional lenses. This however does not limit the user's interaction space to two dimensions. Furthermore, while this thesis centers on graph visualization, it will not develop novel graph algorithms or graph layouting techniques.

Data Exploration Tasks The thesis addresses data exploration, i.e., the process of understanding the data and finding interesting patterns and structures within the graph. To achieve this, it discusses existing task taxonomies and categorization and builds on the wide range of existing literature [AES05; VPF06; Lee+06; Yi+07; BM13; Sch+13; PPS14]. Therein, the research will consider manipulation of the view but not the data itself, meaning it will not focus on graph editing. It will further not discuss data presentation where a single user often navigates data in previously prepared ways to present it to an audience, e.g., in boardroom settings.

Interaction Modalities This thesis explores the potential of magic lenses for a wide range of interaction modalities. In order to keep the scope of this research manageable, this work addresses modalities suitable for work on large vertical displays including touch as well as tangible interaction and its extension to spatial movement, and finally the investigation of the body as an input medium. Certainly, there are further modalities and display setups that are of interest for data exploration but are not part of this research work, including but not limited to speech input [SS17], gaze-based interactions [Ste+11], and augmented reality settings [Büs+17].

Enabling Multi-User Scenarios While the focus of this thesis lies in extending and enabling manipulation of lenses, it also considers co-located multi-user scenarios for the specific case of data analysis. The proposed lens interaction techniques are designed with multiple users in mind and consider specifically requirements of co-located collaboration. This includes phases of closely coupled and loosely coupled, parallel collaboration, each of which will be contemplated in parts of this thesis. However, this work does not focus on collaboration per se and remote collaboration is not in the scope of this research work.

1.2 Methodological Approach

Research presented in this dissertation builds on an initial analysis phase which includes a thorough literature review and the analytical investigation of related work on magic lenses leading to a comprehensive design space. The following phase consisted of multiple iterative design processes with diverse interaction modalities (see Figure 1.5). Within these design iterations, formative studies were conducted to evaluate design decisions and empirical investigations were made to understand user behavior. Details of these phases are described in the following.

Analytical Investigations and Literature Review The research described in this thesis started with a literature review on existing works regarding natural interactions for information visualization tools and graphs in particular. This included a comprehensive investigation into existing designs and developments concerning magic lenses, their application, and interactions. These existing research works on lenses were categorized by their data type, addressed interaction tasks, effect class, and effect extent as well as the interaction modalities that were applied to move and configure the lens (if at all applicable). Based on these analytical investigations, challenges and requirements were identified in relation to the thesis goals.

Interaction Design with Various Interaction Modalities Considering the potential of magic lenses in conjunction with the identified requirements, this thesis includes the iterative conceptual design of interaction techniques for data analysis exploring various, diverse interaction modalities to support the configuration of magic lenses. Starting with interactions closely coupled to the contact display using touch, this work extends interaction with lenses by designing for user's body awareness and skills [Jac+08]. As a result, interactions are proposed that place the lens or its controller in the hands of the user employing the benefits of proprioception and spatial input to lens manipulation. Finally, this thesis explores the design space of applying the users' bodies as lens controllers.

Prototyping and System Design Multiple consecutive prototypes were designed and realized to iterate and evaluate the proposed concepts. In particular, it required special effort in system design to enable the dynamic combination of lens functions and the exchange of lens shapes. Furthermore, the design had to adjusted to the demands and requirements of the various input channels provided by different sensors (e.g., optical tracking mechanisms) and the coordination of multiple devices in a single application. The prototype components were created to be very modular to exchange input modalities and reuse modules for the varying application cases.

Design Evaluations and Empirical Investigations For interaction modalities that are already established or thoroughly explored in research, like touch and tangibles on tabletops, we evaluated our concrete design solutions by investigating their efficiency and usability in formative quantitative studies comparing them to existing solutions. For less well-understood modalities, novel display and device combinations, we

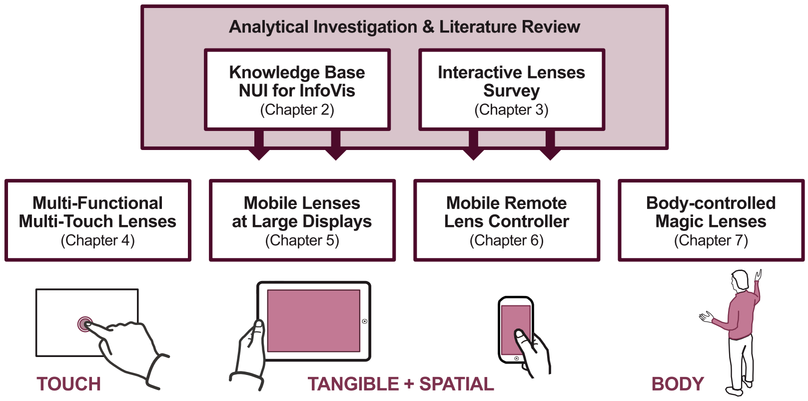


Fig. 1.5.: Building on an analytical investigation and literature review, this thesis explores diverse interaction designs to extend lenses both in terms of function and interactive configuration.

believed it to be more essential to understand the behavior and movements of users. Since there are no comparable setups that support multiple users and ease data exploration with lenses, we conducted qualitative evaluations focusing on the users attention and workflows within the diverse setups and interaction techniques.

1.3 Contributions

The following contributions are presented within this thesis:

1. This thesis provides a **systematic analysis** of the concept of magic lenses for visualization. This includes their definition and a discussion of their properties and dimensions which are core factors in understanding possible manipulations and configurations of the lens. The analysis further contains a categorization of existing literature on magic lens functions as well as interactions for magic lenses.

2. The research presented in this work contributes the **functional extension** of magic lenses from a very specific single-function instrument to a flexible multi-function, multi-purpose tool that can be independently parametrized and allows personal settings fitting the user's current goal and tasks.

3. This thesis proposes several interaction techniques with magic lenses for the exploration of graph data in node-link visualizations using diverse interaction modalities. This includes the systematic realization of multiple **interaction design** iterations and prototype development addressing specific challenges and requirements. In particular, it contributes interaction techniques for multi-touch manipulation of lenses, controlling lenses using mobile devices in front of large displays, and a novel concept of body-controlled magic lenses.

4. Within the interaction designs, lenses were extended further in terms of their interactive character by making them **personal tools and territories** for parallel work in collaborative scenarios. These principles were applied and investigated both in a setup for body-controlled lenses (see chapter 7) as well as for a mobile toolbox of personal, independent manipulations on the device in relation to the context visualization (see chapter 5).

5. Finally, the insights of two qualitative investigations are presented in this thesis that address general **aspects of user position and movement** in front of a large wall-sized display while exploring data. This concerns both the individual exploration of a graph visualization by a single user as well as the collaborative behavior of pairs of users analyzing data in multiple coordinated views.

1.4 Thesis Outline

This thesis is structured into eight chapters summarized in the following:

Chapter 2 provides a summary of the theoretical foundation and existing literature on which this thesis work is based. It gives a brief overview of natural user interfaces (NUI) and some of their relevant interaction modalities before going into detail on NUI for InfoVis in general and graphs in particular, including a description of application cases for graph exploration and a summary of interaction tasks.

Chapter 3 describes the analytical investigations and categorization of magic lenses incorporating its definition and discussion in regard to general InfoVis principles. Furthermore, the basic properties and characteristics of the lens are described. Existing lens functions are categorized with a specific focus on graph lenses. Finally, the chapter provides an overview of existing research that applies interactions to magic lens usage resulting in a range of challenges for this thesis work.

Chapter 4 presents MULTILENS, a novel concept of touch-enabled lenses that incorporates the functional extension of magic lenses. This includes the development of lenses as a flexible tool with multiple lens functions and their combination as well as the parametrization of individual functions. It suggests both widget-based and gesture-based approaches for touch-enabled lens configuration to investigate and discuss different interaction styles adapted to the user's experiences and preferences. The multi-touch design is evaluated in a quantitative study in comparison to a state-of-the-art menu.

Chapter 5 focuses on the use of graspable, tangible objects for magic lenses. The chapter starts by shortly describing possible dimensions of tangible lenses and our investigation of passive tangibles on tabletops. It then proposes GRASP, a mobile *representation* of the lens as a toolbox for graph exploration. It puts a tangible lens in the hands of the user and thereby supports individual exploration of a selected subset of the data. The design is evaluated in a qualitative study investigating the distribution of focus and workflow of users with the tool.

Chapter 6 contributes a tangible lens *controller* to manipulate content from any flexible position in front of the wall-sized display. Integrating graph visualization and lenses into an application of multiple coordinated views, the chapter focuses on the consistent combination of touch interaction on the display with distant interaction in front of the display called DI.VI.CO. A qualitative evaluation with pairs of collaborating users shows the benefits of the approach while focusing specifically on aspects of user positioning and movement during interaction.

Chapter 7 proposes the use of body interactions to control the lens and thereby presents the novel concept of BodyLenses. Therein, the lenses are extended to become more than a magic lens but rather a collection of tools and personal territories for interaction. The chapter provides a discussion of design space dimensions regarding variance in appearance and shape of lenses, suitable mappings of body interaction to lens configuration, as well as concepts presenting possible benefits of BodyLenses for multi-user scenarios.

Chapter 8 compares and contrasts the proposed solutions regarding their value for information visualization exploration, discusses the limitation of the work, and focuses on future extensions and investigations. It summarizes and concludes this thesis with a reflection on the presented work.

2 Background: Natural Interaction for Information Visualization

Contents

This chapter presents an overview of the background and related work upon which this thesis is built. To encompass all the relevant information, it is separated into three parts concerning the general trend of natural user interfaces and novel display environments (2.1), the application of natural user interfaces to information visualization (2.2) as well as an investigation into the specific application case of graph data and the existing use of natural interactions for their most often applied representation as node-link diagrams (2.3).

The former sections give a short overview of the basic principles of natural user interfaces and some of their relevant interaction modalities before going into more detail on information visualization, i.e., the visualization of abstract data to gain a better understanding of the data, trends or patterns, and extract information and knowledge. Interaction is an essential part of visualization with intensive interest and discussions in research concerning the improvement of interaction for more efficient and effective data analysis. Taking this further beyond the desktop using novel, natural interactions is one contribution of this thesis. The chapter will hence examine existing research on making information visualization available in these novel interaction spaces, for example, on mobile phones, tablets, and in meeting rooms. The last section will then look specifically at visualizing node-link diagrams, application cases and scenarios where graph data are applied, and existing work that investigated the exploration or manipulation of graph data with natural user interfaces.

2.1 Natural User Interfaces

Traditional interaction techniques at the end of the 20th century have been relying mostly on mouse and keyboard for working with computers. They build on established principles that are still used in today's user interfaces, e.g., in regard to *Windows Icons Menus Pointer* (WIMP) and *graphical user interfaces* (GUI) [WW11]. With Natural User Interfaces (NUIs), a new type of interaction techniques and hardware advances developed that take another step forward in making interactions with the computer more natural by adapting to the users' natural abilities, capacities, and behaviors. These novel interaction techniques have started to gain popularity outside of research through the widespread rise of mobile devices, both smartphones and tablets, as well as a general trend and openness towards new technology and interaction products (often promoted through gaming products), e.g., Wii Remote, LEAP Motion Controller, MS Kinect, and MS HoloLens. The following sections will shortly describe and discuss the characteristics of NUIs before summarizing the interaction modalities relevant for this thesis including some of their advantages and limitations.

2.1.1 Description and Characteristics

Natural user interfaces or sometimes more generally *natural interaction* describe a range of interaction modalities and techniques that focus on getting interfaces and interactions closer to the users and their natural behavior by taking advantage of the users' experiences from the real, physical world [Nor10; WW11; PD15]. Therein, this principle is strongly connected to the concept of *reality-based interaction* [Jac+08] which takes into account the capabilities, awareness, and skills of humans in regard to naive physics, their body, environment, as well as their social context. This is used to design user interfaces that are not limited by the WIMP principles but apply a wider range of interaction possibilities. In particular, these interfaces hope to ease communication between human and machine in a way that is closer to human-human communication [PD15]. As such, it is their goal to further bridge the *gulfs of execution and evaluation* [ND86] reducing the gap of the user's intention to the systems mechanisms and the gap between the systems reaction and the user's perception and understanding thereof, respectively.

This is to say that users apply existing principles from the physical world such as touching and grabbing content or gesturing to convey meaning. Many NUIs extend the existing WIMP and GUI paradigms by adding interactive gestures. While functions in graphical user interfaces can be visually identified (e.g., as buttons or handles) and thereby rely on our ability of *recognition*, gestures need to be *recalled* [Nie94]. In some cases this interaction may be intuitive enough to be easily accomplished because of its similarity to the real world (e.g., grabbing and rotating an item). However, other more complex or abstract features are hidden in the interface (e.g., resizing an image using pinch) and require additional effort of memory. They can be easy to remember though when well-designed metaphors are in use to ease this recall and learnability. As a consequence, they allow users to apply their natural behavior to invoke results and features that would not be possible in the real world. Therefore, they build on human experiences and knowledge and enhance what is possible with additional digital effects.

While these interaction techniques are designed to focus on the natural abilities of the users, the term *natural* has been disputed and questioned [Nor10] as the word itself is difficult to characterize, since even mouse interaction applies human motor skills. Further, even so-called natural user interfaces have a range of limitations. First and foremost, humans' natural interactions, and gestures specifically, are very complex, depend on context, and are multi-layered. As a result, they are very difficult to interpret, sometimes even for another human, and can hardly be consistently understood by a computer system. While natural interactions aim to understand natural behavior, what is considered natural interaction today has not yet reached that point of interpretation. Furthermore, simply because interactions are more *direct* does not make interactions natural. For example, bringing interfaces to touch-enabled surfaces does not necessarily result in *natural* interactions. An important aspect is the discoverability, visibility, and feedback provided by natural user interfaces. This is often related to the trend of reducing the number of widgets and icons by allowing more direct interaction and thereby an increased hiding of the interface (as predicted for the computer of the 21st century [Wei91]). Compared to mouse and keyboard interactions, the switch to invisible interfaces can be interpreted similar to one from menu-based usage to hotkey or shortcut usage – switching from expert mode in one system to novice in the other, which has been called *gulf of competence* by Wigdor and Wixon [WW11]. However when learned, gestures have the additional advantage that they can be phrased together [Bux95] in sequence to form continuous, more natural interaction flows. It is these principles of fluid interaction [Elm+11] and phrasing of interaction chunks [Bux95] in combination with the direct manipulations [Shn83] that make natural user interfaces *seem* more natural. For this reason, this thesis will continue to refer to these novel interfaces and setups as natural user interfaces despite the existing discussions of the term.

2.1.2 Interaction Modalities for Natural User Interfaces

In the following, an overview is given presenting interaction modalities associated to NUIs that are relevant for this thesis and can or have been applied to the specific

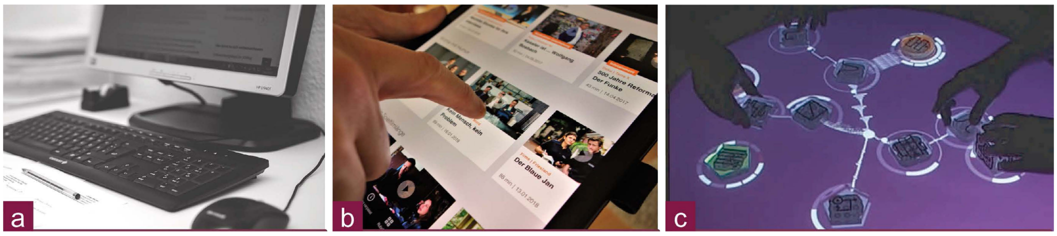


Fig. 2.1.: The predominant interaction with computers in the past relied on mouse and keyboard interaction (a) that mostly apply graphical user interfaces. However, interactive surfaces have become increasingly available leading to popularity of touch-enabled devices (b) and as a result research on natural user interfaces considering the users' natural abilities and haptics (c, *reactable* [Jor+07]).

application case of information visualization, and/or can potentially be used for interaction with magic lenses. It is by far not comprehensive. Speech commands, pen interactions, gaze-based interactions, and other modalities are not part of this overview since they will not be addressed in this thesis work. Some of them are however very shortly discussed in regard to existing interaction with magic lenses in chapter 3.4. For a more extensive review of the diverse interaction modalities used in natural user interfaces please refer to the overview works of Wigdor and Wixon [WW11] and Preim and Dachselt [PD15] (in German).

Touch Interaction For touch-enabled user interfaces, direct manipulation [Shn83] is extended by eliminating the indirectness of the pointer control and allowing the users to touch data objects and widgets with their fingers. While touch interaction has been investigated as early as the 1980s [LBS85], its main break-through arrived with the introduction of the Apple iPhone in 2007 [@Bux07]. While the product sales and hence the public's perception of touch usage is still mostly focused on mobile devices, such as smartphones and tablets (see Figure 2.1b), as well as touch monitors for laptop or desktop use, in research there has been a variety of touch interactions on and around larger display surfaces, such as tabletops and wall-sized displays. Especially these larger touch-enabled surfaces allow taking advantage of the natural abilities of bi-manual interaction and the different roles of the hands while interacting [BH99; MI94]. Beside these advantages of natural and direct interaction, there are a range of limitations to these devices: For smaller touch-enabled devices like mobile phones or tablets, an obvious challenge is to cope with limited resources (e.g., display space, computing power, memory) [Chi06; EWE09; Ber+13] when looking at presenting either information spaces or visualization. However, an even more general limitation of touch interaction is summarized as the *fat-finger problem* (e.g., [BWB06; Voi+09; WW11]), which describes the ambiguity in position that arises from the imprint of the soft fingertip together with the occlusion of the target object by the finger. This strongly restricts the precision achieved in touch interaction. Different solutions have been proposed to overcome this problem: For instance, one possibility is using an offset between finger and cursor [BWB06] which, however, interferes with the directness of the interaction. Alternatively, the use of digital pens provides smaller contact input [Hin+10]. As the main consequence, this limitation requires that touchable areas, e.g., touch buttons, have to be adjusted to appropriate size and the distances between objects increased to reduce unintentional interactions with the system.

Tangible Interactions The term *tangible user interfaces* [IU97; SH10], originally called *graspable user interfaces* [FIB95] and more broadly defined as *tangible interaction* [SH10], describes the use of everyday physical objects to reduce the gap between the digital and the physical environment. These tangible, graspable objects add haptic feedback to the control of digital objects and properties and thereby more strongly couple input and output space [Ish08]. As a result, they serve the natural abilities of perception and haptics that humans learn to understand from early childhood in the physical world. Therefore, these physical objects have the advantage of supporting fine adjustments through the capabilities of the hand with its various grips [MI94; SH10]. There have been varying categorization of tangibles focusing on types of tangibles from constructive assemblies and actuated tangibles to token and constraints and augmented objects as well as tangibles used on interactive surfaces [UIJ05; Ish08]. These tangible interfaces have often been investigated in research for their aid in providing increased efficiency or productivity [ZG13]. Using tangibles in combination with interactive surfaces enhances touch interaction with the surface by adding the haptic feedback tangibles provide. Here, tangibles have been applied to invoke functions, as physical controls to adjust parameters, representing specific data objects, or as containers for data objects [RUO01; Wal+06]. Successful examples of tangible on display surfaces are the *MetaDesk* by Ullmer and Ishii [UI97], which already includes first physical lenses as windows into the information space, as well as the system *reactable* by Jordà et al. [Jor+07] that has been used for live music performances (see Figure 2.1c). In these cases, interactions with the tangible are not limited to the placing and re-positioning, but also incorporate the rotation as well as flipping and stacking of tangibles [BBR10; Cha+12; Klu+12].

Spatial Interactions Strongly related to tangible interaction, spatial interaction focuses on lifting a tangible object from the surface and moving it in space. Sometimes the context relation of its movement is given by another display, however spatial interaction mainly regards the movement of the object relative to the user's body. When used for parametrization or selection, this body-relative movement eases recall of previous positions by supporting users' natural ability for physical mnemonics and perception. Extending the vocabulary of tangibles on tabletops, this additional dimension creates an even richer set of interaction possibilities [Spi+10]. However, the range of this vocabulary strongly depends on the tracking system used. As for

Fig. 2.2.: Adding mobile devices to large displays supports remote manipulation of parameters (a, [JDF12]) or controlling a cursor from a distance (b, *PointerPhone* [SBR13]) to transfer content (c, [Lan+16]).

tangibles on tabletops, the graspable, spatially-aware object may only function as a controller while the output of the selection or parameter adjustment is shown on another display. However, the tangible used for movement is not necessarily limited to an input controller but may also incorporate an output display (either independently or in addition to a context display). This may be a high-fidelity prototype using projection on cardboard [SSD09], which is lightweight but requires additional hardware, or nowadays can simply be a mobile device, smartphone or tablet. As such, mobile devices can be seen as an extension on the traditional tangible. They have further been used as peepholes into an information space [Fit93] and their spatial movement has been shown to support more efficient navigation [Yee03]. This spatial movement of a mobile device has also proven to be faster for navigational tasks (zoom and pan) than standard touch interactions [Spi+14b]. Further, the spatial movement has been used in both vertical and horizontal physical layouts of the information space with different effect on physical and mental demand, vertical having higher physical and lower mental demand than horizontal [Mül+15]. When adding an additional context display, the mobile device may present different views depending on its relative position [SSD09; SBD12].

Interaction in Multi-Device Environments In combination with large displays, mobile devices lend themselves to present detailed information or selected parts of the data at high-resolution [Voi+09; CLM12; Zad+14]. However, the connection and transfer of content between devices requires additional software solutions (e.g., [BFE15]). In this setup, the role of a device is often dependent on its distance to the large display. The mobile device can be used near the display surface [Lei+15] and hence trigger the connection through direct contact [Sch+10a]. To synchronize content and control the view of a large display or projection during multi-disciplinary team discussions, tablets allow shared pointing, navigation, and annotation possibilities [Olw+11]. When tracking the position and orientation of the device, the entire mobile device can function as a direct remote controller and manipulator of the content on the large display [Bor+10]. Similarly, mobile devices can be used as a directional pointer towards the large display [SBR13] (see Figure 2.2). This can be used to select and navigate content at different scales [PNB09] or transfer and arrange information and data objects from a distance [Chu+14; Lan+16]. User interface items on the mobile device may further be used to manipulate content and set parameters [CBF14; Zad+14]. These elements can be sketched on the mobile device in order to dynamically define the varying possibilities [TBJ15]. Alternatively, tangible objects can be attached to the mobile [JDF12] and allow eyes-free control of the view on the wall-sized display. This is especially relevant since *attention switches* between displays can have a negative effect on performance (even if not necessarily perceived as such by the user) [RNQ12b]. This effect can be influenced by considering aspects such as the displays' contiguity, the angular size covered by the displays, content coordination between devices, and factors of input directness [RNQ12a]. While this thesis addresses adding devices to a larger context display, there are other multi-device solutions which have introduced the possibility of combining multiple mobile devices to extend display space and distribute content either by using touch gestures [OT12] or recognizing proximity [LK12; Räd+14; LHD18].

Physical Navigation and Proxemics on and around Large Vertical Displays The main advantage of high-resolution, wall-sized displays are their display space and resolution. These properties makes them interesting for both presenting visualization for data analysis, providing space for large amounts of data and/or various views onto a data set, as well as room for multiple users investigating the information space. In comparison to tabletops, large vertical displays also eliminate the orientation problem as all users interact from the same side. However, reachability of all areas of the information space is a concern and requires additional solutions [Kha+04; GCR14]. Furthermore, vertical displays have the advantage of allowing visibility of information at varying distances from the display, e.g., already getting an overview of information from afar (which is hardly possible on tabletops due to the unsuitable angle). At wall-sized displays, movement may even be required to perceive all information presented on the display and has been shown to improve performance in spatial visualization using *physical navigation* [BNB07], i.e., the physical movement of body, head, or eyes to navigate an information space to select what is perceived, instead of virtual navigation consisting of zooming and panning. Using this principle, studies have proven that the spatial organization of content on large displays can support recall and improve sensemaking tasks [AEN10; AN13]. Interaction with visual representations presented on large screens depends largely on the input capabilities of the display (e.g., touch or pen). Additionally, it can be helpful to employ sensors that detect the physical movement in front of the display. For public displays, this registered movement has been interpreted to provide information on the users interest and intention [VB04; Mül+10] and specific distances and zones in front of the display have been assigned to distinguish passers-by from subtle and more direct interactions [Mül+10] also taking into account the users' attention towards the display [Dos+14].

Taking these inherent movements further, the principle of proxemics, originally discussed for inter-human contexts [Hal90], has been applied to human-machine interactions as *proxemic interactions* [BMG10; Gre+11] applying dimensions such as position, distance, orientation, movement, and identity to interpret and react to the user's behavior. This principle has been used to zoom-in and provide more details when a user leans closer [HD08], adapting interface elements from public to private modes [VB04; Bru+14], or to present specific device controls based on their distance [Led+15]. These large displays also lend themselves to multi-user scenarios and *collaboration*. When collaborating, users have been seen separating the space into varying territories, e.g., private, public or group, and storage areas [SCI04], which has similarly been observed for large display collaboration [Aza+12]. However, these territories have been identified to be transient both when users were moving around a tabletop instead of sitting [Tan+06] as well as for interactions around wall-sized displays [JH14]. As users position themselves when collaborating, it became clear that the distance between users has an impact on the effectiveness of their collaboration [Haw+05], specifically users' collaboration improves when being closer together. However, users have been observed to move fluidly between working closely together and separating from one another for parallel work phases [JH14]. Furthermore when working together and exchanging information, users position themselves in so called f-formations [Ken90; MHG12] which can be evaluated to identify users' intentions.

Body Interactions and Mid-Air Gestures at Vertical Displays Besides considering the general body position as has been done for proxemics, the body often functions as a reference for any interaction. The concept of body-centric interactions has been used to describe the virtual attachment of features and tools to the body to invoke functions or adjust parameters [Sho+10]. Together with a large display, the active application of the body as a tool can be made visible, e.g., by rendering the shadow of the body [STB07; Elh+15] (see Figure 2.3). Similar to using fingers for touch on the wall-sized display, the shadow's fingers can be used to eliminate the reachability problem of elements too high to reach otherwise [STB07]. For multi-user scenarios,

Fig. 2.3.: The user's body contour can be used to support advanced interactions (a, *Shadow Reaching* [STB07]) and may further provide additional tools (b, *Shadow-Touch* [Elh+15]). This is especially useful as mid-air gestures support triggering content even from the distance (c, touchless menus [CB14]).

these shadows further enhance the separation of user-associated tools and support individual application icons within the users' contours [Elh+15] as well as individual widgets such as personal keyboards or arm-attached sliders [Sho+10]. Due to our body awareness and skills [Jac+08], movements along these body-defined axes do not require visual attention. Even more, these types of mid-air gestures support remote interaction with devices and displays which would otherwise require the user to step closer and touch. Using mid-air gestures users can select elements [Jak+15] or trigger menus [CB14] from afar and thereby reduce the required movement and enable flexible positioning. However, general body movement as well as mid-air gestures are very natural behaviors of the users – as such they suffer from the always-on problem [WW11]. While mid-air gestures can be consciously performed, in the majority of cases they are also part of unintentional movements and behaviors, which are very hard for a system to distinguish from intentional ones. As in any gesture-based system, gestures need to be learned and hence require additional effort in their design, e.g., by showing additional objects to which a user may react to make the gesture self-revealing and selecting appropriate gestures that conform to a suitable metaphor of the function to ease recall [Nie94].

2.2 Information Visualization goes NUI

With increasing amounts of data, there is a growing need to extract, understand, and explore information. Visualization is about representing this data in a way to improve communication and perception of the information [Rob+14], while information visualization (InfoVis) as one of its sub-categories focuses specifically on abstract data representations [Spe14]. As such, information visualization enables getting an overview of interesting data and quickly finding the relevant aspects to explore further [Fek+08]. While the static representation of data is a first step, it is well known that information visualization increases tremendously in power with its interaction [CMS99; Spe14; Tom15].

With the development of natural user interfaces, InfoVis techniques can harness the advantages of this technological advancement to make interaction with data more direct, graspable, and immersive [Rob+14]. This benefits information visualization by reducing the number of UI elements to focus more strongly on the data itself and thereby reducing the gap between the user and the data for more direct interaction [Lee+12; Dru+13]. Another advantage of NUIs is that they support fluid interaction for information visualization by reducing indirectness, and creating continuous interaction flows to enable smooth data exploration [Elm+11]. Furthermore, NUIs enable joint data exploration with multiple people collaborating to gain more insights into the data [Lee+12]. However, all in all NUIs also create new challenges [Voi+09; JD13], e.g., large amounts of data lead to dense visualization, cluttering, and overlapping items which stands in clear contrast to the imprecision of touch (cf. *fat-finger-problem* on page 16). While there are already building blocks for visualization design, there are still a lot of challenges of engineering and designing consistent interaction vocabularies for use in InfoVis [Tom15] since users' usage and behavior when working with NUIs in the context of visualization needs to be investigated and well understood to design appropriate interfaces and tools.

At the beginning, research at the border of InfoVis and NUI mostly took advantage of the tabletop setting [II13]: Examples make use of its touch capabilities (e.g., [Voi+09; FHD09]) as well as the comfortable positioning of multiple users (e.g., [Wal+06; SSD09]). Furthermore, tabletops allow use of tangibles which is made easier in comparison to horizontal surfaces where placing tangibles requires additional solutions. More recently, with their increased availability and the distribution of mobile devices into everyday life, the number of investigations into InfoVis on multitouch-enabled tablets increased as well [BLC12; SS14; SS16b]. At the same, InfoVis has also been applied to the high-resolution environment of wall-sized displays [JDF12; BI12; Jak+13; Zad+14] with its advantage of stepping from and towards the display for improved overview and detail (e.g., [Ise+13]). Since all these interactive surfaces share their capability for touch interaction, the following sections will start by focusing on the possibility to *touch* for configuring information visualization including directly interacting with data objects, axes, canvas, and exploration tools. Afterwards, the research on large display spaces and specifically vertical, wall-sized displays will be elaborated on including investigations into the users' movements as an essential component of interaction. Adding to this display setup, additional devices and their capabilities in relation to larger context displays will be examined before focusing on the application of these setups to collaborative data analysis.

2.2.1 Making Data Touchable – Supporting Data Analysis Using Touch

Touch interaction for InfoVis opens up a range of possibilities to improve the directness of interaction with data. As for any interaction design, it is important to understand the users' intentions, process, tasks, and context. Researchers hence frequently base their design on task analysis to map appropriate touch gestures to fitting tasks [Dru+13; SS14]. Similarly, it is possible to thoroughly investigate the actual use of visualizations on whiteboards to understand possible touch and pen behaviors when creating and manipulating visualizations [Wal+12]. Alternatively, user-elicitation studies have been used to develop a deeper understanding of the users' mental models and preferred interaction gestures [FHD09; WLI14].

A range of specific individual visualization techniques have been made available on interactive surfaces and tablets specifically, e.g., stacked graphs [BLC12], scatter plots [SS14], or star plots [Lan+15], as well as visualizations with multiple coordinated views [SS16a]. The following section focuses on touch on interactive surfaces independent of their size. However, it should be mentioned that visualization on mobile devices comes with its own advantages and challenges [Chi06; EWE09; Ber+13] regarding resources such as processing power, memory, and display space.

WIMP and Gesture-based Touch Interactions for Data Analysis

A first step of designing for interactive surfaces has often been made by applying known interface solutions from traditional desktop interfaces. Existing principles from graphical user interfaces and the WIMP paradigm are familiar to the users and have the advantage of providing them with visible interface elements to recognize for first exploration of the interface (cf. previous *recognition vs. recall*). Using this familiarity, researchers have applied known widgets to touch-enabled surfaces, e.g., selecting from a combo box to sort data items [Dru+13] (see Figure 2.4a) or pushing a button to present an alternative encoding like a table version of the data [SS14]. Especially when systems exist on multiple devices their consistency is often achieved by reusing the traditional interface components for touch interaction with only limited adjustment. At the same time, touch interaction would allow enhancing the interaction with data by making it more direct, e.g., touching data directly to select or pinching on the data canvas to zoom-in in-place of the interaction.

Drucker et al. [Dru+13] present *TouchViz* which they used for a case study that compares a conventional interface with touchable control panels against another touch-focused interface that reduced interface elements by supporting a range of direct touch interactions (see both interfaces in Figure 2.4a-b). The majority of their participants not only preferred the gesture-based interface, but made significantly fewer errors in significantly less time than with the conventional interface design. However, as discussed for gesture-based systems in general, questions of learnability, recall, scalability to a wide number of functions, and consistency between interfaces

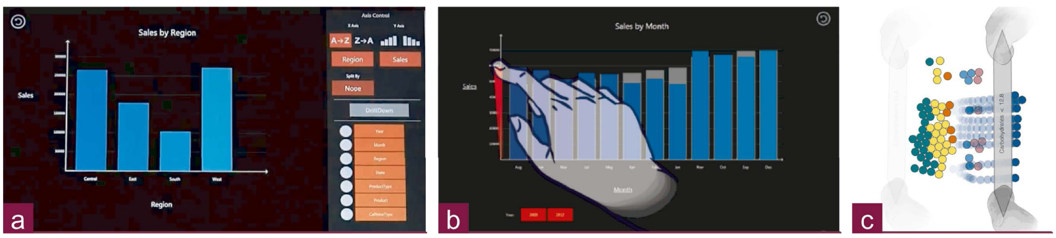


Fig. 2.4.: While WIMP-like interfaces can be applied to touch surfaces (a, *TouchViz* [Dru+13]), novel gesture-based ones outperform them by bringing interactions closer to the data (b, *TouchViz* [Dru+13]). These types of interaction can be extended using physics-based exploration tools (c, *Kinetica* [RK14]).

Fig. 2.5.: There have been various solutions adapting touch interaction to information visualization techniques, e.g., stacked graphs (a, *TouchWave* [BLC12]), scatter plots (b, [SS14]), and star plots (c, [Lan+15]).

on various different display setups remain. Taking both types of interaction into account, recent works have tried to balance interaction design by using interface elements while adding gesture-based possibilities to the interfaces, e.g., using control panels for advanced features while also supporting the user in selecting interesting data items by encircling them as has been investigated by Sadana et al. [SS14; SS16b] (see Figure 2.5b). The following section will go into more detail about individual interaction techniques and the interaction vocabularies that have been used on the varying components of InfoVis techniques.

Touch Interaction with InfoVis Components

In the previous section, the advantages of directly touching data were discussed. Many research works already apply this advantage supporting touch on the actual **data objects** within the visualization. In the simplest case, the basic movement of objects can help better understand the visualization. This is the case for node-link diagrams, where moving nodes using touch [FHD09; FHD10] can accomplish direct interaction to understand relations between data objects by seeing the movement of links and visually decluttering the graph. Frisch et al. [FHD09] specifically apply both touch and pen interaction for the creation and manipulation of nodes and links, adding pen for precision and its natural role for drawing, e.g., to create both nodes and edges. Using the same visualization technique, Schmidt et al. [Sch+10b] further add naive physics [Jac+08] to this natural interaction by allowing pinning and strumming of links, resulting in their vibration and thereby clear visualization of their connectivity and reach within the node-link structure. Riche et al. [Ric+12] add further concepts of grabbing multiple links together to cluster or spread them in user-defined ways. Further examples of natural touch interactions include *TouchWave* by Baur et al. [BLC12] which allows very direct interaction with individual parts of a stacked graph: *TouchWave* supports simply dragging data items out of the stacked graph to investigate them further in a separate view (see Figure 2.5a). Similarly, *TouchViz* by Drucker et al. [Dru+13] provides natural filtering possibilities in the bar chart where filtering can be accomplished by simply flicking bars downwards out of the visualization removing them from the view.

All these examples focus on actively touching data objects in the visualization for exploration. However, not only data objects but also the base components of the visualization techniques can be touched and thereby configured: First and foremost **axes** are an essential component to adjust for data exploration but also the **canvas** on which the visualization recites can be used for interaction. Sadana et al. [SS16b] explore alternative direct interactions for selecting data objects, e.g., by encircling items or spanning rectangular selections through drag on the canvas along the horizontal or vertical axis [SS16b; SS16a]. Both Drucker et al. [Dru+13] and Willet et al. [WLI14] use dragging along the axis for interaction, either to sort data objects (see Figure 2.4b) or to select ranges. Furthermore, the axis can be grabbed with multiple fingers and stretched to zoom into the visualization and make more room for data in a relevant area: In our work on graph visualization in relation to time [Mor+14], we use this action to stretch a time axis so that data objects' positions are adjusted and more information in the selected time span can be loaded. Especially in multivariate data visualizations with multiple axes, the interaction with these axes can make exploration easier, e.g., reordering the axes of a parallel coordinate plot to identify relations [Ins85] as well as using direct interactions to rearrange, zoom-in, or compare data in a star plot by touching, moving, and expanding axes [Lan+15] (see Figure 2.5c).

Finally, besides these inherent InfoVis components, additional **tools for data exploration** can be created and moved over the data using touch. Sadana et al. [SS14] implicitly create a magnification lens when using a pinch-to-zoom gesture on a scatterplot visualization which can then be dragged around. Schmidt et al. [Sch+10b] as well as Riche et al. [Ric+12] developed specific elements representing tools to pin edges in a node-link diagram to specific points or areas and thereby declutter parts of the visualization. Further, *Kinetica* by Rzesotarski and Kittur [RK14] presents a range of physics-based tools that can be moved over the data to explore, categorize, and filter data (see Figure 2.4c). Most of these tools could equally be applied to mouse-based interaction. However, their naturalness, behavior, and flow are emphasized even more by touch as the natural interaction fits their character.

2.2.2 InfoVis on Large Displays – Physical Navigation and Proxemics for Data Analysis

As previously described, large wall-sized displays can benefit information visualization due to their increased display space and resolution and are hence especially of advantage for complex, more difficult tasks [Liu+14]. This offers opportunities regarding presentation of large data sets and/or providing more information, details, and varying views for data exploration. At the same time, challenges arise as not all pixels can be perceived at a time and hence movement is necessary to see all

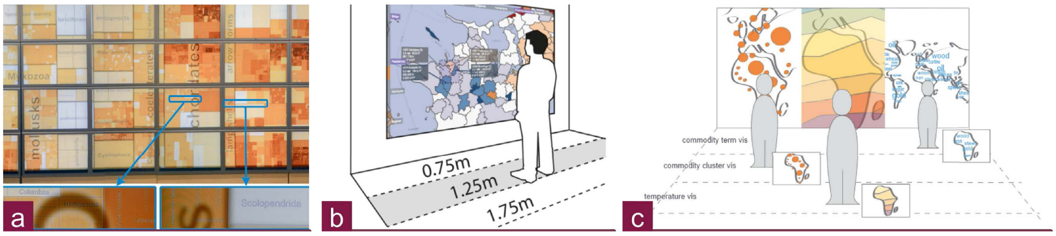


Fig. 2.6.: The user's distance to the display has been used to invoke alternative views either by taking advantage of the natural limitations of perception (a, [Ise+13]) or by actively tracking the user to provide more information (b, [Jak+13]) or other visualization types (c, *SpiderEyes* [Dos+14]) when stepping closer.

content presented (see general interaction in 2.1.2). Beside these general aspects, there are additional challenges of large display spaces regarding data perception that are especially relevant for visualization.

A challenge to consider for presentation of information visualization is the visual discontinuity resulting from bezels between the individual tiled displays that make up the display wall. There are two traditional approaches to adapt to the bezel: offset (i) continues the visualization on the next display as if no bezel existed while overlay (ii) removes part of the visualization as if the content remains behind the bezel. Both solution have disadvantages regarding perception making content at the bezel appear bigger than it actually is or hiding possibly relevant information, respectively. Almeida et al. [Alm+12] investigate solutions to this problem that take into account the user's position. For an additional aspect of perception, Endert et al. [End+11] investigate how visual encodings are influenced by the size and diversity of data on large displays finding that in the design of visualization visual variables can influence users' need for physical navigation. This is important since the perception of the data visualization — specifically the perception of visual variables such as length, angle, and area — are strongly influenced by distortion. Hence, the position and distance of the user may result in errors of judgment when selecting inappropriate viewing angels and distances as observed in two user studies conducted by Bezerianos and Isenberg [BI12].

However, the features and properties of human perception can also be actively applied to the advantage of data exploration. Isenberg et al. [Ise+13] propose hybrid-image visualizations that contain both details (visible at close range) and overview (visible from afar) in one presentation. Assuming that users aim to get an overview of the data when stepping back and taking advantage of the limited perception of content from afar, a blurred overlay is perceived at a distance (see Figure 2.6a) while the detailed structures become clear when stepping closer. This presentation at different distances however may also imply that users require interaction both when being close as well as when being farther from the display [Jak+15].

Furthermore, the position and distance of the users can be actively tracked and used as an additional input parameter. Jakobsen et al. [Jak+13] suggest a variety of mappings to the user's movement specifically for visualization purposes, including changing the visual encoding or filtering content due to the user's distance to the display as well as presenting more or less information depending on this distance (see Figure 2.6b) or changing a slider value by stepping left and right. In their follow-up work, Jakobson and Hornbæk [JH15] investigate physical and virtual navigation in a comparative study and found benefits of physical movement vary dependent on the use case. Providing more details and zoomed-in states when stepping closer seemed to be worthwhile and understandable to the users, presumably because this only increases the nature of physical navigation where our perception allows us to see more details when close and limits those details when afar. However, when virtual navigation is possible and used, these benefits can be made void as the spatial position of content changes over time. Dostal et al. [Dos+14] also adapted this principle of changing parameters dependent on the user's distance with the addition of considering multiple users interacting at varying distances and with diverse states of attention toward the display. Specifically focusing on providing a toolkit for tracking users and recognizing attention and collaboration states, their *SpiderEyes* system supports mapping the distance to varying detail states, zoom levels, or visualization types. Therein, each user or user group, recognized by their proximity and approximated gaze direction, is associated with a specific slice of the display surface and can explore their separated copy of the data visualization in this slice using their position (see Figure 2.6c). As an alternative to separating large display space into multiple slices, researchers have also explored the possibility of distributing content onto multiple displays.

2.2.3 Display Combinations and Distributed Visualization Views

Adding smaller, mobile devices in addition to a larger display has the advantage of providing more display space and allowing additional details and alternative views on the separate display instead of making room for these views on the display wall [Zad+14; BE16]. This enables keeping the layout and structure of the visualization on the large display consistent which is the prerequisite for the benefit of physical navigation and aids the user's spatial memory [Räd+13; JH15]. The separate, mobile displays may extract parts of the context visualization to show more details or alternative views: Spindler et al. [Spi+10] use additional mobile displays to provide magnified and thereby decluttered representations of the visualization, more details, or expanded views. The movement of the display itself is used to define the selection in relation to the context display.

Fig. 2.7.: Mobile devices have been used to control a selection on another display with the main visualization (a, *Thaddeus* [Woź+14]) and to show additional information and details (b, *Visfer* [BE16]). Multiple devices can also be joined together to support flexible combination of visualization views (c, *VisTiles* [LHD18]).

The additional mobile device can also be used to control the visualization on the larger screen, e.g., by defining queries to reduce the results shown. Klum et al. [Klu+12] investigated *Stackables*, tangibles that allow collaborative refinement of queries for a vertical display. *CubeQuery* by Langner et al. [LAD14] extends this work for a music collection on tabletops. Therein, the way the tangible mobile devices are positioned can have an influence on the combination of queries, e.g., rotating the display for negating [Klu+12] or positioning mobile displays on top of each other or side-by-side for AND or OR connections [LAD14]. Using precise positioning, Wozniak et al. [Woź+14] track the location of a mobile device in space and investigate their relative movements around a stationary tablet display. In their *Thaddeus* system, the mobile phone is used to present details of a bar chart, line chart, or tree visualization by selecting a specific slice of the data when moving the phone (see Figure 2.7a). Similarly, Horak et al. [Hor+18] investigate the use of small devices like smartwatches for use as clipboards for selected data sets that can be transferred to the context display for comparison and filtering later on. For scientific visualization, Besançon et al. [Bes+17] recently applied an additional mobile device to reconfigure cutting planes while having the main visualization view on a larger display at a distance. They show that the combination of touch on the mobile device as well as the movement of the device for tangible, spatial input is worthwhile and was received well by users in their study.

On the basis of tracking multiple mobile devices on a tabletop surface, *VisTiles* by Langner et al. [LHD18] presents a system of multiple coordinated views [Rob07] distributed onto multiple mobile devices. These varying views are adjusted in terms of scale alignment (see Figure 2.7c) and color scale to ease comparison when placed next to each other while smaller devices may also support UI offloading or overview and detail views. This can support a single user exploring data with multiple devices but may further improve joint data analysis as multiple users can flexibly arrange data views and pass them to one another.

2.2.4 Collaborative Data Analysis

It has been shown that collaboration for data analysis has great benefit and increases the number and quality of insights [CKW09; Ise+12]. To support this collaborative data analysis, many works first and foremost investigated the users' behaviors and strategies when collaborating on data exploration. As previously discussed, general aspects such as territoriality on and around different interactive displays (tabletops [SCI04; Tan+06] and display walls [Aza+12; JH14]) have been observed. Additionally, distance and proxemics between collaborating users [Hal90; Haw+05] and their device(s) [MHG12] as well as their positioning as a group [Aza+12; ZD17] have an influence on collaboration and need to be considered. Even more, all these aspects are very dependent on the current collaboration style, with users working at times in *closely or loosely coupled collaboration* [Ise+12]. In their work, Isenberg et al. [Ise+12] specifically focus on observations of collaboration for data analysis: Using an established VAST challenge as an analysis task, they observe pairs of users during interaction and as a result provide a set of eight collaboration styles grouped into loose and close collaboration. These include styles of engaged discussion on one side and parallel work on different problems on the other side, with an additional disengaged style representing the state when users remove themselves from the collaborative data analysis for a time. In their observation of users, they found strongly varying team strategies in regard to the time spend in either loose or close collaboration. Consequently, well-designed collaboration tools need to support a range of varying collaboration styles and the transitions between them.

In their analytical work on digital tables for collaborative exploration, Isenberg et al. [Ise+10] summarize a range of existing example applications designed for use in public contexts, e.g., museums, or specifically for data exploration. Out of these applications, they identify challenges that arise when multiple users explore data collaboratively. Among those is the often used manipulation of global parameters in visualization systems. These global changes of the view can trigger conflicts between users and hence interfere with group work. They require specific solutions, e.g., various independent, parallel views onto the same data set. Another aspect discussed by Isenberg et al. [Ise+10] is the complexity of data exploration which may result in reduced attention towards other users. They found that otherwise well-understood attention cues were not perceived while users concentrated on their own data exploration. Addressing this problem among others, *Lark* by Tobiasz et al. [TIC09] provides a meta-visualization in a shared information space that shows a representation of the visualization pipeline which resulted in the presented views. Thereby, it enables understanding of interaction steps and connections between views supporting user coordination since they can track the progression and the individual steps that lead to the current view state. Similarly, the use of brushing and linking with individual user colors can be used to coordinate interactions and highlight documents and searches already done by other collaborating users [IF09]. Building upon this collaborative brushing and linking, Mahyar and Tory [MT14] investigate linked common work to enable representation of relations between different users' findings for improved communication and coordination of the work.

In the context of interactive surfaces, most previous work focused on interactive tabletops for their advantage of allowing multiple users to sit down together and share the space from multiple sides. While some collaboration behaviors might be equally observable for vertical displays, not all insights can be adopted unchanged since aspects of eye contact between users, non-varying orientation of documents, as well as occlusion of content and movement from the display for overview purposes may occur differently from horizontal setups. While there has been research on collaborative work on large vertical displays (e.g., [Hal+10; Bra+11; JH14]), very little work in that setup exists that specifically addresses collaboration for the use case of information visualization and data exploration. Prouzeau et al. [PBC17a] apply this setup to specifically look at the benefit of multiple users for the task of selection in a node-link diagram (see Figure 2.11b on page 39) and also showed the benefit of checking and discussing each others work after loosely coupled, parallel interactions. At the same time, the difference in their tool selection, either locally restricted or with stronger global effect – "larger visual footprint" [PBC17a], highly influenced the need for coordination between users and hence time efficiency. Using the same setup, Prouzeau et al. [PBC17b] compare path-finding tasks with pairs of users on the large vertical display with multiple desktop interfaces and found desktop interfaces to be more efficient in regard to time. However, it also became clear that interaction on the wall-sized display produced results of high quality from the start while lack of time efficiency seemed to have occurred due to increased communication, including planning communication and coordination to avoid physical conflicts. This is equally observed in a study by Liu et al. [Liu+16] where pairs of users had to do manipulation tasks dragging and dropping discs using different techniques and asking for help from the other user resulted in both cognitive cost due to interruption as well as time cost. Nonetheless, this increased communication eased by the wall-sized display should also be acknowledged as a benefit when considering more complex tasks that require analysis and discussion. Participants of the study presented by Liu et al. [Liu+16] perceived communication to be worthwhile and lack of communication as less efficient and less enjoyable. This supports earlier work by Isenberg et al. [Ise+09] which describes that beside efficiency, aspects of shared knowledge, fun of collaboration, the opportunity to brainstorm, and the shared process of forming consesus are considered beneficial for data analysis. Regardless, to reduce additional efforts for conflict avoidance, further investigations and tool development for collaborative interaction on large vertical displays is required.

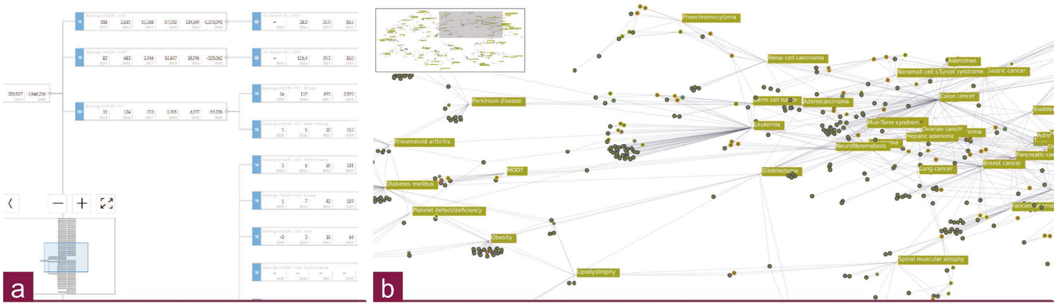


Fig. 2.8.: There are a variety of application examples and use cases that benefit from graph analysis, e.g., value driver trees (a, [HKD17]) or human disease networks (b, data by [Goh+07]).

2.3 Interactive Graph Visualizations

The analysis of graph data is relevant for many application cases and scientific endeavors [HW04; Lan+11; HSS15]. In the following, some of these application scenarios and interesting data sets will be shortly described to give a glimpse into the variety of use cases requiring graph analysis. Due to this importance, many investigations have been conducted into the various possible representations of graph data and the required tasks for analysis and exploration. This chapter aims at giving a brief overview of general principles in regard to graph data and node-link representations and will then specifically discuss interaction tasks for InfoVis exploration in general and graph visualization in particular, before getting into more detail on existing research on natural user interface solutions for graph analysis.

As this thesis focuses predominantly on interaction aspects and an interactive tool for graph exploration, please refer to the following survey and book publications for more information on general aspects of graph theory [Die05], multivariate graph data including their temporal and spatial references [HSS15], the representation of group structures [VBW17], or specific graphs such as time-dependent, dynamic graphs [Bec+17].

2.3.1 Graph Visualization and Selected Application Cases

The following will shortly describe three selected application cases for graph visualization and analysis to give an introduction into its usefulness before discussing general aspects of graph data and representation.

Value Driver Tree In business management, value driver trees are used to present a financial model of a company's internal values by presenting calculation and dependencies between key performance indicators and value drivers. These value driver trees (VDTs) can, for example, be used to provide insights into profit and loss by presenting revenue (top line) and expenses (bottom line) of the entire company as well as its departments. According to corporate partners, these VDTs may incorporate up to 2500 nodes with roughly over 3500 edges for large corporations. Therein, the value driver tree – despite its name – is not necessarily a tree but has multiple interconnecting edges where child nodes influence varying key performance indicators, making it a directed acyclic graph. To identify trends and tendencies using the VDT, each driver, e.g., number of product sales in a specific department and product line, should not only present the current attribute value but may also show past and future (i.e., predicted) values (see Figure 2.8a). The exploration of these values is key to a well-informed management and can be the basis for executive decisions. These graphs are hence used for both single users and multiple users to explore as well as simulate and discuss consequences and predictions in executive offices or boardroom setups.

Social Networks Connections between people can be described as social networks. These networks exist in a variety of use cases, e.g., to describe groups of friends, company employee structures, or even to understand criminal and terrorist structure and communication networks. While a social network of friends can simply be interesting to see, even non-professional users might want to examine social networks in terms of connections between literary characters for a book analysis. For professional use, advertising agencies analyze these data sets to identify test users and people likely to have influence on others. Police analysts explore complex social structures and organizations to identify threats and leads to prevent terrorist attacks. While nodes in these networks are always describing individual people or groups, edges can describe a variety of properties, e.g., different types of relations, communication channels used, or simply how well acquainted the connected people are. Depending on the use case, social networks tend to be very heterogeneous in scale: A graph presenting a small groups of friends might have below a hundred nodes. However, a large cooperation's employee network might contain multiple hundreds up to thousands of people while online social networks may incorporate multiple millions.

Human Disease Network Human disease networks describe connections between existing diseases and disorders. The diseases and disorders represented in the network are connected if they share one or more associated genes indicating the common generic origin of the diseases. They are additionally classified by their disorder class, e.g., cancer, cardiovascular, or immunological. Recent progress in genetics resulted in a better understanding of gene mutations as a trigger for a wide range of diseases. The human disease network by Goh et al. [Goh+07] is a join of two separate graphs, a network of human diseases and disorders and a network of disease gene connections (see Figure 2.8b). Hence, additional gene nodes exist and are associated with the disorders and diseases that they can trigger. This allows a much wider focus of studying diseases associations and supports identifying general patterns and principles that would not be apparent in studies of individual disorders. Exploration of these networks can hence be useful both for educational and investigative use.

As became clear, data sets from many domains contain items and relations that make up graph data and can hence be represented as nodes (aka vertices V) and edges (aka links E) forming a graph ($G = (V, E)$ with $E \in [V]^2$ [Die05]). The presented graph application cases contain graphs of sizes from hundreds to few thousand nodes where visual data analysis is still feasible with appropriate tools. Other application cases of larger and more complex dimensions may require an initial processing step of clustering or querying the data to reduce the number of nodes and enable visual analysis. In any case, graph data from the real world is rarely made up only of structural information of nodes and edges but also associated attributes to describe their type, weight, characteristic properties, or similar domain-specific information, creating a multivariate graph [HSS15]. Visualization research focuses on appropriate mappings of the data to visual representatives including the representation of those attributes but also structural information. While there are a range of possibilities to represent graph data, the most common and relevant for this thesis is the node-link representation. A large set of research works concentrate their effort specifically to designing appropriate layout techniques for effective presentation of these node-link diagrams [Lan+11] which is especially complex for large numbers of nodes and edges since these real-world graphs are hardly ever planar and lead to cluttering, mostly due to the number of edge crossings.

Different representations of the graph data have different qualities in regard to the characteristics of the data they present [Lan+11; GFC05]. While the node-link representation is the most common and hence most familiar representation showing the structure of the data as well as the individual entities that make up the graph, there

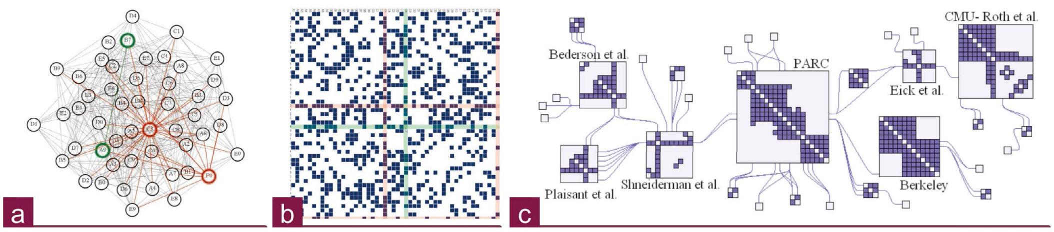


Fig. 2.9.: While the node-link representation (a, [GFC05]) is the most common and familiar, the adjacency matrix has been shown to be more efficient for specific tasks and larger graphs (b, [GFC05]). These representations can be combined to show only local communities in matrix form (c, NodeTrix [HFM07]).

have been others specifically for more refrained instances like hierarchical graphs and trees, e.g., space filling techniques such as treemaps [Lan+11]. For graphs in general, the adjacency matrix (see Figure 2.9) has been shown to be suitable for summarizing link connectivity and eliminating the problem of long edges that are difficult to trace in a large, possibly cluttered graph [GFC05]. Combining both these representations, Henry et al. [HFM07] propose NodeTrix which presents only sub-graphs in matrix representation while leaving the general structure connected in node-link representation.

This research on visual mapping and graph representation is extended by the aspect of interaction with these graphs which is essential for data exploration, analysis, and understanding [Lee+12; Tom15; Elm+11]. The next section will therefore examine existing research on task categories and taxonomies presenting an overview of related work regarding these tasks.

2.3.2 Interaction Tasks for Graph Exploration

The design of appropriate interactions for graph exploration requires understanding the general goals of the users and their low-level tasks that make up interaction for information visualization and graphs. This overview uses the general principles from the taxonomy of Brehmer and Munzner [BM13] as a guide. At the same time, it focuses on the generalizable tasks of Yi et al. [Yi+07] and applies those to the specific case of graph components according to Lee et al. [Lee+06] (see Table 2.1 for an overview).

The users' goals and intentions – why do they interact?

As a basis for interaction design, it is essential to understand the users' underlying intention for interaction with the data – synonymously named their goals [Sch+13], intents [PPS14] or the aspect of *why?* [BM13] the user performs a task. This incorporates sub-categories that involve aspects of improving the understanding: For example, understanding the value drivers of a company by making assumptions and verifying hypotheses [PPS14] regarding the reasons behind an increase or decrease of income resulting from a product line in value driver trees (which falls into the sub-category: consume information [BM13]). As a result, the intention behind interaction might be locating [VPF06] a specific component of the graph to retrieve its value [AES05] or its relations, i.e., adjacent nodes [Lee+06] (sub-category: search [BM13]). Finally, it might be the user's goal to get a wider overview of the data to aggregate information, identifying [VPF06] distributions, (attribute) ranges, clusters [AES05], or trends [PPS14] and thereby predicting the future [PPS14].

why? [BM13] goals [Sch+13], intents [PPS14]	
consume	make assumptions [PPS14], verify hypotheses [PPS14]
search	locate components [VPF06], retrieve value [AES05], find neighbor [Lee+06]
query	identify [VPF06] distributions, ranges, clusters [AES05], identify trends [PPS14], make predictions [PPS14]

how? [BM13] (*how?*–manipulate [BM13]), means [Sch+13]	
select	select [Yi+07; BM13], highlight, mark [BM13]
explore	navigate [BM13], scan [Lee+06], find anomalies or extremes [AES05], find cluster [Lee+06]
reconfigure	arrangement [BM13] esp. layout manipulations, sort [AES05]
encode	encode [Yi+07; BM13], configure, visualize [VPF06], change mapping [BM13]
abstract/elaborate	aggregate, segregate [BM13]
filter	filter [AES05; Yi+07; BM13]
connect	highlight association and relationships [Yi+07], aspects of adjacency and connectivity [Lee+06]

Tab. 2.1.: Overview of categorized exploration tasks adapted from Brehmer and Munzner [BM13] with focus on exploration tasks by Yi et al. [Yi+07].

The users' means – how do they interact?

To accomplish the above goals, users need to interact and manipulate the visualization. Brehmer and Munzner [BM13] categorize this as the *how?* category. These interactive manipulations of the view (*how?*–sub-category: manipulate [BM13]) can be described as the means [Sch+13] of how users explore the data and achieve these goals. These tasks hence focus on the specific interaction with individual components of the graph visualization. Adapted from Lee et al. [Lee+06] and its generalized extension by Pretorious et al. [PPS14], the interactions can hence be considered in regard to the basic structure they are concerned with, categorizing them as topology/structure-based, attribute-based, browsing-related, and as overview/estimation tasks. These high-level abstractions are made up of low-level interaction tasks. Lee et al. [Lee+06] apply those of Amar et al. [AES05] while the following will focus on tasks identified by Yi et al. [Yi+07]. Browsing tasks consist of repetitive low-level tasks from other categories and are hence not discussed separately here. Each low-level task can be applied to the various entities that make up the graph:

- node(s) + node attribute(s)
- edge(s) + edge attribute(s)
- node connectivity, path(s)
- structural properties, clusters, groups

Often considered the most basic interaction, **select** [Yi+07; BM13] provides a foundation for high-level tasks, supporting the specification of which node or edge lies in focus of interaction and should be considered for further examination. Selection can be applied to any of the aforementioned components of the graph, e.g., individual nodes, clusters, attributes, paths. Therein also lies the more complex aspect of multi-entity selection where the definition for selecting specific sub-structures and paths, including the selection of entities of different type, need to be considered. The **explore** [Yi+07] category describes the general navigation [BM13] to other components within the graph to scan [Lee+06] individual entities (nodes, edges, or attributes) or sub-graph structures and find relevant information, e.g., anomalies or extremes [AES05].

Beside selecting and moving through the graph to interact with the existing representation, **reconfigure** [Yi+07] operations allow changing the arrangement and relative position of visual elements providing a different perspective onto the data. For node-link representations, this specifically concerns the layout of the nodes considering, e.g., arranging [BM13] disease nodes from the human disease network according to a specific attribute value like disease class or sorting [AES05] them by number of affected genes. Even more than simple rearrangement, **encode** [Yi+07; BM13] incorporates the change of visual attributes of the visualization. This allows the user to configure [VPF06] and change [BM13] the inherent mapping of data to visual representation including the selection of which dimensions of the data are to be visualized [VPF06]. Note that Brehmer and Munzner [BM13] categorize encode in the form of the initial encoding of the visualization and hence as a separate entity of the *how?* category, parallel to the interactive modification of visualization elements (*how?*–manipulate). However, since it is possible to interactively manipulate the encoding on demand [GT14], this thesis considers encode as part of the manipulate category.

The category of **abstract/elaborate** [Yi+07] focuses on interactions that aggregate or segregate [BM13] information and thereby adjust the level of abstraction and visible number of entities and detail. Among other interactions, this includes the navigation between presenting either an overview of (sub-)graph structures or individual nodes and clusters with their attribute data. In contrast to the reduction of visual elements by aggregation, the **filter** [AES05; Yi+07; BM13] task focuses on eliminating entities that do not fulfil a condition or fit into a specific range. This may incorporate the adaptation of visual attributes of entities or an actual visual removal of elements, e.g., removing all nodes presenting people of male gender in a social network.

The **connect** task [Yi+07] is specifically relevant for graph visualizations as it describes many essential tasks regarding edge connections. For example, highlighting association and relationships [Yi+07] of entities like connected genes of a selected disease is part of this group. This is very related to the process of identifying adjacent nodes which Lee et al. [Lee+06] consider a low-level task. However, in this overview this task was removed since there are diverse possibilities to achieve finding adjacent nodes, e.g., following the connecting edge (explore [Yi+07]) or pulling in a node's neighbors to reduce the visual distance (rearrange [Yi+07]).

The following chapters will refer to these presented interaction tasks to relate existing work and designed techniques to the associated task that they support. General interaction tasks for exploration (E) will refer to the presented category names defined by Yi et al. [Yi+07] (e.g., E [SELECT]) which are refined when necessary with graph-specific tasks (G) from the taxonomy of Lee et al. [Lee+06]. While the permanent manipulation of the visual representation and the process of data editing is not at the forefront of this thesis, basic manipulation possibilities will be mentioned where appropriate. Therefore, this thesis will refer to our overview of basic manipulation tasks [Gla+15a; Gla+15b] (M) which considers *add* (create or insert), *update,* and *delete* operations on the previously listed individual components of the graph. An overview of the tasks E, G, and M can also be found in Appendix A.1.

2.3.3 Interactive Graph Creation, Exploration, and Manipulation Using Natural User Interfaces

There are a lot of different interactive techniques for graph exploration and manipulation using traditional user interfaces (e.g., [Wat06; MJ09; TAS09; Dör+12; FD13]), mostly applying mouse and keyboard interaction. This section however focuses on the application of natural user interfaces and novel display environments to graph visualization considering specifically interaction aspects.

Frisch et al. [FHD09; FHD10] explore the capabilities of pen and touch on interactive surfaces for graph creation and manipulation (M [ADD/ DELETE NODES AND EDGES]). In a user-elicitation study [FHD09], they investigate suitable gestures for basic

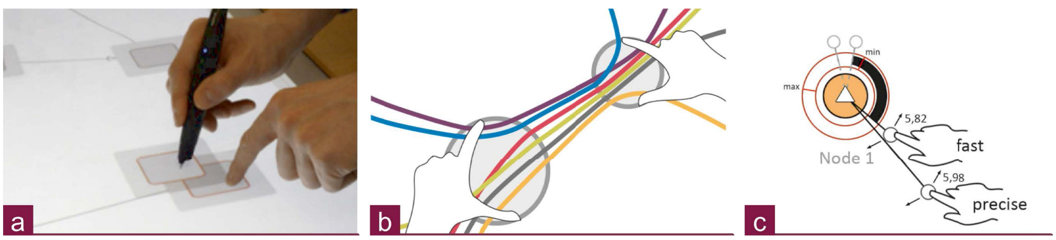


Fig. 2.10.: Touch and/or pen interaction can be used for graph creation and manipulation (a, [FHD10]), to change edge routing [Ric+12], or adjust node attributes [GT14].

Fig. 2.11.: Node details can be presented according to the user's position in front of the screen (a, [Leh+11]). This setup also supports multiple users exploring the graph (b, [PBC17a]) where details can be provided on additional displays (c, [CLM12]).

and advanced interaction tasks including creation of nodes and edges as well as copying sub-graphs. As a result, an interactive system for UML diagram manipulation [FHD10] was developed where manipulation is made possible through the very direct use of pen and touch interacting with the components of the graph (see Figure 2.10a).

Focusing more strongly on aspects of exploration, Schmidt et al. [Sch+10b] develop a set of touch-enabled tools that focus on the design of comprehensible and physical metaphors allowing the pinning, strumming, and plucking of edges to remove clutter and improve understanding of the topology of the graph (E [CONNECT]; G [ADJACENCY]). For the same purpose, Riche et al. [Ric+12] present four interactive techniques to control link curvature which support the local bundling or fanning of edges to improve visibility of routing (see Figure 2.10b), as well as magnets extending the pinning principle from Schmidt et al. [Sch+10b] and stretching of edges for labeling purposes.

Aiming to integrate both exploration and manipulation tasks, Gladisch et al. [GT14] present a touch interface that focuses on supporting changes in the visual encoding (E [ENCODE]) and manipulation of data attributes (M [UPDATE ATTRIBUTES]) by interacting with individual nodes in the visualization and pulling out interface elements at the node for fluid interaction (see Figure 2.10c). In a follow up work, Gladisch et al. [Gla+15c] investigate the combination of node-link and adjacency matrix representation for graph editing (M [ADD/ DELETE EDGES]) on a mobile device taking into account the advantages of each representation.

Applying physical navigation for graph exploration, Lehmann et al. [Leh+11] track a user's position and head orientation in front of a large vertical display (see Figure 2.11a) to visualize parts of a graph at different levels of detail (E [ABSTRACT&ELAB.]). Defining regions of interest based on distance (similar to previous work [VB04]) and an alternative detail lens technique, they examined general aspects of physical navigation in front of the display including readability and ease-of-use as well as information overload that occurred when providing too many details in both focus and peripheral vision. Roberts et al. [Rob+12] explore graph visualizations on large stereoscopic display. Due to the difficulties of reading text in 3D, they investigate the use of additional mobile devices for reading textual information associated with a 3D graph structure. Similarly, Cheng et al. [CLM12] discuss presenting different overview and detail views on mobile devices in conjunction with a large vertical display where multiple users can look at individual parts of a graph visualization (see Figure 2.11c).

Focusing on collaboration between pairs of users, Prouzeau et al. [PBC17a] investigate collaborative selection of nodes and their neighbors to identify common connections (G [CONNECTIVITY]). They compare a set of techniques for selection including propagations of selections comparing their influence on the users amount of movement and coordination (see Figure 2.11b). Again using a traditional graph task, Prouzeau et al. [PBC17b] observe multiple users for path-finding on a large vertical display comparing the effectiveness, efficiency, and communication between users to that when using desktop setups.

2.4 Summary

This chapter presented the research context of this thesis discussing the value of natural user interfaces for information visualization. By presenting the variety of interaction modalities for natural user interfaces, it discussed the advantages and challenges regarding novel interactions including the possibilities of more direct interaction with the data, problems of recognition and recall, the movement of individual users and strategies of their collaboration for data analysis which are yet to be fully understood and supported by practical and feature-rich systems.

In addition, this chapter considered application cases of graph visualization and node-link diagrams specifically. It summarized existing graph representations and focused on the various tasks of graph interaction that need to be supported by tools and systems. Regarding these tasks, solutions from related work that apply natural user interfaces to graph analysis are presented and shortly described. However, these are small independent solutions for very specific and individual problems. In the next chapter, the focus lies on magic lenses as a specific tool for a wide range of diverse problems and tasks in visualization.

3 Interactive Lenses: Theory, Categorization, and Application

Contents

Magic lenses have first been introduced as interface tools by Bier et al. [Bie+93] in 1993. Their interactive properties are part of the Toolglasses concept which describes a transparent layer of widgets and filters that are moved above the visualized content to manipulate a selected part of the view. While the Toolglass widgets are used to permanently manipulate the view (e.g., assign new colors), the Magic Lens filters support the users in their interaction by temporarily altering the view in the local area of interaction, e.g., by showing a wired frame of overlapping elements for improved target selection. Through this temporarily limited effect on the visualization, they support the adaptation of an interactively definable focus region in the moment while keeping the context intact. Bier et al. [Bie+93; Bie+94] apply this concept to a variety of example use cases, such as magnifying visual items, adjusting graphical properties, querying precise data values, or dynamic filtering. Today, magic lenses as tools have been extensively researched and extended in terms of possible functions for manipulating the view. Especially in both information and scientific visualization, more than 40 different functions have been introduced to temporarily and locally add, remove, or reconfigure parts of the visualization depending on the users' current needs and goals [Tom+17]. They have proven to be useful tools for a wide range of interaction tasks supporting the exploration and even the manipulation of visualized data sets.

This chapter focuses on magic lenses as a tool for data exploration starting with its general principle and definition (3.1) including a discussion of its relation to *overview and detail* and *focus & context* techniques. Secondly, section 3.2 presents an investigation into the characteristics and properties of magic lenses aiming at understanding the parameters that describe the magic lens and which a user may need to adjust and manipulate to configure it. Concentrating specifically on individual tasks and goals of the user, section 3.3 gives an overview of existing lens functions categorizing them by the task and data type for which they were designed. Finally, section 3.4 presents the basic interaction techniques that have been introduced to work with lenses, highlighting existing limitations and gaps many of which will be addressed in this thesis.

Parts of the definition, description, and survey of related work concerning magic lenses have previously appeared in the following survey publication. However, this chapter extends this work and explores in a much wider extent the principles and categorization related to graph lenses.

Christian Tominski, Stefan Gladisch, **Ulrike Kister**, Raimund Dachselt, and Heidrun Schumann. 2017. Interactive Lenses for Visualization: An Extended Survey. In *Computer Graphics Forum (CGF), Vol. 36, No. 6* (September 2017). pages 173–200, ISSN: 1467-8659.

3.1 Definition and Description of Magic Lenses

Magnification lenses as they also exist in the analog, physical world are tools for looking at information spaces at different levels of detail. This principle has been adapted to the digital world. It is part of a variety of concepts that present more information (i.e., additional details or a larger representation) in addition to a high-level context view for orientation in the larger information space. The most prominent examples are *overview and detail* and *focus and context*.

In large information spaces, problems often arise as users can either see the whole structure but lacking details (zoomed-out) or seeing individual details but losing the overview and structural information of the entire space (zoomed-in). **Overview and detail** is a technique that presents both high-level information as well as more detailed, low-level information for a specific region of interest [CKB09]. This is accomplished by two spatially separate views that are provided to the user (see Figure 3.1a). Feedback on the location of the low-level view is often incorporated within the high-level one. In comparison to interfaces without separate overview, the technique is much preferred by users [HBP02]. However, studies have also shown that the separation of these views results in additional effort and split attention up to even reducing task completion time [HBP02; Bau+02]. Applying the principle of overview and detail but reducing these disadvantages, **focus and context** techniques integrate these separated views into one single dynamic display [CMS99] (see Figure 3.1b). Thereby, they present the focus "seamlessly within its surrounding context" [CKB09] and decrease the additional effort by using our natural abilities regarding peripheral vision [Bau+02]. While the manner of this integration can vary, a visual distortion is often applied to incorporate the focus view into its surrounding context. Most often applied example techniques are bifocal views and fisheye distortions (e.g., [CLP04; SB94; Tom+06; Alv+14]). However, this distortion brings its own disadvantages, as it decreases the ease of target acquisition [ACP10; Alv+14] due to the fact that content within that distortion is repositioned during interaction, i.e., movement of the focus region.

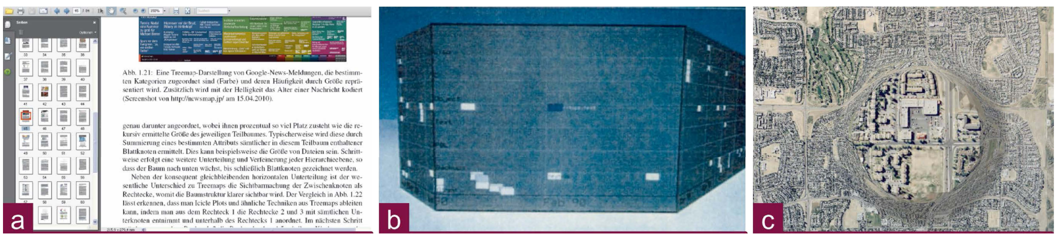


Fig. 3.1.: While for *overview and detail* the two views are separate (a, [PD10]), *focus and context* incorporates one view into the other (b, Perspective Wall [MRC91]). Distortion can further improve this integration (c, magnification lens [CLP04]).

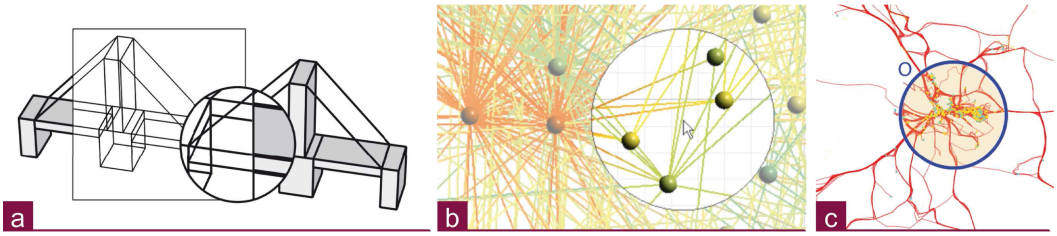


Fig. 3.2.: Magic lenses are operators that manipulate a region of interest, e.g., presenting its wire frame (a, [Bie+93]), removing irrelevant edges (b, LocalEdgeLens [Tom+06]), or filtering specific trajectories (c, TrajectoryLens [Krü+13].

Similarly to these principles, magic lenses in their most basic form for magnification provide interactive, movable "display regions that overlay the default window" [CKB09] by providing the more-detailed region of interest on top. As such Cockburn et al. consider them an *additional view* which is separated on the z-plane from the original view, resulting in a definition of magic lenses as an overview and detail technique. For general magnification this might very well be true, as the views are strongly separated if no additional steps for integration are taken, e.g., fisheye distortion or transparency (see SigmaLenses [PA08]) to merge both views at the lens border. However, magic lenses as actual interface tools were introduced as part of the concept of Toolglass widgets [Bie+93], which are see-through, transparent interfaces where the lens is one component on a Toolglass sheet layer between cursor and application [Bie+94]. In that context, magic lenses are the visual filters that *modify* the presentation of objects "to reveal hidden information, to enhance data of interest, or to suppress distracting information" [Bie+93], thereby extending the lens to be more than a tool for pixel manipulation and magnification. As part of a transparent, see-through layer, the lens is placed on top of the content, but it is not a separated view, it manipulates the content underneath (see Figure 3.2). As that, it can be moved flexibly around the information space to manipulate a region of interest. This interactivity is an essential part of the concept of lenses as it allows the flexible definition of focus and context region.

While Toolglass widgets as presented by Bier et al. [Bie+93] support permanent manipulation of the content, the magic lens filters they describe are tools that are *transient* and only temporarily alter the visual presentation to support interaction, e.g., by providing an enlarged area or wire frame representation to ease target selection. As transient visualizations they are characterized by being immediate (i.e., creating instant involvement), temporary, and both spatially and semantically close to the focus of attention and region of interest [JH07]. These principles have been adapted for a whole range of application cases. Especially in visualization, a variety of magic lenses have been introduced to temporarily adjust the visual representation of data to support understanding the presented data and ease interaction. Lenses are used to show additional information or coherences for a local region of interest

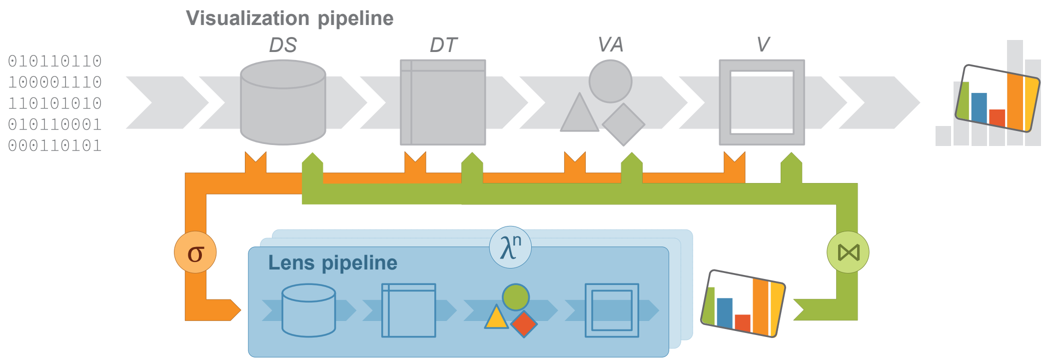


Fig. 3.3.: Magic lenses add an additional pipeline to the traditional visualization pipeline consisting of lens selection σ, lens function(s) $\lambda^{(n)}$, and join $\bowtie$. [Tom+17]

which might be overwhelming when presented from the start or all at once in the entire visualization. Further, lenses can reduce clutter and emphasize interesting information in a region of interest (e.g., Figure 3.2b).

In relation to visualization and considering the previously discussed aspects of interactivity as well as the spatial and temporal limitations of the effect, this thesis is based on the following definition of magic lenses (as published in [Tom+17]):

> *An interactive lens is a lightweight tool to solve a* localized *visualization problem* by temporarily *altering a selected part of the visual representation of the data.*

Taking the visualization pipeline [CMS99] as a foundation, the magic lens can be described as an additional pipeline parallel to the standard pipeline (see Figure 3.3) which acquires data from one of the transformation stages through the **lens selection**. It then processes this selection in one or more stages of the separate lens pipeline according to the **lens function**, possibly even applying multiple functions, and finally merges the result of the transformation back into the standard pipeline applying the **join**. In the following, these steps are described in more detail:

Lens Selection σ The spatially-limited, local input area on which the lens is placed to gather more information describes the lens selection. It is the user's region of interest and defines the input for the lens function. Depending on that function, the selection may have to occur at different stages of the visualization pipeline, either selecting data from the data tables, data primitives from the visual abstraction, or addressing pixels in the view.

Lens Function(s) $\lambda^{(n)}$ Originally described as the operator of the filter [Bie+93], the lens function describes the lens' effect, a mapping from input to output space. It processes the selection through the stages of its visualization pipeline to transform and manipulate the content. The resulting effect may be influenced by function-specific parameters, e.g., the zoom factor of a magnification. Exemplary lens functions and their parameters are described and categorized in section 3.3. Additionally, the region of interest can be manipulated by multiple lens functions. While existing works mostly focus on one single lens function, the benefits of the combination of lens functions have been discussed early on [Bie+93; Fox98; TFS08]. Up to now – if at all, the combination has been used with a fixed set of lens functions, e.g., the Composite Lens [Tom+06] is made up of three specific lens functions. A flexible combination of lens functions requires managing the lens functions' order and modifications. Different solutions for this management will be summarized in section 3.3.3.

Join ⋈ Finally, the result from the lens pipeline is merged back into the original visualization pipeline. Manipulations in the early stages of the pipeline can be integrated into the following stage, e.g., adjusting the property of a data point or visual properties of the visual abstraction before further processing it in the original pipeline and creating the merged view. Alternatively, the lens pipeline may be fully sufficient, processing all stages up until creating the pixel representation that is placed on the original visualization (cf. *overview and detail* above). By original definition, this view is then presented at the location of the lens selection. However, some research works have introduced a separation of the input area of the lens selection from the output area defined by the join, presenting the output beside the original input either to make use of the entire screen space [KAP11; Tom+12] or to be able to move output areas next to each other for comparison [BHR14] (see Figure 3.4b). Furthermore, some techniques introduced as lenses manipulate elements outside the lens selection, e.g., accessing neighbors of graph nodes in the selection (see Figure 3.4c). As a result, the *effect extent* of these lens functions is not restricted to the area described by the lens' geometric properties, which define the limits of the lens selection, but influences the global visualization. Section 3.3 will include some of these lenses as long as they conform with parts of the definition of lenses or generally adhere to the look and feel of a lens.

Fig. 3.4.: Traditionally, lenses only affect elements inside their boundary, e.g., [FG98] (a). However, their output can also be decoupled and placed close for comparison [BHR14] (b) or may affect elements outside the lens [Tom+06] (c).

3.2 Geometric Properties Defining the Lens Selection

The lens selection is spatially limited by the geometric features that represent the lens. The most fundamental interaction with lenses lies in manipulating the region of interest the lens affects by manipulating these geometric properties. These properties are dependent on the dimensionality of the space. As previously discussed, this thesis focuses on information visualization of 2-dimensional space where lenses themselves are 2D selections. As a result, the geometry of the lens defines a 2-dimensional spatial boundary that clearly separates the interior from the exterior, i.e., elements in the lens selection from elements outside the lens selection. For processing the data, this form needs to be precisely defined. However, the visual representation of the lens ca be more integrated and merged with the global visualization (see *join* above). Generally however, the lens is represented by a visible component describing the border. The properties describing this geometric form of the lens are position, shape, orientation, and size. In the following chapters, these will be called *lens properties* as they are generally independent from the lens' currently selected function or content.

Position Placing and fine positioning of a lens above the content that is to be explored is one of the most essential interactions, defining the current region of interest. For real-world magnification lenses this is even the one major parameter to be changed. When changing the position of a magic lens, i.e., moving the lens around the context from a starting to a target location, automatic adaptations can be made, e.g., applying its current velocity and acceleration to improve target acquisition [ACP10] or guiding the lens by snapping along a route [Alv+14]. Further, the center of the lens can play an important role when it comes to the lens function, i.e., for fisheye distortions, affecting a central data point, or re-arrangement of data representations. Defining the center is easy for circular lenses but may become difficult for arbitrary shaped lenses.

Shape Theoretically, there is no restriction concerning the shape of the lens, as long as it is a closed form with a clear definition of inside and outside. The shape should be chosen to reflect the requirements of the application either concerning the data, the display and interaction capabilities, or the lens function requirements, e.g., circular for circular layout rearrangement. While it is necessary to clearly define the shape for the lens selection, the join may incorporate a merge of both views, e.g., using transparency to blur the lens' shape. Sigma Lenses [PA08], for example, use time and translucence to smooth transitions between the filtered content and the surrounding non-modified context. Despite this flexibility, most

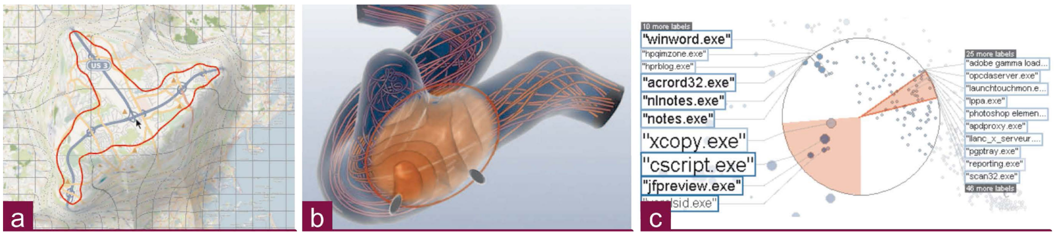


Fig. 3.5.: The geometric properties of the lens can be adapted to fit the data underneath: shape (a, *JellyLens* [Pin+12]), orientation (b, *FlowLens* [Gas+11]), and size (c, [BRL09]).

lenses presented in previous research are of circular form (e.g. [Bie+93; SCG10; ZDS12]), possibly because it resembles the metaphor of a physical lens, but also as it is a very clear and simple form with defined center. Further, rectangular shapes have been used extensively [SFB94; CLP04; Hei+11] matching the shape of windows and views within their visualization applications. While these are fixed primitive shapes, lenses can also have arbitrary contours. However, very few existing solutions adapt their shape dynamically. When dynamically adapted, the lens shape can either be explicitly determined by the user (e.g., with a sketch) [Bro+13; KE12] or by the underlying data [Ion+13; Pin+12]. For example, Pindat et al. [Pin+12] implemented a *JellyLens* for map exploration where the lens shape adjusts according to the content on the map below (see Figure 3.5a). Furthermore, a deliberate change of the shape of the lens was suggested in the work of Kim and Elmqvist [KE12] where flexible, cuttable transparencies are used as magic lenses. Finally, using lenses of varying shape in the same application can further lend itself to distinguish different lenses (and their lens functions) [SSD09].

Orientation The lens selection can further be influenced by changing the orientation of the lens. By rotating the shape, users can adapt the lens geometry to fit the focus region they are interested in. Clearly, rotation has no influence on the geometry for circular lens shapes. Similarly, shapes that adapt to the data below may also show no effect on simple change in orientation. However, the rotation could still be used to switch through or manipulate the shape-defining function in these cases. In contrast to this use in 2-dimensional space, orientation may be much more essential for 3-dimensional data sets where it can be used to define specific slices in the volume representation, e.g., in medical data [SSD09; Gas+11] (see Figure 3.5b).

Size The lens' size refines the extent of the lens selection around its position. Hence, by repositioning and scaling it the user can adjust the lens to his or her current region of interest. Depending on the selected shape, the lens' size might be changed uniformly or independently for individual dimensions or parts of the shape. The extended excentric labeling lens of Bertini et al. [BRL09] automatically adapts its size according to the data density, thereby ensuring to not exceed a certain number of labels to be shown (see Figure 3.5c). While transparencies and cardboard may allow flexibility in shape (see above), it is a limitation of these tangible representations of lenses that they cannot be manipulated dynamically during interaction.

3.3 The Diversity of Lens Functions and Their Application in Visualization

The lens function represents the operator or mapping that defines how the lens manipulates the view. It is often also referred to as the lens filter or lens effect. While introducing the concept of virtual magic lens filters, Bier et al. [Bie+93] describe a whole range of lens functions of varying purpose and effect. Among them the possibility to preview changes of a property before editing, improved target selection by applying a wire frame lens (transparency of objects) or rescaling/shrinking items (for overlapping objects), and adding a new visualization layer, e.g., showing a heat map representation of a specific property on the mapped data. The authors extended these ideas in their following work [Bie+94] focusing on presenting even more additional items, e.g., by showing additional markers (representing spaces and tabs) between text characters as handles for manipulation, as well as presenting recently deleted objects (ghost objects) for undo operation. Even additional control panels have been included as a lens function, e.g., showing individual controls per object to manipulate and set properties, which is at the very border of our definition of lenses for visualization. Stone et al. [SFB94], with partially the same group of authors, add details on maps and reordering of image components to these examples, showing the rich possibilities that lenses can provide for both exploration and manipulation tasks in visualization and other application cases.

Based on these first examples, the set of lens functions has been extended, adjusted, and applied to a whole range of application cases in visualization. In our survey [Tom+17], we[1] analyzed more than 50 different works of visualization and interaction research that propose lenses with diverse functions. To consistently describe and categorize these lenses, we provide taxonomy dimensions based in part on the initial taxonomy of Bier et al. [Bie+94]. Of these dimensions, the following relate to the lens function:

- data type
- user task
- effect class
- effect extent

[1]'We' in this chapter relates to the author Ulrike Kister, as well as Christian Tominski, Stefan Gladisch, Raimund Dachselt, and Heidrun Schumann as co-contributors on this research.

In the following, the categorization of lens functions is described in more detail by characterizing these dimensions as well as some exemplary lens functions. Afterwards, section 3.3.2 focuses specifically on graph-related lens functions and their parametrization, followed by a discussion of the challenges of lens function combination.

3.3.1 Categorization of Lens Functions

Lens functions are created to support data analysis by solving limitations of specific visualization techniques and aiding the user in reaching their goals. Therefore, they can be categorized by the data type and interaction tasks they address. Furthermore, they vary in the class of effect they provide to solve this visualization problem and how much this effect extents in regard to the focus area within the lens or the entire, global visualization context.

While multiple data types can be effectively portrayed in one visualization, lens functions are applied to a specific data type. They can hence be categorized by the data type they process. We classify lenses according to the following list of **data types**:

- temporal data
- geospatial data
- flow data
- multivariate data
- graph data
- text and document data

Note that in contrast to our survey on interactive lenses [Tom+17], this selection does not consider three-dimensional data representations as they are not in the focus of this thesis work.

When analyzing this data, lens functions are designed to solve a problem by supporting and enabling specific tasks for the data analysis. Section 2.3.2 already presented a range of tasks in relation to visualization and graphs specifically. For the following categorization, we apply the established task taxonomy for exploration by Yi et al. [Yi+07] (E) to categorize lens functions according to the **user's interaction task**. However, other previously presented task taxonomies could equally be used to describe the purpose of the lens function and thereby this dimension. Naturally, selection is part of any lens (see *lens selection* in 3.1). However in this case, we consider *select* to mean any function that aids the subsequent selection of individual elements or groups. We further add general manipulation (M) to highlight the few existing lens examples that address permanent editing. This results in the following task categories:

- E [SELECT]
- E [EXPLORE]
- E [RECONFIGURE]
- E [ENCODE]
- E [ABSTRACT/ELABORATE]
- E [FILTER]
- E [CONNECT]
- M [MANIPULATE]

To accomplish the support of the interaction task, the lens function manipulates the view in different ways. While Bier et al. [Bie+94] define an operation class for the see-through tool interface, we adapt this property and refine it for lens functions to describe the **effect class**. This dimension describes the way the lens function manipulates elements in the lens selection, either by *suppressing*, *altering*, or *enriching* elements or properties of the visual representation within the visualization. While the general lens effect is transient for all exploration tasks, this needs to be reconsidered when also incorporating manipulation. As such the actual effect classes remain the same, but the manipulation of the view may become permanent on confirmation.

As already discussed when describing the lens' *join* stage (cf. section 3.1), the **effect extent** of a typical lens function only applies to the elements within the lens selection (i.e., effect extent is *lens interior*). However, in some cases this manipulation will indirectly effect elements outside of the lens selection: For example, changing the position of graph nodes in the lens interior affects the routing of their edges to nodes outside of the lens selection (e.g., Layout Lens [TAS09]). Similarly, the lens can explicitly affect the position of neighbors of the graph nodes in focus (e.g., Bring Neighbors Lens [Tom+06]). In these cases the effect extent is considered with *side effects*. This is especially relevant when considering multi-user contexts where the side effect could influence another user's current focus region and thereby create conflicts between user interactions. Finally, when creating a completely new view separate from the lens selection with a copy of the original content and/or additional elements, the effect extent is called *separate view*.

The following describes and classifies a set of examples to show the diversity of lens functions and the application of the taxonomy dimensions. These functions were selected for their range in interaction task, effect class, and effect extent.

Magnification on Diverse Data Types The ***Document Lens*** (*data type:* text and document, *user task:* abstract/elaborate, *effect class:* alter, *effect extent:* side effect) by Robertson and Mackinlay [RM93] was developed in 1993 at roughly the same time as Bier et al.'s magic lens proposition. It presents part of a document at large scale while the surrounding text pages are pulled toward the lens and aligned in perspective forming a truncated pyramid (see Figure 3.6a). While generally a type of magnification lens, this lens of rectangular shape focuses on presenting text in its

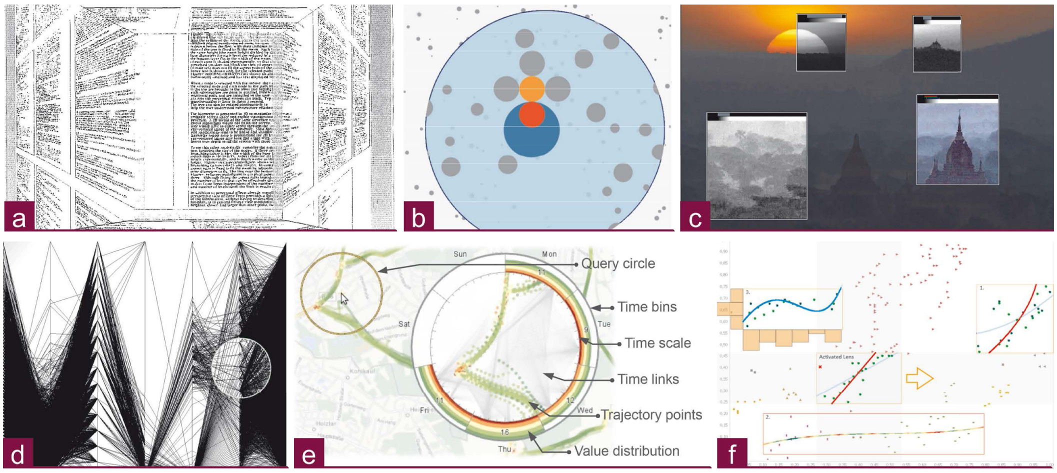


Fig. 3.6.: Magnification is the most common lens function (a, Document Lens [RM93]) and often used to improve selection (b, Bubble Lens [MW14]). But lenses may also re-encode data to show hidden information (c, Color Lens [EDF11]), suppress data objects (d, SamplingLens [EBD05]) or enrich the region of interest, e.g., with aggregated data, additional details (e, Time Lens [Tom+12]), or abstracted data (f, Regression Lens [Sha+17]).

document context by also affecting elements outside its boundaries as the remaining pages are distorted to be still readable. This supports understanding of the structure of the document around the text in focus.

These types of magnification lenses have often been used to improve on imprecise selection. The ***Bubble Lens*** (*data type:* multivariate, *user task:* select [& abstr./elab.], *effect class:* alter, *effect extent:* lens interior) proposed by Mott and Wobbrock [MW14] was designed to specifically support selection of small targets. As a further example of magnification lenses, this lens does not only enlarge the visuals but also the motor space in which the user interacts making it easier to select small items. The magnification is restricted to the lens interior (see Figure 3.6b) and is automatically adapted according to the size and number of small data items in the lens selection.

Diverse Interaction Tasks within the Lens Interior Elmqvist et al. [EDF11] present a ***Color Lens*** (*data type:* various, *user task:* encode, *effect class:* alter, *effect extent:* lens interior) for previewing changes for the entire visualization in a selected region. The lens function encodes the underlying representation using another color scheme (see Figure 3.6c). This also includes changing the color scale within the lens region to visualize relative changes that would otherwise be hidden in the global scale. The preview is presented only within the confines of the lens geometry while the remaining representation is left untouched. The Color Lens is shown to be suitable for a whole range of application cases and data types including node-link graphs, multivariate data representations, and geospatial data visualizations.

Equally designed to reveal hidden patterns, the ***Sampling Lens*** (*data type:* multivariate, *user task:* filter, *effect class:* suppress, *effect extent:* lens interior) by Ellis et al. [EBD05] reduces clutter in scatter plots or parallel coordinate plots by sampling from the data resulting in a more sparsely populated view. This supports filtering to better understand trends within the data since relations can be made visible that were hidden in the cluttering (see Figure 3.6d). The lens effect suppresses content and is only visible within the lens interior.

Providing Aggregated Data with Varying Effect Extent The following two lenses enrich the existing visualization with additional, aggregated data to locally present a refined view with more details. ***Time Lens*** (*data type:* geospatial, *user task:* explore & encode, *effect class:* enrich, *effect extent:* separate view) by Tominski et al. [Tom+12] presents another encoding of the data by showing aggregated information for the selected area. The lens content is presented completely separated from the original selection with only an additional visual element to clarify the association (see Figure 3.6e).
Another example of presenting aggregated data and calculations in place within the data is the ***Regression Lens*** (*data type:* multivariate, *user task:* explore & encode, *effect class:* enrich, *effect extent:* lens interior) by Shao et al. [Sha+17]. It presents a line chart as an overlay within the rectangular lens while keeping the original scatter plot data points in place (see Figure 3.6f). Instead of the usual considerations of the global distribution in regression analysis, the line chart presents the best model for the distribution of the local selection and hence can present inherent patterns within the data.

3.3.2 Lens Functions for Graph Analysis

As this thesis applies the developed principles for lens manipulation to the specific application case of graph exploration, this section describes existing lens functions addressing graph visualizations in more detail. While the focus lies on lenses specifically designed for graph data, it also includes functions which – though not proposed for graphs – are closely related and suitable for application in graph visualizations. The lens functions are discussed in regard to the addressed problem and applied solution, including providing enlarged views and detailed information, reducing edge clutter, concerning node connectivity and relations, and finally manipulation within graph structures. Beside the description of the lens function, this section also specifically identifies parameters that define or extend their effect. To actually apply these lenses to varying data sets and application cases, these lens function parameters should be adaptable to fit the lens to the user's current needs and goals. An overview of all discussed graph lenses in regard to their categorization according to user task, effect class, and effect extent is given in Table 3.1.

Technique	User Task								Effect Class			Eff. Extent		
	select	explore	reconfigure	encode	abstract/elab.	filter	connect	manipulate	suppress	alter	enrich	interior	side effects	separate view
Fisheye Views [SB92; SB94]					x					x		x		
Labeling Lens [FP99; BRL09]	x	x			x						x		x	
Fuzzy Lens, Base-Pair Lens, Ring Lens [Bak+02]			x	x	x						x	x		
EdgeLens [WCG03]			x							x			x	
Abstraction Lens [HW04]					x						x		x	
Local Edge Lens [Tom+06]						x			x			x		
Bring Neighbors [Tom+06], Layout Lens [TAS09]			x				x			x			x	
NodeTrix [HFM07] as lens			x	x							x			x
Bring & Go [Mos+09]		x	x				x			x			x	
Network Lens [JDK10]				x							x	x		
Push Lens [Sch+10b]			x							x			x	
Expand Tangible View [Spi+10]			x		x						x	x		
Bundled Neighbors [LAM10]							x		x			x		x
Mole View [HTE11]			x							x		x		
EdgeAnalyser [Pan+11]				x	x					x	x		x	
Hypergraph Lens [UČK12]		x				x			x			x		
Edit Lens [Gla+14]								x	x	x	x		x	

Tab. 3.1.: Categorization of existing graph lens functions and lens-like principles for graphs (sorted by year).

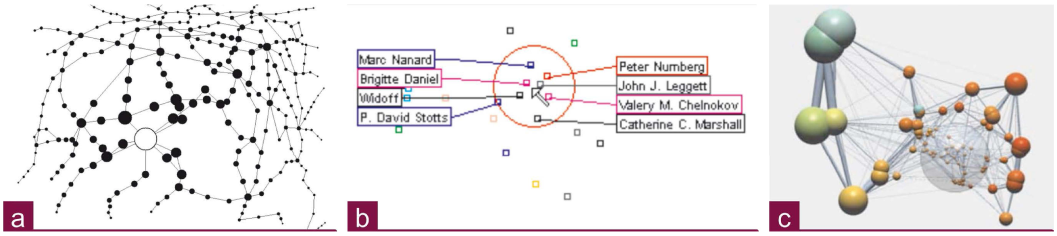


Fig. 3.7.: For graph analysis, fisheye magnification [SB94] may highlight connections to a node in focus (a). Lenses can present details and labels [FP99] around the nodes (b) and expand inherent node structures [HW04] (c).

Providing Details or Additional Information

As the predominant lens function for lenses, independent from application case, the magnification lens is also suitable for graph applications. Not only do magnification lenses support better visibility of individual elements but in the context of node-link representations of graphs, they can improve the visibility of structural information and connectivity between nodes. In 1992, Sarkar and Brown [SB92] discussed the advantages of magnification and fisheye views specifically for the application case of graphs proposing a ***fisheye lens*** approach (see Figure 3.7a). Unlike fisheye views for image representations, Sarkar and Brown take into account structural information of the graph so that position, size, and level of detail of objects are based on user-specific functions of an object's distance from the focus and the object's preassigned importance in the global structure. Tominski et al. [Tom+06] apply the same principles while also presenting a more spatially-restricted version of the effect by integrating it into a graph lens. In their work, the fisheye distortion is used as part of a lens to spread previously pulled-in nodes that are positioned too close together. Here, it is very important to be able to influence the amount of distortion, i.e., the *zoom factor*, applied to the local graph to balance the overlap of nodes in focus. Especially when enlarging elements in densely populated areas overlapping can occur and clutter can critically reduce the positive effect of the lens function. When considering the distortion of the presentation space according to Carpendale and Montagnese [CM01], additional factors can be incorporated to manipulate the presented view and distortion. Varying drop-off functions can be used to limit the space needed for integrating the focus with the context visualization and increase the visibility of details in the center, e.g., using an inverse cosine or Manhattan Lens drop-off to provide more or less information within the distorted transition. Furthermore, the lens can be separated into rings with varying degree of magnification, stretching the structure only in interesting parts to exploit the complete lens space (cf. Ring Lens [Bak+02]). To limit the clutter of elements in focus, an additional *repulsion* can be added to push nodes to the border of the lens area before manipulation and can similarly provide more room to the focus region.

While magnifying a region of interest may provide a first solution to provide more space for nodes regarded as important, it is necessary to present details about these nodes in focus. Especially for nodes in cliques or nearly complete sub-graphs, layouts can result in very little space for node description and labeling (see Figure 3.7b). There has been extensive research on the problem of labeling on densely positioned areas (e.g., [MS91; CMS94]). Applying lenses to this problem, Fekete and Plaisant [FP99] propose an interactive, transient solution by presenting a locally restricted labeling. Bertini et al. [BRL09] extend this idea creating a ***labeling lens*** that adapts its size to the density of the underlying data to keep the number of labels shown to a presentable limit. In the spirit of providing more details (similar to lenses for details on maps [SFB94]), this principle can be extended to provide additional information, such as attribute data for multivariate nodes, in place for a local region of interest. Jusufi et al. [JDK10] present small visualizations, including star plots, parallel coordinate plots, and bar charts, as glyphs for each individual node in their ***Network Lens*** to support visualization of attribute data and possibly comparison of the nodes in focus.

For hierarchical graph structures, providing more details about individual nodes may also extend from just representing multivariate data to including the inherent structures within nodes. In their work, van Ham and van Wijk [HW04] discuss the interactive expanding and collapsing of nodes by combining this effect with a fisheye lens (see Figure 3.7c). Their lens presents local sub-structures to the users without overwhelming them by expanding globally. However, the lens incorporates a *side effect* (effect extent) as it also expands connected sub-nodes of the visualization outside of the lens to clear up relations between sub-graphs. Spindler et al. [Spi+10] apply the same principle for exploring the hierarchical structure of the ACM classification represented as a tree at different levels of abstraction, expanding and spreading the inherent child nodes or collapsing them again within the lens. For hypergraphs, Ukrop et al. [UČK12] propose a lens that presents hyperedges as a result of a hypergraph actively queried for relations in a local region of interest.

Reducing Edge Clutter

As real-world graphs are rarely planar, edge cluttering is a predominant problem in graph layouting and exploration (e.g., [Lan+11; GFC05]). There are a range of lens solutions that address this problem focusing on reducing the number of edges in a region of interest either by actually removing and hiding edges or by distorting them around that region. Wong et al. [WCG03] present an ***EdgeLens*** that creates a distortion at a given cursor position to enable visibility of edge routing and clearing the area around the cursor position from edges while also spreading them slightly to distinguish their courses (see Figure 3.8a). Addressing the same problem, Tominski et al. [Tom+06] propose affecting only a specific set of edges separating the graph in

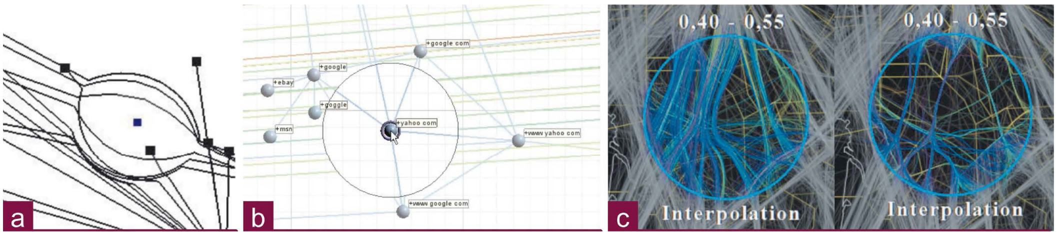


Fig. 3.8.: To reduce edge clutter and see the connections of nodes in focus, edges may be distorted (a, EdgeLens [WCG03]), removed (b, Local Edge Lens [Tom+06]), or bundled together (c, MoleView [HTE11]) by a lens.

relevant and irrelevant edges for a specific region of interest. With their ***Local Edge Lens***, the lens effects the lens interior by removing edges that are not connected to any nodes in the lens selection thereby removing edges that simply run by the area without being of relevance to the focus region (see Figure 3.8b). Visually this is quite similar to the previously discussed ***Sampling Lens*** (see Figure 3.6d on page 52), however the set of removed edges is not arbitrary but strictly defined by the nodes in focus. The lens effect can be extended by refining the criteria of relevance, e.g., taking *edge weight* into account. Furthermore, the *opacity* of the irrelevant edges can be manipulated to not completely remove edges but keep them for context. Schmidt et al. [Sch+10b] combine the ideas of distortion and relevance by creating a ***PushLens*** which removes the unconnected edges by routing them around the border of the lens leaving the focus area similar to the Local Edge Lens but with additional edge bundles around the lens.

MoleView by Hurter et al. [HTE11] introduces a whole range of local deformations incorporated into the lens principle. Generally defined for data sets with any data items that have 2D layout positions, e.g. scatter plot, pixels, or graphs, varying deformation functions are used to de-clutter a region, apply fisheye-like effects on selected data elements, or bundle graph edges. An important aspect of these lenses in comparison to the previously proposed edge-related lenses is that nodes are pushed to the border of the lens according to their attribute values. Interestingly, the same principles are also applied to edges where the edge itself and its attributes are considered for the deformation function instead of the edges' source and target nodes. Finally, these deformations are used to locally bundle edge-filled areas of a graph (see Figure 3.8c) or unbundle previously bundled graph edges for exploration. Panagiotidis et al. [Pan+11] present a similar concept with their ***EdgeAnalyzer*** where edges crossing the lenses can be grouped together using different grouping methods and are hence bundled and unbundled accordingly.

Improved Visualization of Relations and Connectivity

The reduction of edge clutter already aids the analysis of relations and structures within the graph. Beside these edge-related lens functions, there have been others that specifically address the visualization of adjacency and connectivity within the graph. Tominski et al. [Tom+06; TAS09] developed graph lenses for layout rearrangement to improve the representation of structural information: The ***Bring Neighbors Lens*** [Tom+06], later also called Layout Lens [TAS09] for generalization of the concept, has been developed to present the adjacent nodes, or neighbors, of a node of interest within the constraints of the lens. For that, the distance of focus node and neighbor is decreased in relation to the distance of the focus node from the center of the lens. When the lens is directly on top of the node of interest, all neighboring nodes are pulled into the lens so that the originally farthest neighbor is placed on the border of the circular lens (see Figure 3.9a). Its extension, the ***Generalized Bring Neighbors Lens*** [Tom+06], can be applied to multiple nodes in focus where each of these nodes is given an attraction weight depending on its distance to the lens' center. How much each nodes' neighboring nodes are brought in is calculated according to this individual factor. However, this still requires moving the lens' center on top of a node of interest to gather all neighbors of this node within the lens. An additional *attraction factor* can increase this movement towards the lens and hence support seeing neighbors of multiple nodes of interest within the confines of the lens, without having to look for moving nodes outside of the lens. The previous layout rearrangements all kept the relative distance of neighbors to their focus node intact. Moscovich et al. [Mos+09] present ***Bring & Go*** which, while used for navigation and not as a lens, is a local layout arrangement that pulls neighbors as well as neighbors of neighbors of a node in focus onto concentric rings while preserving the direction of their original location (see Figure 3.9b). This supports identifying the degree of adjacency of each node by their distance to the focus node. Lambert et al. [LAM10] adapt these previous lenses for edge-bundled graphs, highlighting the neighborhood of a certain range within the lens-like circular area while visually fading out other nodes and edges in that region of interest (see Figure 3.9c). Applying Bring & Go, the neighbors can also be brought closer together

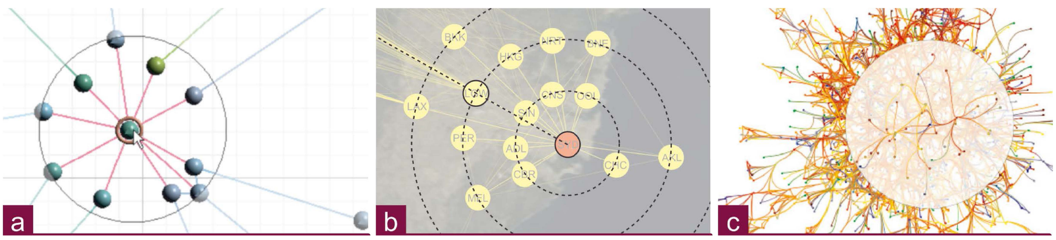


Fig. 3.9.: To identify the neighborhood of a focus node, adjacent nodes are pulled in (a, BringNeighborsLens [Tom+06]). Extending this to a further range of neighborhood, nodes can be placed onto concentric circles (b, Bring & Go [Mos+09]) while other elements are faded out [LAM10] (c).

to further emphasize their connections. Contrary to its predecessors, connections to other nodes are removed to reduce edge clutter, thereby however also temporarily reducing the connection to the context visualization. As all of these lens function require rearrangement of neighbors of the nodes in focus, they generally disregard the characteristic of a locally-restricted lens effect. However, this *side effect* (effect extent) could be eliminated by using additional proxy elements for nodes outside the lens selection instead of re-arranging the actual nodes.

As an alternative representation of graphs, the adjacency matrix is a convenient and well-established visualization for graph adjacency. All graph relations (and optionally their edge weight) are visualized in a matrix of nodes which support quick analysis and, with appropriate sorting of rows and columns, allow the visualization of relationship patterns. With ***NodeTrix*** (cf. alternative representations in section 2.3), Henry et al. [HFM07] created a solution that visualizes small sub-graphs and cliques (within an otherwise sparsely connected graph) as adjacency matrices while keeping the structure among these sub-graphs in node-link representation. While NodeTrix has never been introduced as a lens, its local characteristics seem very related to the lens principle and are suitable for adaptation to lens interaction.

Graph Manipulation

While there have been a variety of lenses for graph exploration, lenses have rarely been used for permanent manipulation of graphs. However, adding nodes to existing graph structure can often be considered a spatially-restricted problem of *local* re-arrangement of a subgraph to fit the novel addition and its edges without completely re-layouting the graph. Gladisch et al. [Gla+14] investigated this issue for a literature-curated network of genes and molecular interactions with semantic groupings. In this case, the use of the manually arranged graph is dependent on the user's knowledge and recall of the nodes' locations. As such, it is essential to keep the layout consistent when adding new nodes and edges. The ***Edit Lens*** supports this by using the lens selection as a restraint for rearrangements to fit the added node or re-layout of a subgraph. Placing the lens hence defines the rough position of where

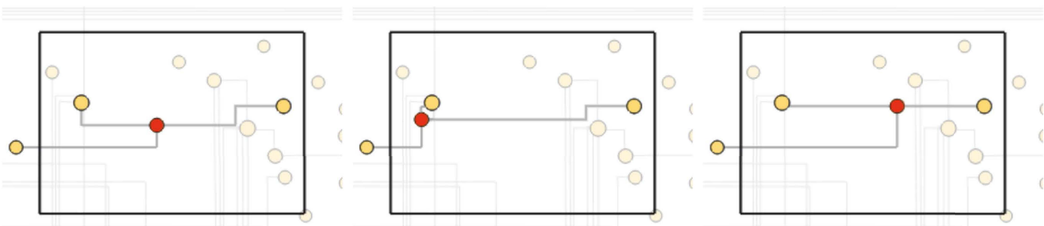


Fig. 3.10.: The EditLens [Gla+14] supports the spatially-restricted addition, update, or removal of nodes optimizing specific constraints while keeping the general layout of the graph consistent between update steps for improved recall.

nodes can be placed or re-arranged while the actual precise positioning is calculated by the computer given a range of possible *layout algorithms* or *optimization criteria*, e.g., node space, edge length, or number of edge bends (see Figure 3.10).

All these individual lens functions support specific tasks and solve specific problems. However, for a real world application where users step through a range of tasks and problems within their workflow, there is need to allow multiple lens functions and multiple lenses at once or in short succession. While this may seem easy, there are a range of problems arising with this use of multiple lenses and possible lens combinations.

3.3.3 Combination of Lens Functions

Already the early works of Bier et al. [Bie+93; Bie+94] and Stone et al. [SFB94] describe the advantages of combining multiple lens functions to accomplish a task (e.g. presenting a wire frame and magnification for improved selection) either by having multiple lens functions within the same lens or by using multiple lenses with full or partial overlap. Panagiotidis et al. [Pan+11] apply multiple lenses where the selected edges of child lenses are dependent on their parent lens selection. However, as data objects are selected globally and same lens functions are applied, the presented combination only regards the lens input selection and does not require merge of output from different lens functions in a local region. Alternatively, fixed combination of varying lenses have been proposed, e.g., the ***Composite Lens*** by Tominski et al. [Tom+06; TAS09] unites Bring Neighbors and Local Edge Lens with a fisheye effect to tidy up the lens area while pulling in adjacent nodes. However, a flexible, user-defined combination of lens functions makes it necessary to consider the possible influence of one lens effect on another. The goal of this thesis is to accomplish flexible, seamless combination of lens functions and use of multiple lenses on the same context. To make this possible, a range of considerations are necessary to make sure the lens functions act as expected by the user. In detail, the sequential application of lens functions requires propagation of the content or a bounding box of the region of interest through the stack of lenses on the canvas. Two examples for graph lenses shall illustrate the problems:

- P1 Consider a lens with multiple active lens functions: a fisheye magnification function as well as the previously described Bring Neighbors function. As a layouting function Bring Neighbor manipulates the position of nodes, pulling in neighbors of the nodes in focus. However, the fisheye effect is dependent on this position to calculate the scale factor of the individual nodes. Hence, if the fisheye function is applied before the Bring Neighbors function, only the original nodes in focus are enlarged while pulled in neighbors remain small –

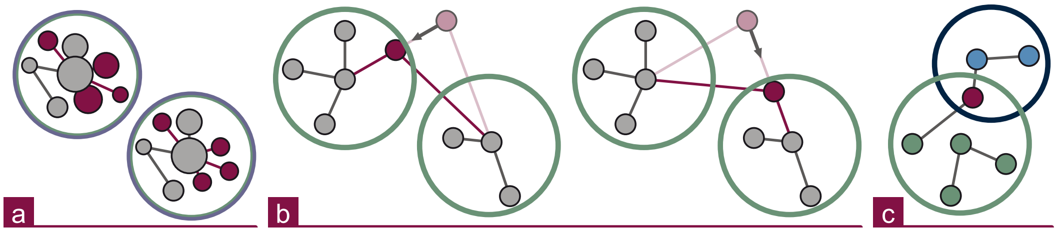


Fig. 3.11.: Various challenges arise when using multiple functions within the same lens (a), multiple lenses (b), or overlapping lenses (c) resulting in partially applied functions (a), inconsistent node states (b), and the need for merge strategies (c).

creating an unclear mixture of enlarged and non-enlarged nodes. If the Bring Neighbors function is applied first, the fisheye effect becomes visible as nodes in the center are enlarged more than elements at the border of the lens.

P2 Imagine two lenses on the context visualization, both with active Bring Neighbors function. These lenses may both affect the same node that is neighbor to nodes in focus in each lens. As a result, the lenses would logically pull the node towards themselves. Applying these lenses sequentially without considering their dependence, i.e., moving it to the first or second lens, will result in incorrect adjacency visualizations in the first lens (see Figure 3.11b-c). Now, this might appear to be a problem of lenses with *side effect* (effect extent), but the same problems occurs for partially or fully overlapping lenses with functions addressing the same attribute value, e.g., two coloring lenses (see Figure 3.11d): The first colors nodes red while the other applies blue. Users might presume the result to be an actual combination of the colors. This however requires special handling by the system.

As can be seen from the example problems, there are a range of aspects to consider when combining lens functions. Identifying these aspects, Thiede et al. [TFS08] describe the following challenges that require handling:

- order or precedence of lens function application
- write conflict, i.e., two lens functions writing on the same attribute
- dependency conflict, i.e., the dependence on another lens function's results
- read conflict, i.e., the next lens function applied could access more than one current value of an attribute (happens after write conflicts)

Order of Lens Function Stack When combining lenses or using multiple lenses on the same visualization, a possible solution is to consider lens functions as an ordered stack above the visualization. Both Tominski et al. [Tom+06] and Thiede et al. [TFS08] propose to consider the visualization pipeline for sorting these lens functions. This means that lens functions are ordered by the stage of the pipeline from which selection occurs. While this is a good step for general ordering, many of the lens functions fall into the same stage category where the problem remains. At this sub-level, the order can be enforced by manually giving each lens function a precedence value in accordance with the visualization pipeline, which can also be manipulated by the user depending on his preferences or goals.

How Do Lenses Access and Manipulate Properties? Different strategies have been proposed to handle the access of property values for the combination and use of multiple lenses. Bier et al. [Bie+93] proposed the strategy **recursive ambush** which supposes that lens functions overwrite the original drawing methods of primitives. For example, a drawLine method is overwritten by the lens function that creates red edges. While this is easy and very cost-efficient, it has a whole range of limitations: i) Not all lens functions can be described by simple drawing of primitives, e.g., clustering of graph nodes would be hard to accomplish. ii) If multiple lens functions overwrite the same drawing method, only the last applied lens function's effect can be seen, i.e., there is no real combination of effects. iii) This is a very intrusive method where the rendering is handed over to the lenses.

Another strategy presented early on by Bier et al. [Bie+93] is called **model-in model-out** where every lens works on a copy of the data model and accesses the output of all previous lens functions in the stack to add its effect to the visualization. While this is very cost-inefficient as all lens function have to manage their copy of the data model, it is a good starting point for extension and allows an actual combination of lens functions. **Reparameterize and clip** is the last combination strategy introduced by Bier et al. [Bie+93]. It proposes parameterizing the renderer to create a completely new version of the presented visualization as a whole and then clipping the image at the lens border to create the local effect. This can again be very appropriate for manipulation that focus on the rendering, working only on the last stage of the visualization pipeline. However, implementing lens functions that affect only some object, e.g., remove fill color of node with certain attribute values, is almost impossible as the renderer cannot access data model attributes. Furthermore, lens functions with *side effect* (effect extent) cannot be accounted for as the image is clipped at the lens border and cannot easily be merged into the context view. Finally, the **delegation strategy** introduced by Fox [Fox98] is based on the model-in model-out strategy and joins it with ideas from recursive ambush. Instead of working on copies of the original data, this strategy works with pointers (called *principal* that reference back to the original, called *delegate*) with the principle class overwriting the draw method of the delegate. To combine lens functions, lenses at higher position in the stack use the principal of previous lenses as delegate for their calculations. Again, lens functions that do not focus on the rendering but on attributes or structural properties of a visualization can hardly be accomplished with this approach, but the general principle of handing over the result from one lens function to the next, avoiding both read and write conflicts, will be applied in later implementations of this thesis.

3.4 Interactions for Magic Lens Manipulation

Now that the basic properties as well as possible lens functions have been discussed, these properties need to be adjusted to actually use the lens for data analysis, as by definition it is not effective without interaction. Interactive operations that have to be considered include creation and deletion of lenses, manipulations of the lens position and geometry (recall the properties discussed in section 3.2) as well as more complex operations, such as selecting and parameterizing the lens function (cf. section 3.3) or combining multiple lens functions or lenses.

Most visualization-based research focuses on the lens function, its purpose, and its advantage for solving a visualization or data analysis problem. Hence, their applications of magic lenses often rely on the traditional mouse and keyboard setting in terms of interaction, disregarding the wide variety of interactive environments that are currently arising. Inspired from the concepts of NUIs (cf. section 2.1), new collaborative environments arise and have now also been applied to magic lenses. In the following, an overview of these existing interactions in research will be given, starting with research on interaction in traditional environments but focusing on novel, natural interaction techniques. These techniques are summarized in Table 3.2 at the end of this section (page 73).

3.4.1 Lens Interaction in Mouse & Keyboard Environments

Mouse and keyboard interaction has been the most common interaction modality of the last decades. Therefore, it is no surprise that it is also the most prominent in the reviewed research on lenses. The typical environment of mouse and keyboard interaction is a one-to-few display setup for a single user. The precision of the cursor movement is the major advantage of mouse input. Additional possibilities of the mouse are given through mouse buttons and the mouse wheel, which can be used for discrete or continuous input, respectively. Keyboard input, on the other hand, is suitable for mode switches or step-wise navigation. The precise direct manipulation of the mouse is especially useful when specifying the region of interest and hence repositioning of the lens. The suitability of this mapping is evident in many of the examined research works (e.g., [Bak+02; ED06; Pin+12; Tom+06; WCG03]). As the lens position is coupled with the mouse cursor, fast repositioning becomes possible. However, when a magnification lens is used, this fast positioning at context scale can hinder target acquisition in the focus area and make pixel-precise positioning impossible [ACP10] (see Figure 3.12a).

Fig. 3.12.: Distortion and magnification makes pixel precise mouse interaction difficult (a, [ACP10]), but snapping can ease positioning (b, RouteLens [Alv+14]). Additional dialogs are used to configure the lens (c, [JDK10]).

Appert et al. [ACP10] introduce different interaction techniques to improve on the target selection when using magnification lenses. First, they present mode switching between focus and context speed by using a keyboard key. Second, they provide a continuous mapping between precision of lens movement depending on cursor speed. The third interaction technique introduces a ring as the lens border where the lens' inner region is navigated at focus scale while the outer region is navigated at context scale. In the experiments, all techniques performed better than regular lens positioning and the ring technique performed best in experiments for small targets and high magnification factors without needing the additional keyboard mode switch. For the specific RouteLens, Alvina et al. [Alv+14] use the knowledge of the underlying data, paths and streets, to predict the lens movement and counteract overshooting due to the magnification of the lens, thereby following the selected route (see Figure 3.12b).

To incorporate the adjustment of parameters other than position, mouse buttons are used to toggle a specific parameter or state [BRL09; HTE11]. Additionally, the mouse wheel can be helpful when changing continuous values, such as attribute ranges [HTE11]. However, as more and more parameters have to be adjusted for complex lens functions in visualization, graphical user interfaces are necessary. Possible controls and widgets include toolbars, combo boxes, or sliders. The mouse is then used to adjust these parameters in either global or context menus (see Figure 3.12c). Some examples can be found in Jusufi et al.'s work where a dialog box is used to create and edit the Network Lens [JDK10]. The Sampling Lens [EBD05] is another example as a slider is used for lens diameter adjustment. The Document Lens [RM93] is controlled with mouse interaction for the x-y-plane positioning and with keyboard keys for movement within the z-plane.

3.4.2 Lens Interaction on Touch-enabled Surfaces

Lenses in touch-enabled environments have often been used to improve the imprecision of the touch modality, using a magnification to reduce the effects of the fat-finger problem (e.g., [BWB06; Voi+09]). It was hence not used as a tool for data exploration, but a tool to support interaction in general. As such, Käser et al. [KAP11] proposed their *FingerGlass* approach where bimanual interactions are employed to select a region of interest with the non-dominant hand creating a lens with a magnified representation to select and translate otherwise small data objects (see Figure 3.13a). This already shows the flexibility of the direct touch interaction approach for lenses, supporting our natural two-handed, multi-touch capabilities as opposed to mouse-based interactions with single cursor and one hand only. In terms of lens interaction, especially the adjustment of the geometric properties (cf. section 3.2) becomes natural and does not require additional widgets. Manipulations (translation, scaling, or rotation) can be accomplished by direct interaction through multi-touch gestures and no menu widgets is required for these adjustments. Hence, the user can fully concentrate on the vital aspects – the lens function and its effect on the visualized data.

Applying the advantages of magnification for improved visibility in data exploration, a range of magnification lenses have been used in touch-enabled applications. The lens-like telescope views of Khan et al. [Kha+04] include a fisheye magnification that can be moved and allow GUI-based adjustment of a zoom factor. Discussing the advantages of multi-touch interaction for scatter plot visualizations, Sadana and Stasko [SS14] also present a zoom lens. This lens is created using a pinch gesture on the data and allows manipulation of the zoom level as one parameter of the lens function. Butscher et al. [BHR14] introduce *PhysicLenses* that present magnification lenses on maps (see Figure 3.13c). A long two-finger tap on the surface initializes the creation for a new lens. Interestingly, *PhysicLenses* apply multiple lenses to the same data view, decouple the resulting view and hence allow comparison of multiple foci next to each other. As the lens always remains the same size, the distance between the fingers changes the amount of magnification within the lens. The *DTLens* system by Forlines and Shen [FS05] uses multiple lenses for magnification and specifically investigates their benefit for multi-user interaction. Similarly, Bortolaso et al. [Bor+14] apply magnification lenses to maps on interactive tabletops to support multi-user interaction without losing the joined context view. *Smarties* by Chapuis et al. [CBF14] uses magnification lenses for map

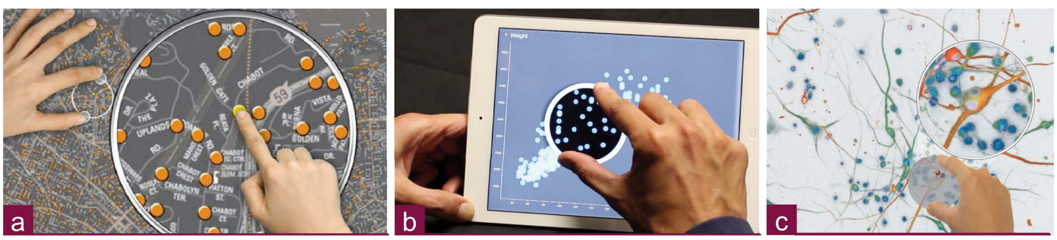


Fig. 3.13.: Lenses have often been used to improve imprecision of touch interaction (a, FingerGlass [KAP11]), but can further support data analysis [SS14] (b). However, interaction is often limited to basics for repositioning (c, PhysicLenses [BHR14]).

data presented on a large display. These lenses' size and magnification factor can be remotely controlled by pinching on a mobile device. Also adding an additional interaction modality, Ramos et al. [Ram+07] apply stylus triggered magnification lenses for small target selection and propose the adjustment of the zoom by direction-dependent crossing of the lens border. They further compare different activation techniques including automatic activation after a certain delay, an always active lens trailing the cursor, and explicit activation by pen pressure which was the best in regard to speed, accuracy, and user preference. As an extension of magnification lenses, Brosz et al. [Bro+13] present *Transmogrification* defining an area to magnify and distort using touch that can later be formed into other geometric shapes, e.g., straightened along a path, and thereby enable rectifying content for comparison or increase focused on specific areas of a view.

Very few works on touch-enabled lenses applied lens function other than magnification and hence had additional possibilities for adjustment and configuration. Schmidt et al. [Sch+10b] address interaction with node-link-diagrams by designing touch gestures for edge manipulation and incorporate the creation and manipulation of their PushLens through multi-touch interaction. The *PushLens* can be created by using three touches close in space and time and can then be repositioned and resized by dragging and pinching on the border. No other parameters of the lens can be adjusted using touch. Rzeszotarski and Kittur [RK14] also use multi-touch when positioning their *Kinetica* tools including a lens for highlighting elements of a certain criteria and optionally even limiting all interactions to highlighted elements within the lens. The lens is created using two fingers, however it remains unclear how the distinction between tools is made after gesture recognition as well as how the lens criteria is defined. Nonetheless, it became clear in the study that users applied lenses in layers on top of each other to reach their goal.

All these examples apply multi-touch interaction to lenses for their specific use case and use repositioning and resizing to manipulate the selection. However, while the lens' selection is an important initial parameter that defines the data in focus, adjusting the lens function and its parameters is important to make the lens effective and support the user's task and goals. As has been shown, parametrization of the lens was rarely used and if at all was applied for the sole purpose of changing a zoom factor. There are no solutions that accommodate flexible lens function parametrization and lenses with multiple functions on touch-enabled surfaces.

3.4.3 Tangible and Spatial Interactions for Lenses

Spindler et al. [SSD09] developed *tangible views* that are passive paper lenses on which information is projected from above (see Figure 3.14a). These tangible lenses

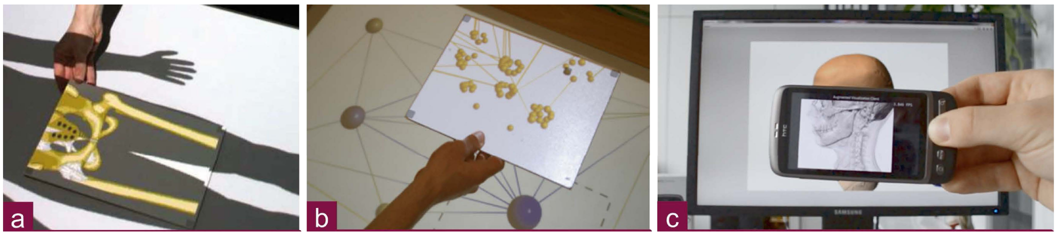


Fig. 3.14.: Spatial interactions can be used to move tangible lenses above a tabletop (a, PaperLens [SSD09]) and applied for information visualization (b, Tangible-Views [Spi+10]) or scientific visualization (c, [Sör+11]).

combine display and interaction in a single 'device' as the display becomes the actual input channel and the interaction is lifted up from the table into the third dimension. The spatial configuration of the paper lens in the x-y-plane above the table determines where the lens effect should be applied. The distance to the table is an additional parameter that can be used for interaction, e.g., to increase the zoom factor. Later on, these lenses were extended with active tablets [Spi+14a] which increases the possible resolution of visualizations on the lens. However, this also increased weight and reduced flexibility in lens shape. Interaction concepts such as translation and rotation are possible as much as flipping, tilting and shaking of the lens, which distinguishes this spatial interaction from interaction with tangibles on tabletops. Tangible views have been used for data analysis, such as graph exploration (see Figure 3.14b) where the distance to the table surface influences the level of abstraction shown in the lens [Spi+10], thus supporting the manipulation of a lens function parameter. Their advantage lies in using the physical coordination and muscle memory of the user to enable fast navigation within this interaction space and recall of certain locations by body position. Sörös et al. [Sör+11] extend this principle to scientific visualization and remove the need for external marker tracking using the build-in camera of the mobile device. As a result, each user's personal mobile can function as a magic lens showing part of the 3D visualization (see Figure 3.14c).

Kim and Elmqvist present *embodied lenses* [KE12], thin transparencies that function as layers on data similar to PaperLens [SSD09] with the additional feature of transparency and see-through properties instead of projection. These tangibles apply physical interactions, such as grasping, placing and moving for creation and manipulation of a lens. Combinations of lenses with different functions is also possible as multiple tangibles can be placed on top of each other. Applications include the exploration of layered data, e.g., data on the human skeleton and nerve system in medical imaging as well as multi-dimensional data visualizations, and tangible map interactions. Using magnification as its main function, Voida et al. [Voi+09] presented the *iPodLoupe* which offloads the lens output, a magnified view of the content, from a touch-enabled tabletop onto a smartphone allowing manipulation of

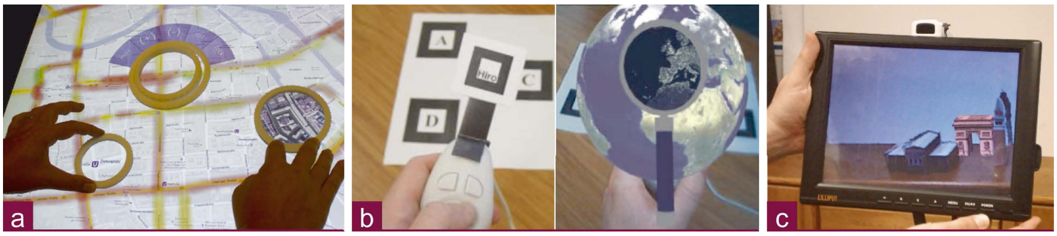


Fig. 3.15.: Tangibles can be used to to physically move and combine lenses (a, [Ebe+13]) or used in space to represent and position AR lenses (b, [LBC04]). Even hand-held AR displays could be considered magic lenses (c, [Bar+12]).

the selection (position and zoom factor) on both devices. This specifically enables collaboration as people are not required to stay in place but can compare their selections side-by-side. For in-contact use on tabletops, Ebert et al. [Ebe+13] introduce lenses with varying map lens functions in the form of *TangibleRings* (see Figure 3.15a). These ring tangibles have two major advantages: they do not occlude or blur the underlying content and they allow for touch interaction within the region of interest on the tabletop. Additionally, rings with different radius can be concentrically positioned so that lens functions can be logically combined.

Moreover, there are several applications in the field of augmented reality (AR), where smart phones or tablets are used as physical interactive lenses that can be moved freely in space: Looser et al. [LBC04] present both volumetric and flat lenses for virtual objects used in augmented reality scenes. They enable magnification and alternative views of the lens and noteworthy use spatial interaction in the form of a hand-held trackball with attached AR marker (see Figure 3.15b) to enable positioning of the lens around the object. In a similar augmented reality context, Baričević et al. [Bar+12] extend the work of tablet lenses by adding user-perspective rendering to improve selection of virtual objects. In their work, they refer to any hand-held AR display as AR magic lenses since, true to the definition of magic lenses, the augmented content can only be seen within the bounds of the tablet display (see Figure 3.15c). As such, other AR works using hand-held displays to present additional views or content including information visualization (e.g., [Büs+17]) could equally be considered an AR magic lens. However, in terms of visualization lenses, the presented examples do not have a visualization context view but use the real world only for situating the presented focus. Since augmented reality and 3D visualization is not in the scope of this thesis, these related works will not be considered further.

A typical limitation of these tangibles and displays are their inflexible form. Every manipulation of the tangible's shape or size requires hardware modification. This can be easy, when using transparencies or paper or may involve more complex modifications for other materials such as acrylic glass up to being impossible for mobile devices. In any case, these tangible modifications are not easily reversible and therefore cannot be adjusted repeatedly and dynamically during interaction. As such, the individual tangibles might only be applicable in specific application contexts. However, their tangibility and the use of our natural motor skills can help make precise and intuitive manipulations of the lens.

3.4.4 Gaze-based Interaction and Head Tracking for Lens Positioning

On large high-resolution displays, controlling lenses with mouse or tangible interaction is infeasible. Gaze-based interaction techniques are a promising alternative [Jac90]. Similar to use on touch-enabled devices, magnification lenses are often used to make gaze interaction more precise, not for the interaction with lenses themselves. Examples for local magnification tools based on gaze dwell time are proposed by Lankford [Lan00] and Ashmore et al. [ADS05]. Yet, dwell times prevent quick execution of commands and hence hinder a fluent workflow, because users need to fixate a point for a certain amount of time. In further research, a second modality is used for explicit interactions, such as selections. Kumar et al. [KPW07] introduce the concept of look-press-look-release to solve this problem: Only when a keyboard key is pressed, the viewed region is enlarged. Stellmach et al. [Ste+11] use touch as the second modality for confirmation of selected elements and specifically discuss remote interaction with large displays (see Figure 3.16b). In these examples, lenses are used to improve gaze as an interaction modality and counteract imprecision. However, interaction with the lens as an interactive tool has also been discussed. Active positioning of these lenses can be cumbersome because of constant eye movement. Stellmach and Dachselt [SD13] developed different regions of movement for the fisheye lens, thereby interacting with the lens through gaze. When looking within the 'inner zone' of the lens, no movement is triggered, gaze focus in the 'active rim' steers the lens continuously and the 'outer zone' helps make fast, absolute positioning.

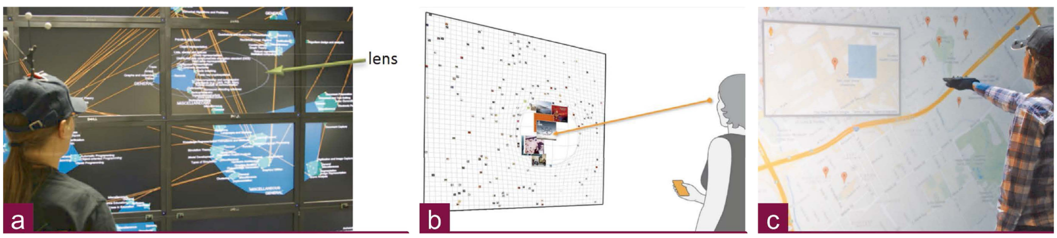


Fig. 3.16.: Head movement can be used to show details at a region of interest when moving around [Leh+11] (a). Additional interaction on a touch device may support the control of the lens [SD13] (b). Additional hand gestures can further invoke more explicit actions and manipulations of the lens [Bad+16].

In the context of graph exploration, Lehmann et al. [Leh+11] use head tracking to allow manipulation of a focus + context lens technique by physically moving in front of a large high-resolution display. The lens position is set to the users gaze position approximated by head tracking (see Figure 3.16a). In their study, users were given the choice between the lens technique and a distance-based adjustment of the level of abstraction which, while easier to control, resulted in information overload as details were presented both in focus and peripheral vision. Beside some negative effects due to the imprecision of the head tracking and natural head movements, users preferred the interactivity of the lens which helped them cope with the amount of information and maintain the context orientation. Head tracking is also used by Spindler et al. when designing Tangible Windows [SBD12] to provide a volumetric perception on the elements presented on the tangible views when working in 3D information spaces.

Recently, after the publication of our own BodyLenses exploration [Kis+15] (chapter 7), Badam et al. [Bad+16] explore the use of movable views as lenses at a large vertical display. Multiple users can manipulate their individual views implicitly by moving in front of the wall-sized display as well as explicitly using mid-air gestures or foot interaction (see Figure 3.16c). Additionally, pairs of users can step together to merge their individual views and switch merge modes using a collaborative gesture where both users lift an arm.

3.5 Summary and Resulting Challenges

This chapter defined and categorized the concept of magic lenses, elaborating on the principle components of these lenses for application in information visualization, their geometric properties that define the lens selection as the input for the lens function, and an overview of the diverse lens functions that have been applied to support varying data types and interaction tasks. Very diverse and flexible lens functions have been proposed to support a wide range of task and user goals, both for graph analysis and general information visualization. As a result, lenses have been shown to provide rich possibilities in way of manipulating the visualization for better understanding, improved exploration, and more extensive analysis of information spaces. However, in terms of **interaction**, these lenses have mostly remained very static tools with only one lens function for one specific purpose. It has been shown that existing work rarely addressed the flexible adjustment of lens functions and their parameters. Many of the presented research works aimed at supporting visualization functionality and goals, i.e., they proposed specific lens functions (mostly in mouse and keyboard environments). These works adjusted their lens functions in global dialogs and forms (if mentioned at all). These adjustment techniques however cannot easily and efficiently be applied to novel display settings and natural user interfaces, nor are they applicable in multi-user scenarios.

Lens research focusing on other novel interaction modalities has rarely used lens functions other than magnification (see "Lens Function" in Table 3.2). Hence, with the exception of a zoom factor which has been adjusted using a pinch gesture or inherently been manipulated when enlarging the lens, lens functions have rarely been parametrized in these cases. Notable exceptions are Tangible Views [Spi+10] where the distance of the cardboard to the tabletop influenced the level of abstraction for a graph representation and rotation has been used to manipulate the degree of distortion for a fisheye effect. Furthermore, lenses in research have rarely been integrated into a system with more lens functions or diverse interaction to solve multiple, varying tasks supporting a complete workflow of data analysis. That is, many of the works support multiple lenses of the same type, but rarely lens function combination has been applied especially combining the effect of different lens functions (see "Multiple Lenses" in Table 3.2). CGV, a system by Tominski et al. [TAS09], has made initial progress in this regard, as it uses three different lens functions for graph exploration and one composite lens uniting these into a new lens function. Still, the **flexible use and combination of lens functions** in one or multiple lenses to our knowledge has not been applied. To summarize, up to now lenses in natural user interfaces lack the ability to support users in not only positioning but also working with the lens – this includes adapting it to the current data selection (resize, reshape, etc.), selecting the appropriate lens function, as well as parameterizing the lens and its function to support the current analysis.

As has been shown with the range of possible lens functions, we believe lenses to be very well suited to support the exploration of data. While this includes the manipulation of the visualization, inherent in the definition of magic lenses is the temporarily limited, transient effect, so that manipulations will not remain. Few works have broken with this inherent characteristic by adding a permanent **manipulation** of the underlying data triggered by the lens. The EditLens [Gla+14] is a good example that the spatially-restricted property of the lens effect can be an important aspect for manipulation as well. Also, the Color Lens [EDF11] has been designed to give a preview of changes that can later be applied to the entire context. As a result, the transition from temporary effect, in terms of preview, and later confirmation to make this effect permanent (e.g., confirming the actual add-operation of a node [Gla+14] into the data source) may very well be a new concept for lens extension which should be investigated further.

It has been shown that collaboration scenarios include times of loose collaborative work, where users work independently and in-parallel, and times of closely coupled collaboration (cf. chapter 2). Characteristics of the lens, such as their interactivity, spatially-restriction (not always, see effect extent in section 3.3) and transient effect have the potential to make magic lenses an excellent tool for **multi-user applications** where users can work on the same context, but have their individual local exploration region that keeps them from interfering with each other. When multiple users work with multiple lenses in the same context visualization, the lenses may provide both manipulations of individual regions of interests but may also enable the merge from parallel work to closely coupled collaboration. It should hence be analyzed and observed what and how lenses can contribute to collaborative settings and display environments, investigating their advantages and potential.

To summarize, there is the need to address the following challenges, resulting from the discussed limitations:

C1 Configure lenses by supporting parameterizing (geometric) lens properties as well as the lens function and its parameters according to the current need.

C2 Turn lenses from the current one-function tool into a flexible tool with selectable functions within one lens, flexible combination of functions, and the possibility of multiple lenses on the same context visualization.

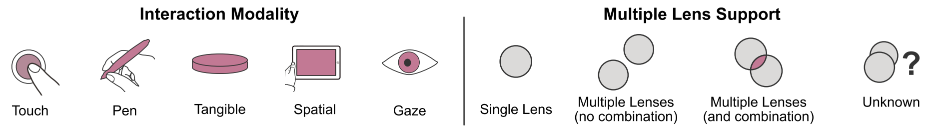
C3 Support both exploration *and* manipulation tasks through lenses also regarding the transition from temporary to permanent manipulation of the view. *(This challenge however is not within the scope of this thesis work.)*

C4 Improve the lens to become a tool that seamlessly integrates into the workflow by allowing fluent and effortless interactions and transitions while enhancing the user's control of the lens.

C5 Investigate the potential of lenses for multi-user scenarios and use of their spatial restriction in both loosely and closely coupled collaboration phases.

After identifying and discussing these challenges, the following chapter will present specific solutions for multi-functional lenses on touch-enabled interactive surfaces that go beyond the magnification of content. These solutions specifically take the first step in addressing challenges C2 and C4, creating a flexible tool with dynamically adapted lens function(s) that can be fluidly adapted by the user.

	Interact. Modality	Main Task	Lens Function	Multiple Lenses	Parameter to Adjust	How to Adjust Parameter
Frisbee [Kha+04]	Touch	E [ABSTR.& ELAB.]	magnification	Multiple Lenses (no combination)	zoom-factor	touch menu at frisbee
DTLens [FS05]	Touch	E [ABSTR.& ELAB.]	magnification	Multiple Lenses (no combination)	zoom-factor	pinch-to-zoom
iPodLoupe [Voi+09]	Touch	E [ABSTR.& ELAB.]	magnification	Multiple Lenses (no combination)	zoom-factor	pinch-to-zoom
PushLens [Sch+10b]	Touch	E [RECONFI-GURE]	remove edges	Unknown	no	–
FingerGlass [KAP11]	Touch	E [SELECT]	magnification	Single Lens	size/zoom-factor	finger distance
Transmogrifi-cation [Bro+13]	Touch	E [RECONFI-GURE]	magnification, deformation	Multiple Lenses (and combination)	(indirectly)	define source and destination
ZoomLens [SS14]	Touch	E [ABSTR.& ELAB.]	magnification	Unknown	zoom-factor	movable item at lens
PhysicsLenses [BHR14]	Touch	E [ABSTR.& ELAB.]	magnification	Multiple Lenses (no combination)	zoom-factor	finger distance
OrMIS [Bor+14]	Touch	E [ABSTR.& ELAB.]	magnification	Multiple Lenses (no combination)	zoom-factor	pinch-to-zoom
Smarties [CBF14]	Touch	E [ABSTR.& ELAB.]	magnification	Multiple Lenses (no combination)	zoom-factor	remotely from mobile device
Kinetica [RK14]	Touch	E [FILTER]	highlight items	Multiple Lenses (and combination)	unknown	–
Pointing L. [Ram+07]	Pen	E [SELECT]	magnification	Single Lens	zoom-factor	stylus crossing border
AR-Lenses [LBC04]	Spatial	E [ABSTR.& ELAB.]	layer select	Single Lens	layer	handheld button
PaperLens [SSD09]	Spatial	E [ABSTR.& ELAB.]	magnification, layer select	Multiple Lenses (no combination)	zoom-factor or layer	distance or orientation
Tangible Views [Spi+10]	Spatial	E [ABSTR.& ELAB.]	semantic zoom, expand nodes	Multiple Lenses (no combination)	zoom-factor, distortion of data	distance or orientation
Aug.Vis. [Sör+11]	Spatial	E [ABSTR.& ELAB.]	alternative layer	Multiple Lenses (no combination)	transfer function	input on mobile device
Embodied L. [KE12]	Tangible	E [ABSTR.& ELAB.]	magnification, layer select	Multiple Lenses (and combination)	no	–
TangibleRings [Ebe+13]	Tangible	E [ABSTR.& ELAB.]	layer select	Multiple Lenses (and combination)	layer	touch menu along tangible
[Lan00]	Gaze	E [SELECT]	magnification	Single Lens	no	–
FisheyeLens [ADS05]	Gaze	E [SELECT]	magnification	Single Lens	no	–
Phys.Nav. [Leh+11]	Gaze	E [ABSTR.& ELAB.]	semantic zoom, expand nodes	Unknown	size	distance to display
StillLooking [SD13]	Gaze	E [SELECT]	magnification	Single Lens	size + zoom-factor	remotely from mobile device



Tab. 3.2.: Overview of related work considering lenses in natural user interfaces focusing on applied lens function and the support for multiple lenses as well as whether and how parameter adjustment can be triggered by the user.

Multi-Functional Multi-Touch Lenses

4

Contents

From the analysis of lens interactions in the previous chapter, it became clear that while in traditional desktop settings a wide range of complex lens functions have been proposed and applied, lenses used on touch-enabled surfaces have mainly been used for very basic purposes, often focused on magnification. As a result, existing touch solutions have been limited to moving and resizing the lens using touch, but have not yet realized the potential of the lens as a flexible, multi-functional tool. Novel multi-touch interaction concepts need to address the limitations and challenges defined in the previous chapter (see section 3.5), especially the need to integrate the lens as a flexible, adjustable tool (C2) into the data analysis workflow supporting flexible, effortless interaction and configuration of the lens as a tool (C4).

Multi-touch interaction has become a predominant and widely spread modality in a wide range of display setups, sizes, and contexts (e.g., smartphones, tablets, tabletops, and wall-sized displays). As such, more and more data analysis tasks, mostly in professional settings but also for personal data analysis, have the need to also be supported in these device environments. This chapter focuses on extending the existing basic interactions for magic lenses on touch-enabled surfaces creating a configurable lens tool that supports manipulation of the currently active lens function, but also the flexible combination of varying lens functions and their parametrization. These concepts aim to apply the principles of multi-touch interaction to magic lenses not only to support existing functional advantages that have been shown in traditional mouse and keyboard settings, but further to improve on this tool by supporting a fluent workflow when analyzing data using lenses. To investigate the possibilities of the configuration of lens function parameters and the combination of functions, alternative touch interaction techniques were designed and combined to support both widget-based interactions as well as gesture-based manipulations of the multi-touch lens (section 4.2). These concepts have been applied for graph visualization. They were implemented in a prototype for graph-specific lens functions presenting details on demand as well as focusing on examining the relations among nodes (section 4.3). A user evaluation compares the widget-based approach with a traditional menu technique, used with both mouse and touch to counteract and investigate the bias of the interaction modality (section 4.4). All in all, this is a first realization of multi-functional multi-touch lenses that are flexible in task-specific lens effects and support user-specific interaction styles.

Parts of the research presented in this chapter have previously been published in:

Ulrike Kister, Patrick Reipschläger, and Raimund Dachselt. 2016. MultiLens: Fluent Interaction with Multi-Functional Multi-Touch Lenses for Information Visualization. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces* (ISS '16'). ACM, New York, NY, USA, pages 139-148.

4.1 Motivation

In existing research, multi-touch lenses have been used mostly to apply magnification to a focus region [KAP11; BHR14; SS14] to show more details or improve the imprecise selection of touch by addressing part of the fat-finger problem (e.g., [BWB06; Voi+09]). Even when applied to solve a data-specific problem using a specialized lens function, e.g., pushing away unrelated graph edges [Sch+10b], only very basic interactions with the lens have been designed. Almost all touch-enabled lens solutions support moving the lens through drag (on either the border or inner area of the lens) and pinching either to resize the lens or increase the zoom factor of the magnification. The research presented in this chapter investigates the advantages of multi-touch interaction for magic lenses in settings addressing more than just magnification and movement of the lens, but the application of varying lens functions and their parameterization.

In data visualization, most research on magic lenses presented the development of a specific lens function addressing a very specific task. To transfer this knowledge into real-world use, multiple different lens function need to be applied as users concern themselves with more than one goal or task. The flexible selection of these lens functions and configuration to fit the user's current goal needs to be possible in a fluent, continuous workflow. To achieve this, it is the goal of this work to examine multi-touch as a possibility to support fluent interaction with the data through lenses and thereby create a touch-enabled magic lens tool that extends the original idea of a lens by giving the user a flexible toolbox of functions for data analysis.

Therefore, we[1] propose MultiLens, touch-enabled magic lenses for fluently manipulating functions, parameters, and combinations of lenses on interactive surfaces. With MultiLens, we designed a flexible touch-enabled lens interaction technique looking at varying ways of interacting with lenses using multi-touch. Therein, we extend the discussion of advantages and disadvantages of WIMP-based touch user interfaces and gesture-based interfaces as investigated by Drucker et al. [Dru+13] (cf. 2.2.1) and the use of flexible, fluent tools adapted to novice and expert users, building on works such as NEAT [FLD11]. We contribute a novel multi-touch menu technique for magic lenses using a widget-based approach with a drag-snap slider for relative parameter adjustment. We also propose a continuous gesture set for rapidly changing lenses and their primary parameters in one seamless phrase. Finally, we also address the flexible combination of lens functions both in the widget-based and gesture-based approach, thereby developing the lens as a flexible multi-purpose tool for data analysis.

[1] 'We' in this chapter relates to the author Ulrike Kister, as well as Patrick Reipschläger and Raimund Dachselt as co-contributors on this research.

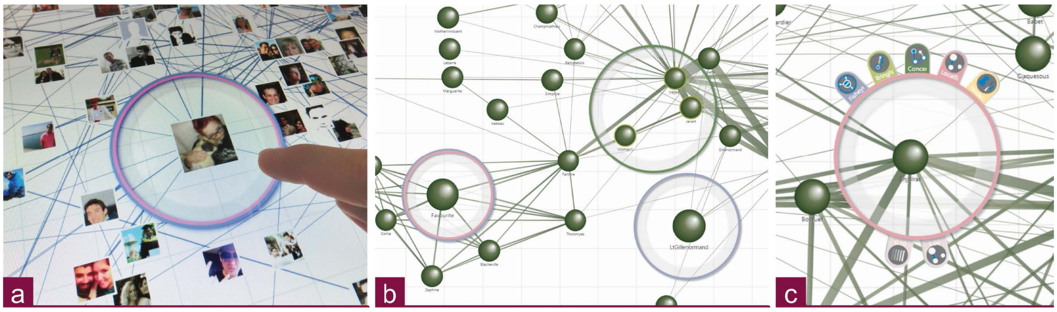


Fig. 4.1.: MultiLens describes touch-enabled lenses (a) that support flexible activation of multiple lenses with multiple lens functions (b) and can be parametrized using either gestures or a radial menu around the lens (c).

4.2 MultiLens: Fluent Interaction with Multi-Touch Lenses

MultiLens is a concept of magic lenses that can be manipulated using touch and incorporate multiple lens functions as well as interactions to manipulate the lens function and their parameters. We apply these touch-enabled lenses to graph exploration using a selected set of graph lens functions (cf. 3.3.2) that are chosen both because they support a range of diverse graph tasks as well as their varying properties in terms of effect class and effect extent: *fisheye*, *Local Edge*, *Bring Neighbor* and additionally a lens-version of Bring & Go [Mos+09] called *K-Neighborhood* lens for exploring connectivity of nodes to a higher extent than immediate adjacency.

The lens is visually represented by its shape and border. As it is the most commonly used form, our proposed lenses are of circular shape. The border is the visual representation of the lens and it is the component from which manipulations are supported. We emphasize the fact that the lens functions as a ring and the inner part of the lens is touch-through, i.e., data elements underneath the lens can be touched and interacted with. That means that all interactions that function on the context elements equally apply in the inner area of the lens. This is important as the data within the lens is the user's region of interest, i.e., the main focus, and should hence be available for any interaction to gather more information on the data. Building on the same principle, the border itself is rather thin to reduce any unnecessary occlusion of the data underneath. However, to support comfortable interactions with the lens, we decided to widen the touchable area around it with a semi-transparent area. The border further encodes the currently active lens functions in its color. As multiple functions can be active in one lens, the border can consist of multiple colored rings (see Figure 4.1).

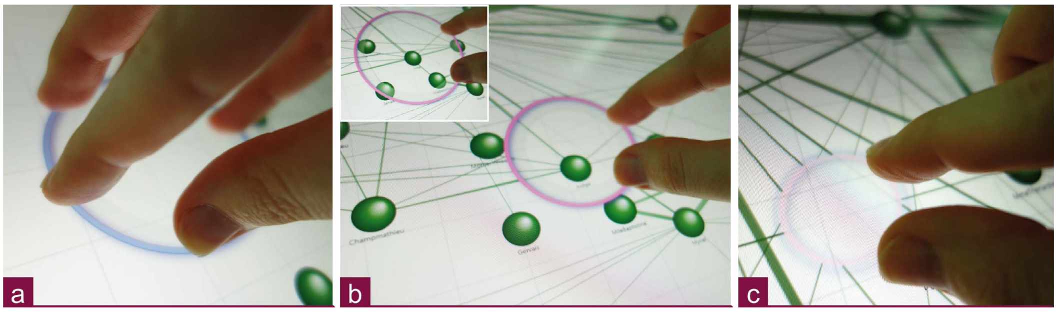


Fig. 4.2.: The lens can be created using five fingers forming a circle (a) and resized using the pinch gesture on the lens border (b). When resizing to a minimum, the lens becomes transparent for deletion on release (c).

The following description of the MULTILENS interaction concepts is structured as follows: First, the basic manipulations of the lens are described in a very brief overview. Afterwards, the radial touch menu will be discussed elaborating on its novel relative slider technique for selection and fluent parametrization of the lens function. This is followed by details on an alternative gesture-based concept for expert users to manipulate functions while immediately setting a primary parameter. Finally, section 4.2.4 summarizes and extends the aspects of these interaction techniques that support combination of lens functions or lenses.

4.2.1 Basic Manipulation of the Lens

Creation We designed interactions for creating lenses as well as manipulating their position and size. Creation can be accomplished by holding five fingers in close proximity onto the interactive surface, similar to *Smarties* [CBF14]. This gesture was chosen as the fingers form a circle which resembles the circular lens. The fingers also provide a diameter as an initial lens size value. As an alternative to this quick expert gesture, a menu can be invoked when performing a hold on the background. It provides possible lens functions in the form of a radial menu to select as an initial value. Tapping one of these functions creates a lens with a default size. Furthermore, we considered and implemented support for drawing a shape by touch or pen for user-defined lens shapes. There are however various disadvantages to this concept: It requires the user to select an appropriate lens shape that is suitable for moving the lens onto the varying layouts of a graph's representation. Further, the use of pen is only convenient for application cases where the pen is already in place, e.g., applications that support manipulation operations with the pen or require interaction at high precision.

Translation and Scaling After creation, the user needs to manipulate the properties that define the lens selection, i.e., position, size, shape, and orientation (c.f. section 3.2). As the border embodies the lens as a whole, dragging on the lens border is used for translating the lens. Scaling can be accomplished using the well-known two-finger pinch gesture on the lens border (see Figure 4.2). As we simplified the lens to be of circular shape, the properties shape and orientation are irrelevant for this realization of touch-enabled lenses. However, this could easily be extended: A change of orientation could be mapped by allowing two-finger rotation on the lens border similar to scaling. Further, an additional menu could be integrated to allow shape changes.

Deletion Finally, lenses can be removed by scaling them down to a certain minimal threshold which will create a fade-out effect and removes the lens on release if it is not resized beforehand. Alternatively, lenses can be dragged off-screen to be removed. However, this of course is only feasible on displays of smaller size where the border of the screen is in reach.

4.2.2 Radial Touch Menu & Drag-Snap Slider Technique

To support novices or the infrequent use of MULTILENS, we make the functionality explicit by using graphical widgets around the lens border (see Figure 4.3a), similar to *GeoLens* [ZDS12]. While *GeoLens* uses a fixed timescale around a circular element, we present a flexible tool for parameter adjustment. A radial context menu along the border of the lens exposes all possible functions and parameters to the user, enabling in-place interaction close to the data. Visibility of this menu can be toggled with a tap on the lens border. To support differentiation of lens functions and parameters, we separated the radial menu in two major groups. The handles at the top toggle the different lens functions, e.g., Fisheye or Local Edge, and are activated by a tap. Multiple active functions are supported and can be toggled sequentially. When a lens function is activated, its parameter handles are placed at the bottom of the lens. Parameter handles are only visible when the respective function is active. Each function has a unique color, which is used consistently for the lens

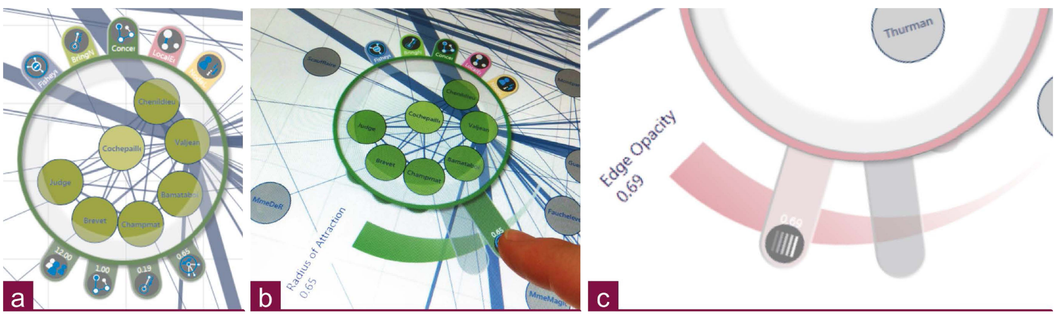


Fig. 4.3.: The lens provides an upper menu for selection of lens functions and handles at the bottom for parameter adjustment which are encoded with the associated lens function color (a). A slider around the lens can be invoked for parameter adjustment (b-c).

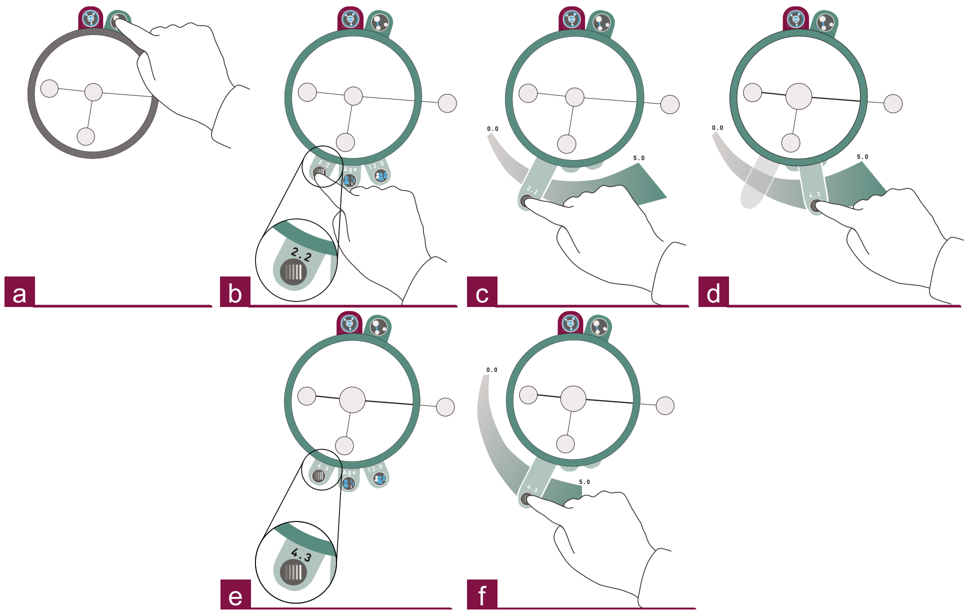


Fig. 4.4.: When activating a lens function (a), its parameter handles at the bottom become visible. A handle is dragged out (b-c) and can then be rotated around the lens to adjust its value (c-d). On handle release, it snaps back to its original position (e). When pulled out again, the slider position is automatically adjusted (f) again supporting relative manipulation.

border and all corresponding handles. For manipulating parameters, we propose two different interaction styles, one with sequential actions and the other with a seamless interaction phrase.

Discrete Tapping A *discrete tap* on a parameter menu handle at the bottom of the lens extends the handle and a slider appears around the lens. Simultaneously, other handles are miniaturized to make room for the slider (see Figure 4.3b). Movement of the handle along the slider adjusts the parameter of the function. Parameters are either continuous with free handle movement or discrete where the handle snaps to defined steps. Another tap on the handle hides the slider again. A shadow at the previous handle position on the slider track can be tapped to revert the parameter to the previous value. We ensure that parameter handles are always positioned in the same order on the lens border, using the user's spatial memory to facilitate fast navigation.

Drag-Snap Slider Technique In our experience, function parameters are often not set to specific values but instead relative to the current value, e.g., making a magnification bigger or smaller instead of setting a specific zoom factor. For more efficient and fluent interaction, we therefore propose a novel continuous technique for relative parameter adjustment on sliders called *drag-snap technique* as an alternative to the traditional discrete tapping. Alternatively to a tap, parameter handles can be pulled out and smoothly dragged along the border of the lens to change the parameter's value in one continuous movement (see Figure 4.4b-d). As soon as the finger is lifted, the handle snaps back to its original position on the lens border, selecting the chosen value. The slider automatically disappears while the selected value is still visible in the handle (see Figure 4.4e). As a result, the individual steps of (i) extending the slider to be visible, (ii) adjusting the value, and (iii) confirming the value by collapsing the slider are phrased into one continuous gesture. On dragging out the same handle again, the slider will have automatically adjusted its position (see Figure 4.4f, cf. 4.4c). Therefore, relative adjustment is again possible. To summarize, if users want to change a parameter, they may drag the handle outwards, move it to relatively adjust the value, and release the touch.

In allowing both discrete, sequential tapping as well as smooth, continuous movement, we support inexperienced users as well as the transition to more fluent interaction for advanced users within the same widget-based approach. Additionally, all parameter handles are positioned at the bottom of the lens to allow observation of the current change to the visualization in the center of the lens while rotating the handle along the slider.

4.2.3 Continuous Gestures for Lens Adjustments

To complement manipulating parameters using widgets, we propose a set of multi-touch gestures for the efficient and fast interaction with lenses. Together with the radial touch menu, MultiLens offers a joint interaction concept with different approaches to cater for different usage styles. Both allow invoking different functions within a single lens. As tools are often immediately configured to suit the individual needs of a user, we developed continuous gestures allowing for a fluent transition from first activating the lens function to subsequently adjusting its primary parameter without lifting the finger. We thereby support an unbroken kinesthetic

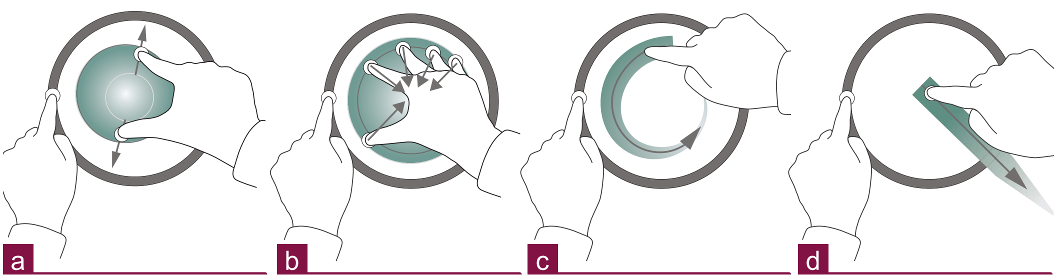


Fig. 4.5.: Each graph lens function is associated with a continuous gesture which activates the lens function while continuation of the gesture support immediate adjustment of a primary lens function parameter.

Fig. 4.6.: Using the continuous gesture, the user can switch from Local Edge (red) to fisheye (blue) function and continuously set the magnification factor as its primary parameter.

tension as recommended by Buxton [Bux95] and recently presented by Gupta and McGuffin [GM16] for improved command selection and parameter manipulation in radial menus. To distinguish lens gestures from interaction with the underlying visualization, the user touches the lens' border with the non-dominant hand to set the frame of reference [BH99] for the interaction of the dominant hand. Besides supporting a user's mental model of decoupling gestures affecting the visualization from gestures affecting the lens, this also avoids complex gestures and provides easily remembered shortcuts for expert users. We present the following iconic gestures for a range of selected example graph lenses.

The purpose of the *Fisheye* function is to magnify the content underlying the lens with continuous distortion toward its border. It is triggered using a pinch gesture (Figure 4.5a). We employ this gesture as the resulting magnification is quite similar to zooming which is predominantly done using pinch-to-zoom gestures. The primary parameter of this lens is the degree of magnification, which can seamlessly be adjusted with the activation of the lens function using the the amount of pinching, equal to the common usage of zoom.

For the classic *Bring Neighbors* function, affecting only direct neighbors, we deliberately designed the gesture mimicking the physical action: The user has to place his or her fingers onto the surface in form of a grabbing gesture, moving four or five fingers towards the lens center (Figure 4.5b). This also immediately manipulates the attraction radius using the diameter formed by the touch points. This way bringing the fingers closer together or pushing them apart will bring neighbors closer to the focus node or further away, respectively.

The purpose of the novel *K-Neighborhood* function is to clearly layout the different levels of adjacency by forming rings. It is invoked using a spiral drawing gesture (Figure 4.5c) depicting these concentric rings. Outward or inward spiral movement controls the degree of neighborhood, and the distance to the lens' center changes the radius of the circles. This gesture has also been examined in *CycloZoom+* [MLG10] where it was used for zooming purposes.

Finally, triggering the *Local Edge* function is performed with a drag gesture resulting in a straight line from within the lens area towards the border of the lens, symbolizing dragging an edge away from the lens area (Figure 4.5d). Transparency of the edges is directly controlled by the distance of the touch to the lens' center, changing the options between slight transparency to completely disappearing edges.

To summarize, the proposed gestures provide a seamless transition from activation to adjustment of the lens function, seamlessly phrasing together individual interaction steps [Bux95] (see Figure 4.6). Experts can invoke functions efficiently and fluently adjust the main parameters within their workflow. Similar to Frisch et al. [FLD11], with MultiLens we advocate the creation of interaction techniques for both novices and experts, emphasizing the need for effective multi-touch techniques for fluent use. To support all users, a switch between gesture and menu-based interaction is possible at any time without explicit mode switches.

4.2.4 Interactive Combination of Lenses

Similar to the presented parameter adjustment of lenses, MultiLens supports alternative ways of combining multiple lens functions. Moving two lenses on top of each other applies their individual lens functions onto the shared selection. This is a very clear interaction for novice users. If both lenses are similar in size (below a certain threshold) and would cover the same area, a colored ring inside the lens indicates that the lenses may be combined. Releasing the touch when feedback is visible confirms the combination. The feedback vanishes when the lens is held for a second, keeping the lenses from snapping together. Lenses of different size can be moved on top of each other without automatic combination. A combination of *Fisheye* and *Local Edge* function can be seen in Figure 4.7. This combination presents a new tool in itself that supports focusing on central nodes. It can be moved as one lens. Our concept of separating the lenses again is accomplished by holding a finger on the border of the lens. This will split the lens border into individually colored

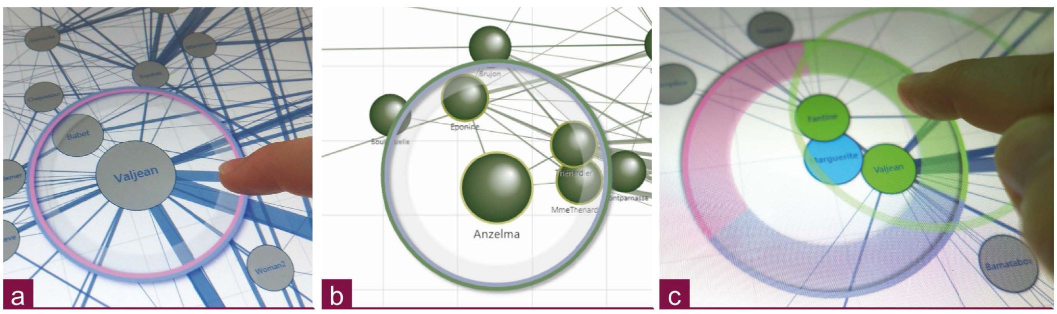


Fig. 4.7.: The flexible combination of lens functions enables improved exploration tools, e.g., Fisheye and Local Edge (a) or Bring Neighbors and Fisheye (b). Lens functions can be added and removed by menu or gestures. Alternatively, the lens can be split into segments and lens functions can be dragged out creating a new lens (c).

segments representing the currently active functions which can then be dragged out from the lens to be removed (see Figure 4.7c, not in current version of prototype).

While this interaction for combining lenses requires two lenses already in place, simply adding a function to an existing lens can also be useful. The first simple way of adding multiple lens functions to the same lens is toggling individual lens functions sequentially in the radial touch menu. Furthermore, we also distinguish two modes when using the expert gestures: replace or add. These modes are differentiated by the number of fingers placed on the lens border with the non-dominant hand, again setting the frame of reference for the interaction. We use two fingers to activate the alternative action, similar to [FHD10]: Performing the gesture with two fingers on the lens border will add the new function to the lens instead of replacing the existing one(s).

4.3 System Architecture & Technical Realization

The MultiLens system was designed for use on a 27" Perceptive Pixel [@Per06] display with a resolution of 2560×1440 px. It incorporates the selected graph lenses for sample graphs in graphML format, e.g., showing the human disease network [Goh+07] of co-occurring diseases and genes, or social networks in JSON format generated by the Facebook Graph API Explorer [@Fac15]. The prototype was improved, re-designed, and extended in various iterations due to various feedback sessions and tests. It was also used for a range of student projects. Furthermore, it was designed and iterated to be structured in a modular way to account for flexible addition and exchange of components, e.g., lens functions and interactions.

The prototype was implemented in C# using WPF and following the Model-View-ViewModel pattern (MVVM). The graph structure is organized using Quickgraph's graph interface [@Qui07] while basic graph algorithms such as layouting were applied through GraphSharp [@Gra09]. The system processes both Windows Touch events and TUIO event, where TUIO events are mapped internally to Windows Touch events. Our *MultipleFilterLens* component contains separate entities for the management of lens shape and lens functions (see Figure 4.8). Shapes, e.g., circle, rectangle, ellipse, and polygon, are responsible for checking if a point or element is inside their boundaries as well as calculations of the resulting intersections with a line, i.e., graph edges. Lens functions or filters define the update function to manipulate nodes or edges in the shape. The implementation for combining lens functions within one lens is adapted from the *model in – model out* strategie (cf. section 3.3.3). It is

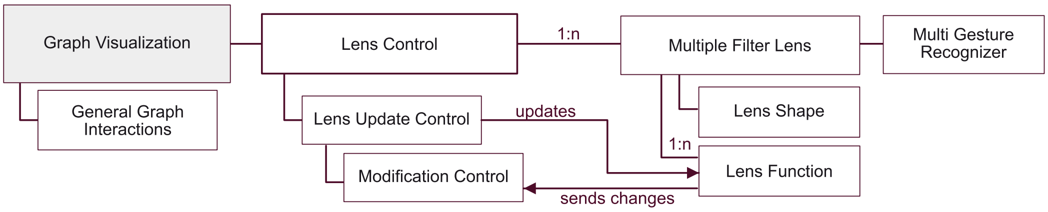


Fig. 4.8.: The MULTILENS graph application contains a lens control which manages multiple lenses. Each lens function in the lens is updated by its *LensUpdateControl* and accesses the graph through the *ModificationController*. Each lens' *MultiGestureRecognizer* notifies it of any identified gesture to manipulate parameters.

managed by the *LensUpdateControl* and its *ModificationController*[2] which manages a list of modifications, i.e., changes to the elements or their properties. The varying lens functions have a property that describes their precedence, ordering them for application to the graph. They are applied one after another where functions always process data from the modification controller which handles the manipulated values from previous lens functions. Depending on data type and appropriateness, different strategies for merging values can be used, e.g., averaging numerical values, appending strings, or using the last value applied.

The radial touch menu with its parameter bars is implemented for circular and rectangular shapes to switch lens functions. However, the circular slider track for parameter adjustment only applies to the circle lens shape and hence is not shown for other shapes. All previously described interactions, both the discrete tapping and the drag-snap slider technique, are implemented so that adjustment of parameters is possible. When the user touches the lens border, an additional, transparent layer above the lens is invoked which receives touches for the gesture recognition. The implemented recognizer controls multiple asynchronous background worker threads to detect the continuous gestures that activate lens functions and set the function parameter accordingly.

4.4 Evaluation of the MULTILENS Concepts

We decided to assess the MULTILENS concept aiming at gaining insights for further iterations as well as evaluating its usability and suitability for practical use and hence designed and conducted a comparative user study. In preparation of this comparative evaluation, we first conducted a pilot study gathering initial user feedback to iterate our concepts and identify shortcomings in our implementation. As an initial evaluation, this study was of qualitative nature and was designed to

[2]These components were designed and implemented as part of the Master's thesis of Patrick Reipschläger which was supervised during this PhD research work.

focus on the users' understanding of the concepts, their personal preferences, and comments on the necessary scope and completeness of the implementation, e.g., the gesture recognition. The comparative evaluation then focused on the widget-based interaction technique as an essential part of the MULTILENS concepts and the interaction basis for all users. It compares this novel interactive lens approach to a state-of-the art global menu which has equally been augmented with the flexible selection, parametrization, and combination of lens functions.

4.4.1 Initial User Feedback

To gather initial user feedback on the feasibility of our MULTILENS concepts and to identify shortcomings and limitations of the implementation, we invited six participants (all right handed, average age: 27, gender-balanced) to take part in an initial pilot study. All participants were either students or colleagues in the computer science department, but had only basic knowledge of graph analysis. However, participants had deep knowledge of HCI and interaction design. All participants considered themselves very familiar with small touch devices such as smartphones or tablets.

The study was performed using a simplified social network graph. Participants were presented with three different lens functions. The basic functionality was demonstrated starting with the radial menu and concluding in the gesture set. Participants then had a trial phase where they could individually explore the different lens functions and their parameters. They were encouraged to think-aloud, ask questions, and report on anything that was unclear to them. We took notes and video recorded the sessions for analysis.
As study tasks, we asked participants questions related to the connectivity of different people in the graph, e.g., 'Who are the direct friends of Alba?', 'Which people connect Alba to Bob?', or 'Find the three people that are not connected to anyone else.' Time for these tasks was not limited. For all tasks, participants could choose which lens function they considered most helpful and wanted to use. Some questions were designed to be easier to solve using multiple lens functions in combination. Finally, participants were asked to fill a usability questionnaire (see Appendix A.2.1) with fourteen lens related questions using a five-point scale (1=disagree, 5=agree). While no statistically significant assertion can be made from such a small number of participants, the initial feedback was used to get a first understanding of the preferences, uses, and possibly limiting factors of the interaction with the MULTILENS system.

From our observations and results, we found that participants understood the concepts of MULTILENS and we observed smooth exploration of the graph even

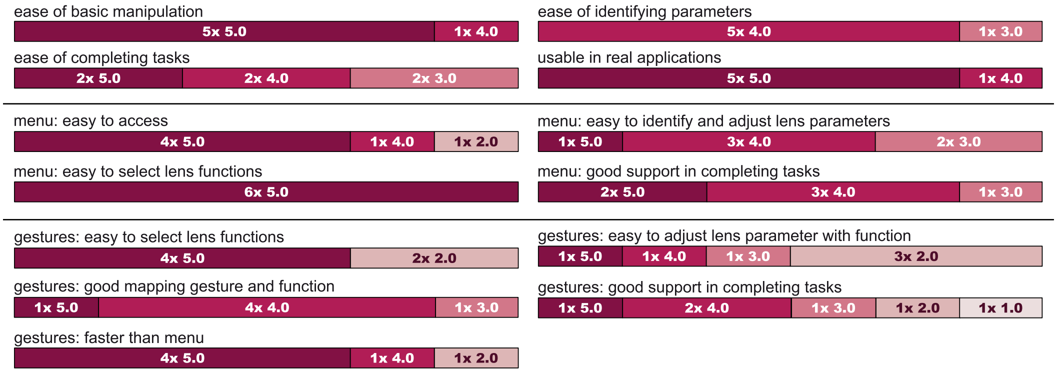


Fig. 4.9.: Participants' answers in the initial feedback session to questions regarding ease of use and assessment for basic interaction, radial menu, and gestures on a scale of 1 to 5.

within the try-out phase. All participants agreed that basic manipulation of the lens was easy to understand and perform (M=4.8, SD=0.37) and that the lenses were helpful in exploring the graph and completing the tasks (M=4.0, SD=0.82).
Five of the six participants started using the radial menu first, exploring each lens function separately and looking at the individual parameters of each lens. Participants clearly agreed that selection of lens functions from the menu was easy (M=5.0, SD=0.0). However, during the tasks, we observed that participants sometimes switched between parameters of the chosen function, testing them out again and again and did not remember their purpose and effect. This is also reflected in the answers to the question if identification and adjustment of function parameters in the menu was easy (M=3.8, SD=0.69). A possible reason is the lack of experience and knowledge of individual lens parameters. A long-term study would be necessary to help evaluate the learnability of these individual lens function parameters. Nevertheless, people understood the different functions well and often commented positively on the lens effects: "Dragging [the K-Neighborhood function] is cool, cause of the reaction on the graph. That's fun!". A couple of participants also used our novel K-Neighborhood differently than originally designed, pulling in multiple levels of neighborhood to see not the neighbors, but loners which were not pulled in.

The lens menu was perceived to be suited well when having to answer the task questions (M=4.2, SD=0.69). This is reasonable as all participants were novices in terms of graph exploration with lenses and this part of the tool was explicitly designed for casual use. There were large inconsistencies in how well participants considered the gestures helpful for performing the tasks (M=3.2, SD=1.34) and for selecting lens functions (M=4.0, SD=1.41) as can be seen in Figure 4.9. Yet, most participants found the mapping of gesture and lens function reasonable and well-suited (M=4.0, SD=0.58). Still, the varying results visible in this initial feedback session make clear that the gestures have to be refined and solutions for teaching the lens functions and its gestures need to be considered for evaluating them further. As inherent in their design, our continuous gestures with their shortcut characteristics are suited for long-term and expert use. While some participants welcomed the gestures, we see higher potential in making lenses easily available to interaction novices, hence focusing on the radial menu in the following.

As a result of this study, we iterated our concepts and improved our implementation of MultiLens. We improved on the gesture recognition allowing more variance in performing the gestures. While there are positive study results for the mapping and the easy selection of lens functions using the gestures, the setting of lens function parameters had strongly inconsistent results. Further extension and investigation for long-term use is necessary to evaluate and iterate these expert gestures. However, we saw clear potential in the use of the radial menu and relative slider technique and believe this to be an easy way to enable magic lenses as a flexible tool for data analysis. Hence, we decided to focus on the widget-based approach and investigate it further to asses its usability and efficiency in comparison to traditional menus.

4.4.2 Comparative Evaluation with a Traditional Menu Using Touch and Mouse

Since the flexible radial menu including the *drag-snap slider technique* is a central aspect of our approach, we investigated its comprehensibility and efficiency. We decided to compare the radial menu concept to state-of-the-art implementations of magic lenses. Most state-of-the-art lens implementations use mouse interaction and global menus or dialog boxes for the parameter adjustments [Tom+17]. To conform to that without limiting interaction, we use a traditional side menu for comparison while supporting the same functionality of lens function parametrization and combination from the MultiLens concept. To distinguish between differences in performance resulting from the interaction modality, we decided to evaluate both touch and mouse interaction for the traditional menu. This results in three conditions for our comparison: mouse + traditional menu (M+TM), touch + traditional menu (T+TM), and our own approach touch + radial menu (T+RM).

We hypothesized that the radial touch menu (T+RM) as part of our MultiLens is more efficient in terms of required time per action than the current state-of-the-art menu versions (both M+TM and T+TM) because of shorter distances between lens and menu. Furthermore, we believed that users will prefer the directness of the novel MultiLens implementation over the state-of-the-art as interaction is less decoupled from the focus (i.e., the selected data).

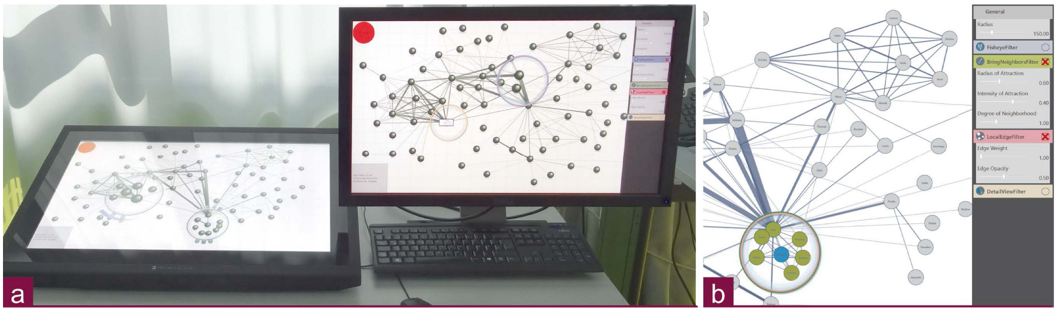


Fig. 4.10.: The study was conducted using a slightly tilted display for both touch conditions and a vertical PC display for the mouse condition (a). The traditional menu was used both with mouse and touch for comparison with our radial touch menu (b).

Participants We recruited 22 unpaid participants (all right handed, 5 female) to take part in this comparative evaluation study, none of which had taken part in the pilot study. The participants were students of the math, computer science, and engineering departments of our university. Their age ranged from 21 to 36 ($M = 24.7, SD = 3.3$). All but two reported daily use of mobile touch-enabled devices, of which eight had occasional to frequent use of larger touch-surfaces (interactive white-boards or wall-sized displays) and eight had never worked with touch on devices larger than tablets. 13 had no or very little experience with graphs, while nine stated they work with graphs occasionally, e.g., for UML diagrams or research projects. Magic lenses were completely unknown or never used by 14, three had little experience and only five had worked with magic lenses occasionally up to frequently for research or student projects.

Apparatus Beside the MULTILENS C#/WPF prototype described in the previous section (4.3), we created a state-of-the-art implementation: While lenses can be dragged and selected on their border, a side menu with WPF menu elements for use with either touch or mouse interaction supported the manipulation of their functions and parameters (see Figure 4.10b). The touch conditions (both T+TM and T+RM) ran on the 27" Perceptive Pixel [@Per06] display which was slightly tilted for comfort. For M+TM, we used a standard vertical PC display setup with the same resolution and size to represent a state-of-the-art setup for comparison (see Figure 4.10a). Both displays were connected to the same PC for equal power in performance.

Methodology and Procedure The time used for the experiment per participant was approx. 50 min. Prior to data collection, participants completed a questionnaire on their demographic details and experience. All participants were introduced to the effect of each lens function that was used in the study by being presented with images on paper (see Appendix A.2.2). Afterwards, they were instructed to make themselves comfortable in front of either the PC display (for mouse condition) or the

Perceptive Pixel display (for touch conditions). The experiment was a within-subject design and the order of conditions was counterbalanced for 21 participants. An additional 22nd participant was recruited to replace an outlier (see observations below).

For each condition the participant started with a training phase where a researcher explained the basic functionality of each menu, starting with the basic repositioning of the lens to individual lens function parameter adjustments. The participants were given time to make themselves familiar with the interactions until they felt comfortable using the condition ($M = 3.1\ min, SD = 0.9\ min$). They could then trigger the beginning of the experiment. In each task, participants were presented with the task description and they had time to carefully read the instructions before pressing the start button and moving to the graph. We created three social network graphs with random names as data sets for the study tasks. To balance complexity, the number of elements (77 nodes, 254 edges) and node degrees were the same. Using a force-directed layout, each graph was arranged differently. They were counterbalanced so that each graph was used equally often among conditions.

We designed the experiment to include two groups of tasks: six *process tasks (A)* and three *exploration tasks (B)*. In the first group *(A)*, participants were given a list of specific instructions stating up to four individual manipulation steps for each task. Examples include (1) 'Invoke lens function Local Edge', (2) 'Set function parameter Edge Opacity to 0.2', etc. We designed the process tasks to conform with normal lens exploration processes. Hence, the tasks often started or ended with a move of the lens onto a specific, highlighted node. Complete example tasks can be found in Appendix A.2.2. For these *process tasks*, we measured task completion times and logged times of individual interaction steps. *Exploration tasks (B)* were less specific in which interaction steps to take. They included questions such as 'Find and select the vegetarian among the friends of Frida' or 'Select all neighbors of Lorenzo'. We designed each graph for these tasks to include the same problems (e.g., edge clutter, so that the node in question had unconnected edges running through and using the *Local Edge* function was necessary). We also logged completion times for these tasks, but their exploratory character makes it difficult to tell how much of the time was affected by the menu technique. These tasks primarily served for observing the exploration process.

At the beginning of each task, its description was shown full screen so that users had time to understand and read the complete task description. After confirming their understanding, they were presented with the graph and depending on the task one to few already placed lenses without active lens functions. A short summary of the task description, either a list of steps for *process task* or the question itself for *exploration tasks*, was given in the lower left corner of the screen. Finally, a red button at the upper left allowed confirming the completion of the current task and triggered the next one (see Figure 4.10a). We video recorded the interactions of each user for later analysis and took notes of our observations during the task completion. After every condition, the participants filled out a separate questionnaire of NASA TLX questions (Raw TLX) answering on a 7-point scale to evaluate the physical and mental demand and stated the perceived ease while manipulating the lens. After completing all three conditions, the users were asked to complete a final comparing questionnaire, checking which of the conditions was perceived as most and least helpful, fastest, most comfortable in use, and applicable in everyday work. The (translated) questionnaire can be found in Appendix A.2.2.

4.4.3 Observations & Results of Evaluation

The analysis of the completion times for the process tasks showed outliers for two participants who we also noticed had problems reading the descriptions or partially disregarded the instructions. We removed those two participants for all conditions. As a Shapiro-Wilk test indicated that the assumption of normal distribution was violated, we used Friedman's ANOVA showing a significant difference between T+RM and the other conditions ($\chi^2(2) = 17.1, p \leq .001$). There was no significant difference between M+TM and T+TM. Even though we hypothesized a better performance, T+RM ($M = 26.59\ s$) was slightly slower than M+TM ($M = 22.95\ s$) and T+TM ($M = 22.45\ s$) (see Figure 4.11a). We suggest that a potential reason for this result was the novelty of the menu and most participants' lack of experience with larger displays and magic lenses. Therefore, we asked 13 of the participants to perform an additional repetition of the six process tasks using T+RM to evaluate potential learning effects. We statistically analyzed the difference between their first and their second iteration with T+RM (normally distributed, ANOVA with Bonferroni correction). They were significantly faster ($F_{1,12} = 13.39, p \leq .05$) in the second

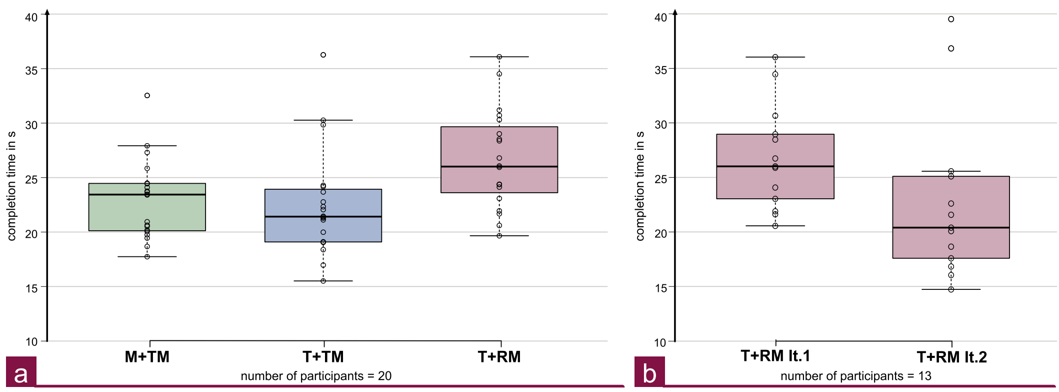


Fig. 4.11.: Box plots using quartiles with Tukey whiskers[3]: T+RM is significantly, albeit in regard to absolute time only slightly, slower than M+TM and T+TM (a). However, the second iteration of T+RM is significantly faster than the first (b).

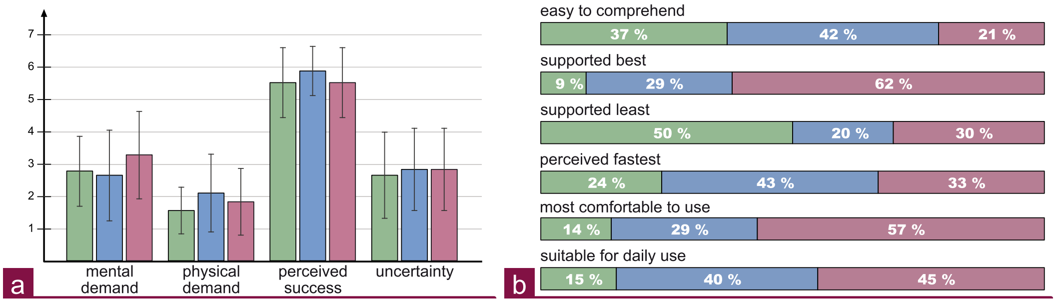


Fig. 4.12.: Questionnaire results: a) NASA TLX-based questions on demand and perceived success, as well as b) questions regarding subjective evaluation of techniques for M+TM, T+TM, and T+RM.

run ($M = 22,79\ s$), with an average of 15.2% increase in speed (see Figure 4.11b). This shows that experience is an essential factor in this scenario. While with further training all conditions would likely improve, the interaction principles of traditional menus are established, so that the strong improvements with the novel condition suggest that the differences between conditions might even disappear.

The evaluation of the questionnaires regarding demand (similar to NASA TLX) found no significant differences (ANOVA) between the conditions regarding mental or physical demand, perceived success, or uncertainty and frustration (see Figure 4.12a). While no assertion can be made, we see no evidence for a difference in perceived demand between techniques. In the final questionnaire, the participants were asked to compare the three conditions. User answered with very mixed results when asked about easiest technique to understand, technique with best suitability for daily use, and assumed speed. However, 62% of participants selected T+RM as the condition they felt supported them best. Comfort of use was also considered best with T+RM by 57% of participants. When asked to guess the fastest of the conditions, 43% of the participants selected T+TM, 33% T+RM and only 24% checked M+TM (see Figure 4.12b). Although we tried to hide which of the conditions were our own design, we cannot rule out a possible bias due to the novelty effect which might have influenced the participant's selection.

Beside the questionnaire, we used the video data and our own observations during both the *process* but also especially the *exploration tasks* to further assess the participant's interactions, concentrating on our T+RM technique to gain insight on how participants worked with the lenses. One aspect of T+RM, which we frequently observed, was that participants had problems identifying parameters based on their icons. They needed to expand the handle to be able to read the label, often resulting in trying out several handles before discovering the right one. This may have been partially responsible for the slower task completion times.

[3] The box plots were created using http://boxplot.tyerslab.com.

Participants were offered free choice between using tap or drag-snap for manipulating parameters. We observed that of the 22 participants, 14 mainly used the *drag-snap technique*, only three used *discrete tapping* primarily and five participants frequently switched techniques. We further noticed that using tap seemed to coincide with situations where participants were unsure on how to proceed or which parameter to select. This assumption is also supported by the questionnaires, where the three users of *discrete tapping* belonged to the group which had little to no experience with large touch displays, graphs, and magic lenses.

In all three conditions, some tasks required the use of two lenses explicitly stating to move and adjust both lenses individually. In the exploration tasks, two lenses were always already placed at startup and could be used freely. We noticed however that when given the choice, 16 of 22 participants used only a single lens, combining various lens functions instead of assigning a unique function to each lens.

4.5 Discussion and Summary

In this chapter, we proposed multi-functional multi-touch lenses that unify functions for diverse purposes of graph analysis into one tool. To enable a fluent workflow with this tool, we integrated different ways of interaction using both a widget-based menu approach and an expert way with continuous gestures. When evaluating the radial touch menu against a traditional menu version used with either touch or mouse, the study helped us identify the limitations of MULTILENS showing that it was slower than the traditional menu. However, differences in terms of overall time were small and more experience could possibly eliminate these differences (as was evident in our second iteration). Nevertheless, in the qualitative questionnaires the users assess MULTILENS with positive values in regard to their personal preferences and overall comfort of use. Furthermore, there are a range of clear advantages to MULTILENS when applying the multi-functional lenses to large displays and/or multi-user scenarios. Firstly, the parametrization of the MULTILENS is lens-dependent and local, so that each lens has its own manipulations and, as opposed to the traditional menu, no selection is necessary to change the focus of the global menu. This allows a range of users to manipulate and adjust their individual lenses without interfering with each other. Secondly, all interaction take place directly around the explored data, minimizing the distraction resulting from gaze switches and enabling improved immersion into the current task.

When comparing the widget-based and gesture-based approach, there is need to evaluate the continuous gestures in long term use. However, for these varying ways of interactions in the MULTILENS concept the general principles of *recognition vs.*

recall apply. That relates specifically to the disadvantage of having to remember the gestures instead of perceiving buttons in the menu and is especially relevant when considering the use of the concept by either **novices or expert users**. As such, the continuous gestures are very comparable to keyboard shortcuts which need to be learned and are difficult to remember in the beginning. For the future of the gesture-based approach, an easier transition and low-level entry point into their use should be considered and investigated. On the other hand, our radial menu approach already aims to support this transition from novice-like, discrete tapping to a more fluent, expert-oriented interaction within the menu. By designing the drag-snap slider technique, we specifically focused on allowing this more fluent workflow that often makes up expert use. We think that providing tools with both smooth widget-based and gestural interaction capabilities yields improved interaction and has potential for efficiently supporting the work of a larger range of users.

Our study indicated a preference to using one tool and extending its functionality instead of using multiple independent lenses. This raises the questions if in this analysis context the use of a single tool (rich in functionality and complexity) is generally to be preferred to multiple single-purpose tools. While we do not aim to answer this general and broad question, we believe that by designing MultiLens, we extended the rather static lens interaction to a tool that supports **flexibility in parametrization and combination** of lens functions, thus creating an advanced gadget for diverse exploration tasks which could be used with a large set of possible lens functions (cf. chapter 3).

To summarize, we contributed MultiLens as novel multi-touch-enabled visualization lenses that provide multiple lens functions within one generic lens tool. As previous work focused on single-function lenses, we applied existing principles and improve on the interaction with lenses considering a) basic interactions, b) activation of different lens functions, c) the adjustment of function-dependent parameters, and d) the combination of lens functions. In contrast to traditional approaches, our lenses support seamless in-place manipulations and adjustments which contribute to the user's flow of interaction. Both our interaction approaches are designed to enable fluent interaction in uninterrupted phrases and support redundant interaction alternatives for varying preferences of users. Furthermore, by enabling the flexible combination of lenses, MultiLens yields entirely new lens functions. With this, we contributed another step in using the potential of interactive surfaces for information visualization by presenting a flexible tool with different interaction alternatives which is one of the goals of this thesis.

5 Graspable Mobile Lenses at Large Vertical Displays

Contents

The previous chapter with its touch-enabled MULTILENS presented a contact-based interaction technique to adapt lenses to the current need of the user by manipulating lens function and function parameters efficiently. This could equally be interesting for large display setups, both tabletops and wall-sized displays, taking into account the benefits of large display space (as described in chapter 2). Adding lenses to these setups may become especially useful for collaboration considering the characteristics of the lenses per se (e.g., their spatial restriction) but also MULTILENS's local, in-place manipulation that does not require global menus. However, large displays also encourage additional movement to let the user perceive all content on the display. This physical navigation [BNB07] has already been shown to improve sensemaking and support data exploration including the change of visualizations on movement [Jak+13]. To use lenses in these setups hence requires the possibility to simply move the lens with the user and to be able to configure it while moving around. Building on the benefits of tangible user interfaces and spatial interaction, this chapter describes the use of graspable, tangible objects as representations of magic lenses. It starts with an investigation into different setups for lenses (section 5.1) questioning what type and properties of tangible in combination with context displays could function as a beneficial lens setup. Applying mobile devices as tangible lenses, the chapter then elaborates on mobiles as personal toolboxes for data analysis (section 5.2). GRASP proposes an interaction design that places the graspable representation of the lens into the user's hand . Thereby it addresses the challenge of configuring the lens according to the user's current need (C1, page 72). It further focuses on integrating the tool into an effortless workflow (C4) by being designed to enable flexible positioning in front of the wall-sized display. A qualitative study (section 5.4) evaluates the proposed interaction techniques and specifically observes participants' workflows and distribution of focus between the two devices.

The research on spatially-aware mobile devices as lens representations (5.2) has been published in the following work:

Ulrike Kister, Konstantin Klamka, Christian Tominski, Raimund Dachselt. 2017. GraSp: Combining Spatialy-aware Mobile Devices and a Display Wall for Graph Visualization and Interaction. In *Computer Graphics Forum (CGF), Vol. 36, No. 3* (June 2017). pages 503-514, ISSN: 1467-8659.

Parts of the discussion of transparent tangibles (5.1) have appeared in research published as:

Wolfgang Büschel, **Ulrike Kister**, Mathias Frisch, and Raimund Dachselt. 2014. T4 - transparent and translucent tangibles on tabletops. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces* (AVI '14). ACM, New York, NY, USA, pages 81-88.

5.1 Considerations for Tangibles in Lens Applications

The general idea of using tangible user interfaces [IU97] to transfer digital content or controls to the physical world creates a range of advantages (see chapter 2.1.2) which have also been shown to benefit data exploration and the control of parameters. As described in chapter 2, we discuss tangibles in terms of graspable, movable objects independent from their output capabilities. That is, we also extend the term to mobile devices for both tangible and spatial interactions. Since these kind of tangibles have been proven worthwhile specifically for controlling parameters (e.g., [Jet+11; LAD14]), it may similarly be an advantage for data exploration tools including magic lenses. However, the question remains which type of tangibles can suitably be applied to lenses and in what way lens interaction may be supported. The following will shortly discuss the possible dimensions for tangible lenses (see Table 5.1).

Context View's Setup, Orientation, and Relation to Tangible Magic lens applications require both a focus and a context view. As such, the use of tangible lenses needs a context view as a basis for interaction. While analog, real-world contexts (e.g., a tracked physical map or a physical space) may be used for augmented reality lenses (e.g., applying our principle of *contact Augmented Reality* [Hin+14a]), this

Dimension	Options		
context representation	*real-world content* i.e., tracked physical object or space	*digital content* i.e., display surface	
orientation of context	*horizontal* e.g., tabletop display	*vertical* e.g., display wall	
relation of tangible to context	*in-contact* i.e., placed or stuck on the display	*in-space* i.e., above or in front of the display	
tangible's input sensing	*no input sensing* e.g., passive tangibles	*input through context* i.e., context interprets input	*input capable* e.g., multitouch-enabled
tangible's output capability	*without visual output* e.g., passive, opaque tangibles	*see-through* e.g., rings or transparencies	*display* e.g., mobile devices or projections
role and purpose of tangible for lens	*lens representation* i.e., tangible embodies lens	*lens controller* i.e., tangible manipulates lens	

Tab. 5.1.: Dimensions regarding the use of tangibles and mobiles for lens application. Colors indicate the categorization of the following tangible lens concepts (transparent tangibles and GRASP, as well as DI.VI.CO in the next chapter).

chapter focuses on the more common and dynamically interchangeable use of additional displays to provide the context visualization. To accomplish the focus view within the context, this context display needs to surround the focus area (or at least extend it to multiple sides) and thereby needs to be appropriate in size. As Langner et al. [LHD18] have shown, there is potential in using multiple mobile displays placed next to each other in use as overview and detail views. For lenses however, larger display space is more appropriate to support the use of tangible lenses and their movement *within* the context frame. As a result, in this context both tabletops, i.e., horizontal displays, and display walls, i.e., vertical displays, should be considered. Furthermore, the relation of the focus view to this context display is of importance: The tangible can be used *in-contact* with the context visualization (e.g., placed on an interactive tabletop [KE12; Ebe+13]), but can also be lifted from the surface and used *in-space* (e.g., in front or above the context display [SSD09; Spi+10]).

Capabilities and Role of the Tangible for Lenses There are a range of different types of tangibles with varying capabilities that need to be considered for their use in lens applications. An especially important factor for this use is the tangible's output capability which is relevant for representation of the lens' output. Traditionally, tangibles have been used in form of passive and opaque objects (e.g., [IU97; SH10; UIJ03]) which may also include transparent tangibles whose markers render their transparency void (i.e., where transparency is used for aesthetic purposes, e.g., FacetStream [Jet+11]). These tangibles can hardly represent the lens with its focus area, but can be used as a handle to the lens or as a separate *controller* for lens manipulation (similar to parameter adjustment for non-lens related contents). More interesting however are tangibles that support an additional content output and thereby allow possible *representation* of the lens, i.e., the lens itself becomes tangible and the entire tangible therefore embodies the lens. This can be accomplished using opaque rings or transparent tangibles (e.g., [Ebe+13; KE12]), where the context display provides the lens content, or using mobile devices, including small display cubes (e.g., used for querying [LAD14]) but also smartphones and tablets or similarly tablet-sized cardboards with projection (as used for PaperLens [SSD09]). These support an active lens output separate from the context visualization and thereby allow a larger range of freedom in terms of movement and personal output visualization. How the interaction with the lens is accomplished depends on the tangible's input sensing capabilities. For passive tangibles on tabletops there is no additional input sensing beside tracking their position. However, taking sensor-enhanced tangibles or mobile devices into account may result in actual input-capable lens representations or controllers that take advantage of the variety of these input possibilities for lens configuration.

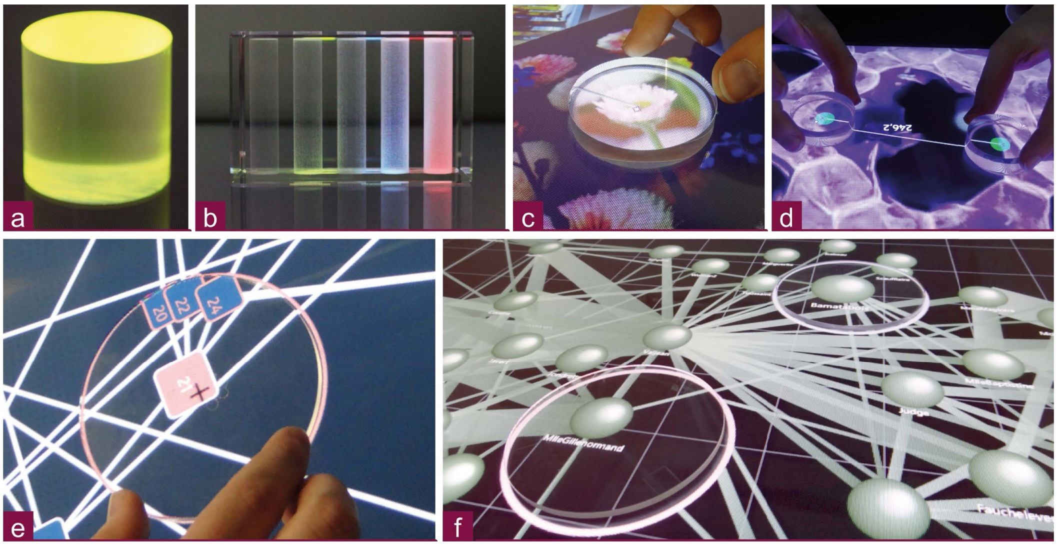


Fig. 5.1.: Translucent and transparent tangibles provide possibilities for lifting content from the display surface (a-b). They can be used as lenses (c), and support precise positioning through their tangibility (d). As an example, we applied these tangibles to various lens functions supporting graph exploration (e-f).

To summarize, when considering tangibles and mobiles for application as lens tools there are certain dimensions that may be considered including the way the context visualization is represented, its orientation and relation to the tangible, the tangible's output and input capabilities, as well as its role and purpose in regard to the lens.

To illustrate the previous categorization and show the diversity of tangible lenses, the following shortly describes an example that we used to explore the design space of tangibles for lenses. As one example for *lens representation in-contact* with the context visualization on a *horizontal* tabletop display (see purple color in Table 5.1), we[1] investigated the use of transparent tangibles whose *see-through* capability supports visualization of the lens content within the tangible. Similar to existing works, we see the advantage of making the lens itself graspable as has previously been investigated for the MetaDesk lens [UI97] (where *in-space* use is considered) as well as for thin transparencies used as Embodied Lenses [KE12] and TangibleRings [Ebe+13] that combine rings of varying size. In addition to these previous works, our work provides a general design space of transparent and translucent tangibles including a survey of form factors, materials, role and function, position of visualization in regard to the tangible, as well as interactions (see Figure 5.1). We evaluated precise positioning comparing transparent tangibles to touch interaction and opaque tangibles and presented evidence of the advantages of positioning these tangibles in contrast to touch. Furthermore, these tangibles support touch on the tangible which is recognized by the context display (*input through context*) and can be used to interact with content in the lens' focus. In our work, we provide various application

[1]The investigations into transparent tangibles were done by the author Ulrike Kister together with co-contributors Wolfgang Büschel, Mathias Frisch, Ricardo Langner, and Raimund Dachselt.

examples and prototypes. Transparent plates and tokens (see [Büs+14] for size categorization) can be used as tangible graph lenses to provide graspable tools, e.g., for bringing in adjacent nodes (Figure 5.1e) or reducing edge clutter (Figure 5.1f). Due to their tangibility, multiple lenses can be placed and moved on the surface simultaneously.

Towards Generic and Flexible Tangibles for Lenses

The previous example of transparent tangibles are very fixed in terms of material, shape, and size. The need for these specific tangibles to enable the use of lenses is certainly a disadvantage for tangible user interfaces which may be one of the reasons they have not established themselves in everyday use. Mobile devices, however, have already become our personal tangibles: With their diversity in applications they provide generic tools for very diverse tasks. Their tangible properties and active displays in addition to their flexible usability raises the question if they could also be beneficial for use as lenses for data exploration.

The idea of using tangible displays as *lens representations in-space* above a tabletop has been investigated by Spindler et al. in a variety of works where lenses were realized as *displays* using top-projection on cardboard [SSD09; SMD12; Spi+14a]. These tangible views have also been applied specifically to information visualization presenting the potential of this use for data exploration [Spi+10]. Building on these work, the research presented in the following chapter uses mobile devices and considers their application to large vertical display spaces for multiple users. For the application to lenses, it is important to keep focus and context in relation to each other. This was automatically the case above a tabletop as the dimension of the table and the user's arm reach in height are restricted. However, the relation between views becomes much more of a concern for wall-sized displays where the possibility of movement in front of the display drastically increases and may hence result in more prominent attention switches between views. By spatially tracking the mobile devices, the mobile display's content and focus can be directly related to the content on the large display making the connection. This enables a strong coupling of the mobile device to the context view – putting the lens into the user's hand.

The following project GRASP examines this device combination. It investigates mobile devices as *lens representation* at large *vertical* displays (see blue highlight in Table 5.1, page 99). While the large display presents the context visualization of which parts can be selected, the mobile device provides detail views and additional tools for exploring the data. This work aims to investigate the following questions: i) How can tangible, mobile devices support exploration tasks for graph visualization at wall-sized displays? ii) How do people perceive and explore graph data with a

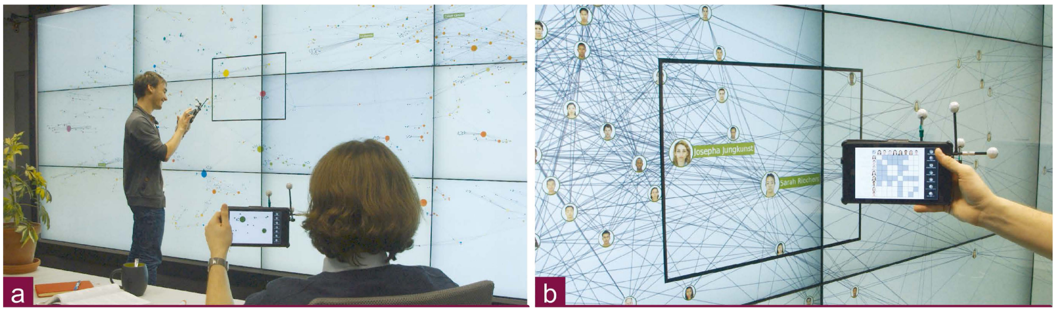


Fig. 5.2.: GraSp provides each user with an individual personal toolbox that supports selection from any position in front of the display wall (a) and enables data exploration with different lenses (b).

mobile device that incorporates a range of exploration tools? iii) Where do people position themselves when given the choice due to the separate mobile display?

5.2 GraSp: Interactive Mobile Lenses at Large Vertical Displays

Mobile devices have become widespread in use and users know them particularly well as their personal devices that incorporate touch input capabilities while mostly containing personal data and tools. While these devices allow visualization and data analysis to become mobile and enable direct interaction with the data, there are a range of well-known limitations [Chi06] of these mobile devices which mainly sum up to the lack of sufficient display space which restricts the amount of information that can be shown. However, this is where large wall-sized displays have the advantage and can provide plenty of space for data visualizations. There have been several works presenting the potential of large display spaces for visualization, interaction, and sensemaking [AN13; Jak+13] (see also chapter 2).

With GraSp, we[2] investigate new ways of combining the advantages of mobile devices and large displays. Our goal is to create a visualization environment that uses the enormous space of large displays and adds to this the flexibility, display capabilities, and interaction modalities of mobile devices. For adding these additional devices, novel software solutions need to be considered to enable the communication and synchronization of their content [Zad+14; BFE15]. To relate the content on the mobile with that on the wall-sized display, a connection has to be initiated. In previous works, this connection between devices has been made via touch [Zad+14], touching the mobile device to the other display [Sch+10a], or by tracking the device in space (e.g., [MHG12; LHD18]).

[2]'We' in this section relates to the author Ulrike Kister, as well as Konstantin Klamka, Christian Tominski, and Raimund Dachselt as co-contributors on this research.

To enable data exploration, we use the wall-sized display as a context visualization giving an overview of the information space while the mobile devices function as a personal toolbox to explore, analyze, and manipulate details in the data. This makes use of the large display space the display wall provides while also relying on the personal character of the mobile devices. Building on top of tabletop work in combination with mobile views [Voi+09; SSD09; Spi+10], we apply a strong coupling of both devices and use the spatial relationship of the devices as the definition of the mobile display's content. In our work, mobile devices are used as personal displays, where detail views, focus points, and additional information are presented to the individual users. Their position, orientation, and movement (and thereby an estimate of the user's position) is tracked in the space in front of the wall-sized display so that individual interactions with the mobile devices can be linked and related to specific objects on the display wall. This supports both natural interaction using touch at the display wall as well as remote interaction through the mobile from a distance. However, instead of enforcing one particular input style, our solution has been designed so as to support flexible user position in front of the large display. In related work, Jakobsen et al. [Jak+15] found users preferred touch over remote interaction for mid-air gestures. Participants in their study only chose mid-air gestures if interaction cost, i.e., movement, was entirely too immense. The combination of wall-sized display and mobile devices allows touch on both devices and thereby is not as limiting as mid-air gestures and very flexible in terms of display output. We do not want to enforce a restrictive movement of users to stay in contact with the wall-sized display or require physical navigation [BNB07] and hypothesize that results will be different from Jakobsen et al. [Jak+15] because of this additional direct touch and output device. Further, our goal was to adapt the interaction facilities to the users' workflow. As such, we aimed to support flexible movements and remote interactions, i.e., interactions when being far from the display wall either during movement or from a standing or sitting position. This way, we enable real-world multi-user scenarios such as meetings and discussions of data analysis, and permit individual user behaviors independent from a specific distance from the wall-sized display (see Figure 5.2a).

Scenarios and Tasks

Our goal was to design techniques that demonstrably benefit from the new technology. Yet at the same time, the techniques should be generalizable and practically relevant. To obtain these generalizable techniques, we apply a set of the previously described task taxonomies and reference them as discussed in chapter 2.3.2 (see Appendix A.1 for an overview). That is, E describes exploration tasks categorized by Yi et al. [Yi+07] which are refined by graph-specific categories G defined by Lee et al. [Lee+06], while manipulation tasks M are added according to Gladisch et al. [Gla+15b]. To achieve practical relevance and gain an understanding of possible workflows, we consider two concrete application scenarios. Both scenarios were devised, discussed, and iterated under consideration of existing work on visualization and further interaction tasks for graphs [PPS14; Wyb+14].

Scenario I "Disease co-occurrences" The first scenario focuses on finding cancer relations through common appearances of diseases using a graph data set of disease occurrences (see human disease network in section 2.3.1). Nodes represent diseases and genes, whereas edges present the disease co-occurrence or a disease's relation to a gene. We assume a group of biologists meet to discuss the research topic of colon cancer. They sit together jointly discussing possibly interesting aspects in the data while also looking for detailed information individually. In a first phase of hypothesis generation, the biologists examine if colon cancer occurs in relation with other forms of cancer or generally with diseases of indigestion. In a hypothesis verification phase, the biologists want to study in more detail the symptoms and diagnoses of diseases that are connected to colon cancer. Experienced participants of the group add and update the graph with data from a recent publication. This scenario incorporates the following tasks:

- Find & select the cluster of cancer diseases
  (E [SELECT/EXPLORE]; G [OVERVIEW])
- Find & select all neighbors of colon cancer
  (E [CONNECT]; G [ADJACENCY])
- Filter neighbors by type of disease
  (E [FILTER]; G [ATTRIBUTE-BASED TASKS])
- Compare the number of diseases per type
  (E [EXPLORE]; G [ATTRIBUTE-BASED TASKS])
- Identify nodes connected to colon cancer
  (E [CONNECT]; G [ATTRIBUTE-BASED TASKS])
- Access details: symptoms and diagnoses
  (E [ABSTRACT & ELABORATE]; G [ATTRIBUTE-BASED TASKS])
- Add and update a disease, relations, and attributes
  (M [ADD NODES]; M [ADD/ DELETE EDGES]; M [ADD/UPDATE ATTRIBUTES])

Scenario II "Advertisement in a social network" In our second scenario, a social network is investigated to find people with wide-spread influence in order to use them for product endorsement and to send them advertisements for a specific new product. There are two different types of nodes: people and product/fan pages. Edges exist between friends as well as between people and certain products or fan pages they liked. In this scenario, an advertisement consultant first identifies cliques of certain size and appropriate target age and gender. She then explores these groups in more detail finding the ones with a sufficient number of neighbors and an active engagement with product pages before finally adding a connection from the most-likely candidates to the product. This process includes the following tasks:

- Filter people by age and gender to see target group
  (E [FILTER]; G [ATTRIBUTE-BASED TASKS])
- Identify cliques of people by size and product engagement
  (G [CONNECTIVITY]; G [ATTRIBUTE-BASED TASKS])
- Separate products and people
  (E [FILTER]; G [ATTRIBUTE-BASED TASKS])
- Analyze common friends
  (G [COMMON CONNECTIONS])
- Examine specific attributes of candidates
  (E [ABSTRACT & ELABORATE]; G [ATTRIBUTE-BASED TASKS])
- Add edge from selected candidates to new product
  (M [ADD EDGES])

GRASP Principles

While the wall-sized display may support interactive exploration and manipulation of the main visualization, this chapter's focus lies on combining mobile devices in this setup and providing a rich set of individual local interactions on each user's personal mobile device. The mobiles are meant to bridge the gap between display wall and user by presenting a rich set of views onto the data, including

- a close-up excerpt from the data on the wall,
- the visualized data with adjusted level of detail,
- alternative representations of the data in focus, or
- User Interface widgets and setting dialogs.

Therefore, we introduce an interaction repertoire of techniques called GRASP, which stands for "Graphs in Space" and incorporates the graspable capabilities of the mobile device as a personal toolbox. Applying research on proxemic interaction [Gre+11; Jak+13], we use the distance and orientation of the user's device to trigger minor adjustments of detail level, focus views, and tool size. However, as a design principle we define the need for *explicit interactions* for major manipulations, such as view changes. As a result, to switch between these views and the diverse features of the mobile toolbox, we suggest the use of an explicit tool menu on the mobile display where users can select the appropriate view.

In the following, we focus on providing lens-like solutions on the mobile device that cover the tasks identified in the previous section including exploration and selection, details on demand, connectivity and adjacency tasks through alternative

Tasks	GRASP techniques
E [EXPLORE], G [BROWSING TASKS]	• physical navigation • mobile focus view selection
E [SELECT]	• individual data and group selection – on display wall – on mobile device – using pointing selection
E [ABSTRACT & ELABORATE]	• labeling technique on movement • details on demand after selection
M [ADD/ UPDATE NODE ATTRIBUTES]	• manipulating attributes in detail view
E [CONNECT], G [ADJACENCY, CONNECTIVITY]	• tangible Adjacency Matrix Lens • tangible Bring Neighbors Lens
M [ADD/DELETE EDGES]	• tap on cells in adjacency matrix
E [FILTER], G [ATTRIBUTE-BASED TASKS]	• tangible Attribute Filter Lens • body-relative range filtering • distribution overview with sieve filter tool

Tab. 5.2.: Overview of tasks and respective GRASP techniques.

representations, as well as diverse filtering techniques. Table 5.2 provides a brief overview of these solutions.

5.2.1 Mobile Focus View and Data Selection

GRASP explores data analysis in a simple multi-device environment with large wall-sized display and additional mobile devices. This requires a distribution of content onto the separate devices. Focusing on the principle of magic lenses, we decided that the wall-sized display presents the main visualization and thereby the context for all other interactions. As a result, users have to select the current focus view for their personal mobile device. The following presents the possible alternatives to define this focus view and then discusses how to actively select data either directly at the display wall or through the mobile focus view before elaborating on how the mobile device is used as a physical representation of the lens and its content.

Mobile Focus View Selection

When working with a mobile device, the user first has to determine on which part of the data to focus (E [EXPLORE], G [OVERVIEW]) and hence what part to show on the mobile display. This can be interpreted as defining the lens selection (cf. section 3.1).

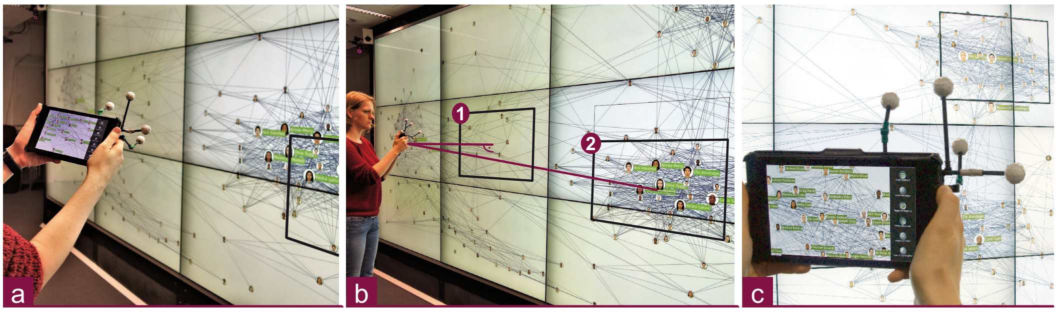


Fig. 5.3.: The mobile device shows a selection of the context (a). Pointing the mobile device towards the display supports active movement of the lens selection either using only its position (orthogonal pointing, b1) or also its orientation (perspective pointing, b2) allowing flexible positioning and reachability even from afar (c).

As any tangible object, geometric properties of the tablet, such as shape and size, remain the same over the time of interaction. However, as the mobile has an active display for output, the size of the lens selection can be adjusted since a larger or smaller extraction of the context can be shown for modification through the lens. Furthermore, changes of the other geometric properties, i.e., position and orientation, need to be considered for defining the lens selection. For that, the mobile device's absolute position in space and its orientation can be used to allow pointing towards the content to select a rectangular focus region of the data for visualization on the mobile. Visual feedback on the wall display indicates the current region of interest which is transfered to the mobile display (see Figure 5.3). The feedback is designed to be minimalistic and unobtrusive to reduce interruptions of other users.

When pointing at the large visualization, alternative mappings are possible: orthogonal and perspective pointing (or their combination [Lan+16]). For orthogonal pointing, only the position of the device and its orthogonal projection to the wall display are used. While this requires body movement in front of the display, it allows a flexible hold of the device and thereby precise, more stable selection on the display wall. In perspective mode, the orientation of the device is used to point at the display. However, we correct the rectangular focus to avoid distortions. As a result, the perspective mapping allows a farther reach of elements on the wall using only a tilt of the mobile device, even when sitting at a distance to the display (remote interaction). A button press can switch between the perspective and orthogonal pointing techniques to adapt to the user's current goals and take advantage of both techniques. Similarly, pointing can also be frozen to focus and fixate a region of interest (similar to [Spi+10]), so that the user can focus on the lens content and may relax their posture. The amount of visible content on the mobile depends on the zoom factor set on the device. It can be adjusted through a slider, a pinch gesture on the display, or using the measured distance from the mobile to the wall (similar to previous work on proxemics without additional devices [Jak+13]). We implemented all of these mappings and features to care for the diverse tasks and scenarios. We believe that especially the perspective pointing allows for very flexible reach of elements independent from the current position of the user.

Data Selection and Details on Demand

As the user moves in front of the wall-sized display to explore the data space, different techniques are required to support selection of visualized graph data (E [SELECT]) both in close proximity and farther from the display wall. We therefore propose a number of techniques for selection directly on the display wall, directly on the mobile device, or from the device by pointing. To find a starting point on where to select and explore the data, the user requires some basic information to identify interesting structures or read individual labels. We recommend a smart labeling technique that shows nodes at different levels of detail (incl. labels, images, basic attribute data) on the wall-sized display depending on their distance to the user. This is accomplished using a level of detail value which invokes different visual representations of the nodes. It is set to higher values for nodes close to a user depending on their euclidean distance and their degree relative to the graph's maximum degree. When multiple users influence the same node, i.e., users are in close proximity, the higher value is used. When near the wall, users can tap to select individual elements on the display (see Figure 5.4a). Furthermore, users can select multiple nodes by encircling them (see Figure 5.4b). Selections are managed on a per-user basis. When performed on the wall, they are associated with the device closest to the location where the selection took place.

To examine the entire information space or select large parts of it, the user may step back and move in front of the display or casually sit at a distance to discuss the data with colleagues. As a result, contact with the display is not always possible or wanted. Users can apply different selection techniques for remote interactions and during physical navigation in accordance with our design principles, keeping the interaction consistent with the techniques on the wall-sized display as an essential interaction guideline: It is possible to tap elements on the mobile to select individual nodes or encircle a group of elements to select them (see Figure 5.4c). Beside these tap and lasso selections, a rectangle selection of all elements currently visible on the mobile can be triggered. This is especially useful for selecting clusters. Afterwards, individual elements of the selection can again be deselected using tap on the mobile device. Elements currently selected on the mobile are highlighted by color on the wall-sized display to ease rediscovery when looking at the display wall.

As the user aims to get more information on the content after selection, the role of the mobile device immediately changes to a second screen for additional visual output. While the wall display presents an overview of the nodes and edges, the mobile device may display a detail view with more information associated with

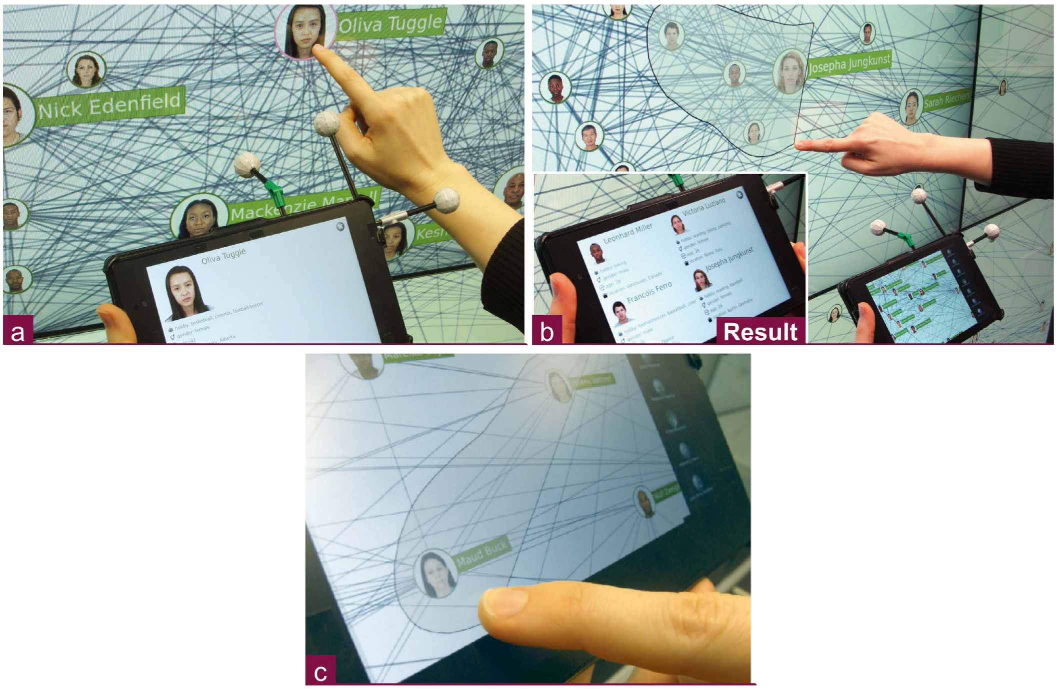


Fig. 5.4.: To select elements and open associated details, the user can tap (a) or encircle (b) elements on the wall-sized display. The same techniques can also be used on the mobile device (c).

selected content (E [ABSTRACT&ELABORATE]). When a single node is selected, detailed information of the object are shown by presenting an organized table of the multi-variate data attributes. For a group of nodes in the social network example, we currently provide individual information on all selected nodes in a grid overview (see Figure 5.4b). As the nodes in the disease scenario contain more numerical data, aggregated visualizations of information on graph attributes may be shown and could be extended to other alternative representations for comparison (e.g., star plots or parallel coordinate plots) or the specific node-link relations within the selected group (different layouts of sub-graphs, etc.). In addition, the second screen can be more than just an output device. A characteristic of mobile devices is their suitability for personal input, including the advantage of providing a familiar personal keyboard. They can hence be used for manipulating data, e.g., to edit data values and enter new data within the detail view (M [ADD/UPDATE ATTRIBUTES]).

5.2.2 Tangible Graph Lenses

Within our GRASP concept, we decouple the main visualization from detail views, as described before. This can be extended for focus and context views, decoupling and distributing the views on the two devices. As a result, the mobile device showing the focus view becomes a tangible graph lens. It can be argued that this decoupling of both views creates a separation that should be considered overview and detail (cf. 3.1). Nonetheless, the mobile moves within the space in front of the wall-sized visualization and we believe both to be in the user's line of vision. Aspects of angular size and content coordination need to be considered here to address possible attention switches [RNQ12a]. However, because of the synchronized movement of mobile and context selection, we see a strong sense of connection between the mobile view and the context view which encourages our understanding of this combination as part of the magic lens principle.

As described for the mobile focus view selection, we restrict the lens to be of rectangular shape to fit a typical tablet screen. While the absolute size of the mobile device is fixed, the relative size of the lens (i.e., the amount of graph elements within it) can be adjusted analogous to the manipulation of size while pointing, either using a slider on the mobile, by pinching on the display surface, or through the user's distance from the wall-sized display. This definition of the region of interest is consistent with our previously described concepts for pointing and hence again allows both physical navigation as well as remote interaction from a casual position – creating a perspective tangible lens tool. The lens movement can be frozen and decoupled from the device's pointing position using a button on the device (similar to [Spi+10]), so that the user can focus on the lens content and may relax their posture. It can also be used to move around and actively explore the data set with a given lens function and identify regions of interest.

As we have seen in the previous chapters, lenses support diverse tasks for graph analysis. While a whole range of lens functions is possible, we chose a selected view for application in GraSp which relate to varying interaction tasks including *filtering* nodes by attribute data, identifying the *connectivity* and *adjacency* of nodes in focus as well as changing the *encoding* of the visualized content and *manipulating* relations. We used three existing lens functions for application to the GraSp concept and by making them tangible lenses on the mobile device identified extensions that will be described in the following.

Bring Neighbors Lens To support connectivity and adjacency tasks focusing on the relation between nodes, we implemented a variation of the Bring Neighbors Lens [Tom+06]. By pulling in the adjacent nodes of all nodes in focus, the Bring Neighbors Lens highlights their inherent connections (E [CONNECT]; G [ADJACENCY]). As a rule, the manipulation of the lens function happens only within the bounds of the lens and for GraSp by extension only on the mobile. While the Bring Neighbors Lens in its original definition has a global effect, this is eliminated by the additional copy of the graph on the user's mobile. As a result, the nodes are only pulled in on the mobile device while the context visualization remains intact (see Figure 5.5a). This eliminates the problem of interrupting other users interactions with the graph,

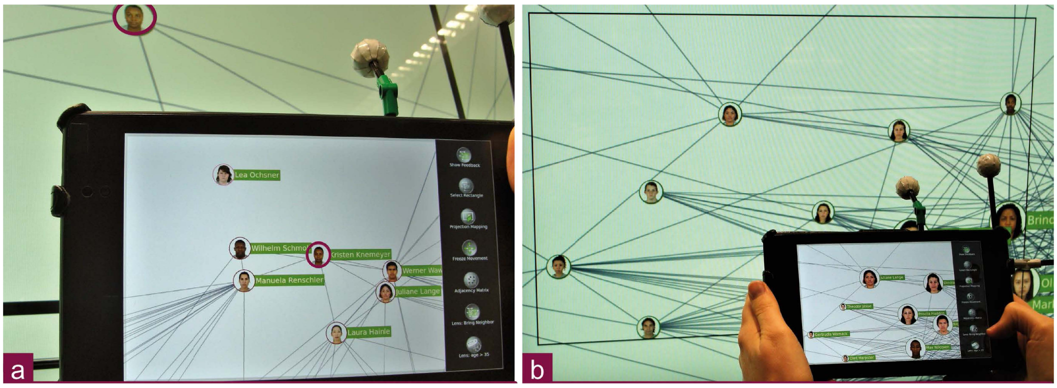


Fig. 5.5.: Lens functions can be applied on the mobile while the context remains intact: Bring Neighbors pulls in adjacent nodes of those in focus (highlighted in purple) (a) and the Attribute Filter enlarges nodes that fit the defined criteria (b).

but requires additional effort in synchronizing manipulations. Additionally, the user might be interested in where certain neighboring nodes might be located on the context display which requires additional interaction with the proxy element on the mobile, e.g., a tap on the proxy triggers an animation of the associated node on the wall-sized display.

Adjacency Matrix or NodeTrix Lens Alternative representations and different encodings of the graph data can help identify connections or patterns in the topology. By providing those representations on the additional mobile display, they can become movable in space and hence can be flexibly arranged to investigate the coupling with the original representation. Seeing both adjacency matrix and node-link representation simultaneously is worthwhile and can support different tasks [GFC05]. As part of GRASP, we visualize the adjacency matrix for parts of the data similar to NodeTrix [HFM07], but in a physically decoupled way to show both representations at the same time. The adjacency matrix fills the mobile display, while feedback on the wall-sized display shows which nodes are currently included in the matrix visualization. The adjacency matrix on the mobile device allows a direct overview of connected elements by looking at rows or columns of the matrix instead of following individual edges in the node-link representation.

The main focus of the adjacency matrix is to present relations as an overview. However, these edges may also need to be manipulated and configured (M [ADD/DELETE EDGES]) as described in the previous scenarios. When aiming to create an edge on the display wall by connecting two nodes through dragging (e.g., as suggested by users in [FHD10]), moving along a wall-sized display might be tiresome or impossible when other users are occluding parts of the display wall. The decoupled view on the mobile display and the alternative representation allow remote addition and removal of edges in an elegant way. After unlocking the edit mode tapping on a button at the upper left corner of the mobile display, the user can simply tap on a cell

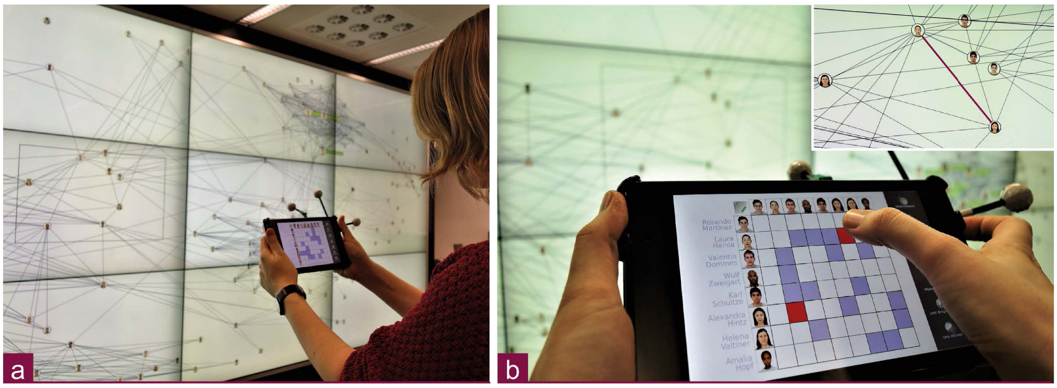


Fig. 5.6.: The mobile device provides a lens with the alternative adjacency matrix representation of the selected sub-graph (a). It can be used to manipulate the graph by adding and removing edges (b) which are propagated to the context visualization.

of the adjacency matrix to create an edge between nodes [Gla+15c]. The creation then also appears on the wall-sized display and the matrix is adapted to reflect the change (see Figure 5.6b). Similarly, an edge can be removed by tapping on the according cell. Further touch manipulations are possible. For example, dragging on multiple cells within a row or column, or even diagonally when selecting blocks of cells, can manipulate multiple edges at once, allowing quick creation of clusters. This interaction incorporates the magic lens principle back into a manipulation tool (as originally designed for Toolglasses [Bie+93]).

Attribute Filter Lens An important requirement of many scenarios regarding data visualization is the support of overview & filter tasks in versatile ways. In the most basic form, this can be the access and further analysis of individual target groups by reducing the visual elements through filtering of attribute values, for example, seeing only young adults for certain product endorsements. For further analysis of the data set, users may want to explore the separate layers and understand the distribution of data points along specific data attributes and dimensions. For our disease scenarios, this relates to separating the types of diseases in an interesting cluster and investigating these separated groups. To address the filtering of content by attribute value, we added to the tangible lens character of the mobile by integrating a configurable Attribute Filter Lens which allows highlighting of nodes with certain attribute values or within a specified value range (E [FILTER], G [ATTRIBUTE-BASED]) locally on the device. To configure the lens and select appropriate attribute ranges, the user can select existing attributes and values using touch on the mobile device (similar to Figure 5.7a). For our social network scenario, Figure 5.5b shows an example lens configuration for highlighting all people older than 35 as a possible target group for a specific advertisement. Note that in contrast to the following techniques, the focus of the Attribute Filter Lens lies in the definition of criteria to create a single set of resulting nodes. However, criteria can be a set of parameters from different attribute types defining this result (e.g., females from Wales older than 35).

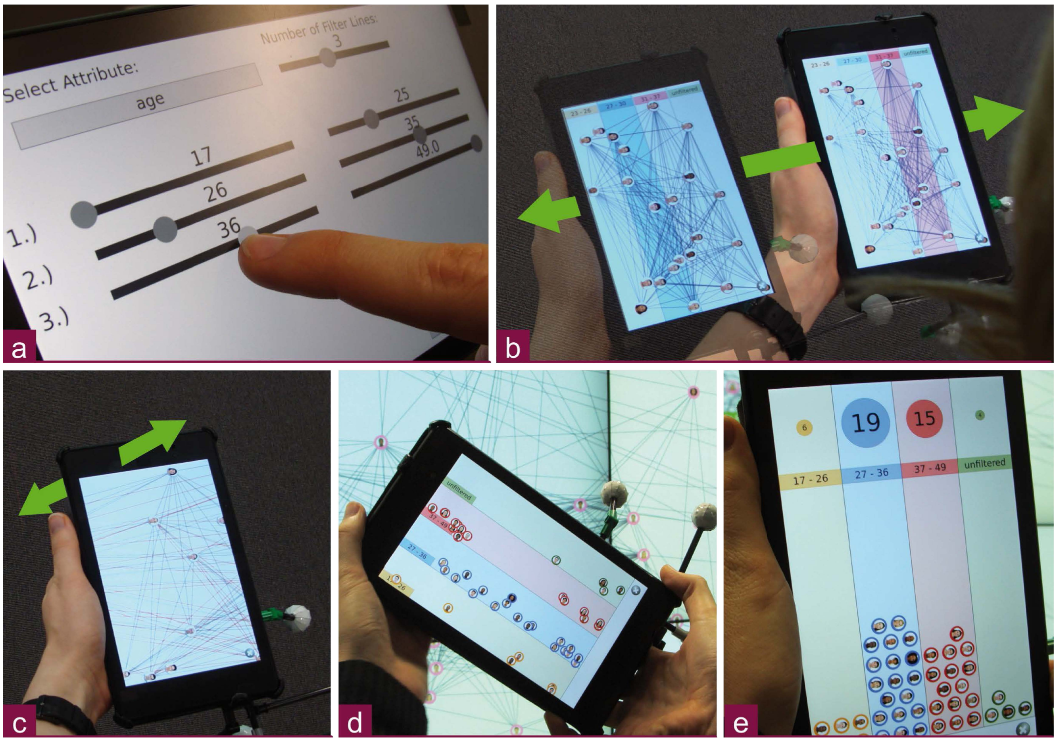


Fig. 5.7.: For analyzing attribute characteristics, users can select attribute ranges (a) to be spatially separated into layers (b). While horizontal movement highlights a range (b), vertical movement shows them in detail (c). Rotating the device lets nodes fall through our sieve tool (d) to give an overview of their distribution (e).

5.2.3 Body-relative Range Filtering and Sieve Filter Tool

As we have seen in previous techniques, spatial tracking of mobile devices can be used directly related to a wall-sized display in various ways. While our wall-centric approaches (cf. tangible graph lenses) refer to the entire context visualization and are characterized by a direct spatial coupling and physical navigation to identify regions of interest, these additional concepts further separate the views allowing the user to explore and filter further details from the data.

We propose the use of body-relative layers which enable decoupled, personal interactions by parameterizing the interaction space around the user (see Figure 5.7b-c). By defining the node attribute in focus and its ranges in an initial configuration step, this 'body-relative range filter' enables newly layouting the nodes on the mobile device. All selected ranges are laid out in the horizontal dimension. Holding a clutch button on the mobile and simultaneously moving the device relative to the previous position creates a spatial degree of freedom which can be used to browse through the layers that represent the ranges of the attribute distribution. Using the vertical movement, the user can then activate a range to expand it and see a more detailed view (see Figure 5.7c). In both cases, the body awareness can help support this browsing of graph subsets (E [EXPLORE], G [OVERVIEW]) and filtering of attribute ranges (E [FILTER]). In addition, we argue that the spatial parameterization facilitates the ability to navigate and remember range positions based on physical mnemonics and human proprioception (cf. [MBS97]).

Another essential task is providing visualizations that enable understanding of the distribution of nodes within an attribute dimension, e.g., the distribution of disease types in a cluster around colon cancer. In order to avoid creating multiple instances of attribute filter lenses and switching between them, we propose using naive physics [Jac+08] to advance understanding of data. This GRASP technique simulates pouring graph nodes through a filter, applying the metaphor of a sieve tool with differently grained holes as representatives of different filter criteria as specific attribute constraints (cf. [RK14]). The sieve filter tool visualizes every selected node as a physical object. A shake gesture allows the user to virtually throw all elements through the previously defined filter barriers formed by the configured attribute ranges. Conforming to the expectations of our metaphor of a sieve tool all nodes are filtered by their attribute and are spatially separated in their visual position (see Figure 5.7d). This allows a fast recognition of the different attribute groups, and the change of layout can be followed easily by seeing generally-known physical behaviors. In addition, the user can rotate the mobile device and all filtered data items fall to the ground within their valid attribute boundaries. This enables a comparative view (similar to [HVF13]) showing the distributions of the previously defined attribute regions by the total amount (absolute height) and with additional labels that highlight the amount of nodes inside each range (see Figure 5.7e).

5.3 Technical Setup and System Architecture

The GRASP prototype was developed using Python with libavg [@lib03] as basis for the user interface, as well as pymunk [@Blo07] for physics. In addition, we used the NetworkX library [HSS08] to handle graph data and integrate graph algorithms. For the social network data set (218 nodes and 1530 edges), we used an anonymized export of a facebook account, which we linked with face images from the Chicago Face Database [MCW15]. The data for the human disease network (1419 nodes and 2738 edges) were provided by Goh et al. [Goh+07]. All data were processed in the GraphML format.

The technical setup consists of the Interactive Media Lab Dresden's large, touch-enabled display wall of 4.86 m in width and 2.06 m in height (frame height at 2.3 m), with a resolution of 7680 × 3240 pixels as well as Google Nexus 7 tablets with attached IR markers as mobile devices which are spatially tracked by the 3D tracking system OptiTrack [@Nat] (see Figure 5.8). In the future, these could be

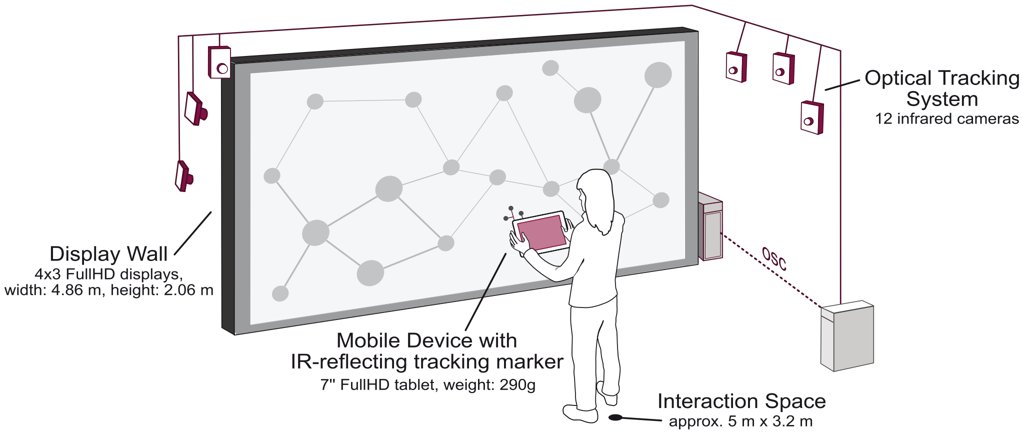


Fig. 5.8.: Technical setup of GRASP prototype.

replaced by mobile devices with on-board tracking, such as Google Tango tablets. All application content is processed on a 2-core Intel Xeon processor, clocked at 3.3GHz and with two AMD HD 7970 graphics cards. Graphics are streamed to the mobile devices using *remote user interface* components designed by the authors of SleeD [Zad+14] and touch input from the device is returned to the central display wall application (see Figure 5.9).

To initiate the connection of device and wall applications, our *DeviceManager* component receives both the tracking data as well as pings from the mobile devices and matches these according to time and movement. As a result, a *GraphLensDevice* is created which receives a copy of the graph data on the display wall where each proxy element remains connected to its original to propagate changes, e.g., edge creation, when needed. These graph proxies are manipulated either by the *SelectionControl*, the *PhysicsManager*[3] for the sieve filter tool, or the *LensControl* which contains a

[3]The physics component was implemented in the student research project of Norman Lorenz and was then improved and iterated by the author Ulrike Kister.

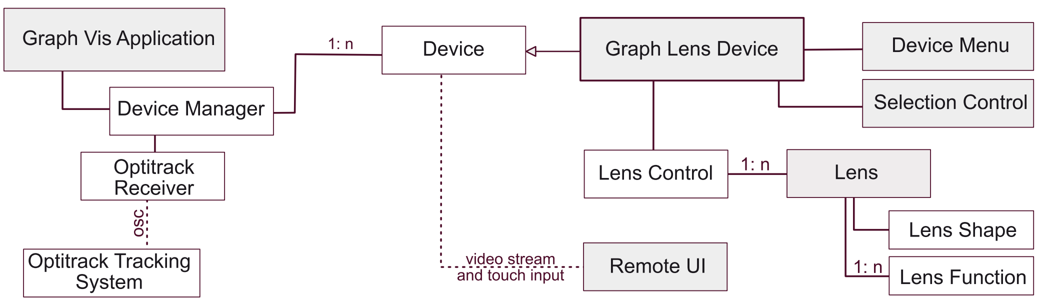


Fig. 5.9.: Simplified prototype structure: The main application of GRASP runs on the display wall presenting the context graph visualization. Devices connect to it via the remote UI registering themselves with the *device manager* which associates the appropriate tracking identifiers from OptiTrack. This creates a *graph lens device* which contains a *lens control* for graph view manipulation.

range of possible lenses which can be triggered through the *DeviceMenu* (see the MultiLens prototype in 4.3 for more details on the components of the *Lens Control* and its multi-function lenses). The device receives additional selections from the *SelectionControl* on the graph application at the display wall.

5.4 Observational Study and User Feedback

To gain a better understanding of the GraSp repertoire and the users' workflow as well as evaluate the cooperation of our techniques, we conducted a qualitative user study to get hands-on feedback with our system. We recruited 9 participants (three female, six male) between the age of 22 and 35 years (avg.: 26 yrs) through mailing lists including students and post-doctoral personnel with background in visualization, but not necessarily graphs, to explore the social network data set (see Appendix A.3 for details on the data set). While they were not part of our research group, the students had previously attended courses at our institute. They used the setup as described in the implementation section. All participants use mobile touch-enabled devices daily, but have little experience with larger wall-sized displays. Two worked with graph data frequently, while three used node-link diagrams only occasionally.

5.4.1 Study Design

The sessions took approximately 45 min per participant and were divided into two sections: In the first section (S1), the experimenter would describe a possible task and ask the participant to suggest how they would accomplish a solution. Afterwards, the experimenter explained the individual features including the possible alternatives within the system while the participant interacted. This process was repeated for every feature to make the participant familiar with the application. The second section (S2) addresses our main research question of how and if the techniques are used in sequence and combination and how they fit within a user's workflow. This section consisted of five larger exploration tasks that incorporated a sequence of two to three sub-tasks. Participants were asked to accomplish these tasks without help from the experimenter who stayed at a neutral position away from any interaction to not interfere with movement and behavior. Participants were encouraged to describe and comment on their every action. The sessions were video-taped and a second experimenter was present to record comments and actions along a semi-structured protocol. Finally, participants filled out a post-study questionnaire with seven questions relating to the assessment of our techniques and additional open questions.

Phase S2 incorporated tasks that were organized to form logical interaction steps. Tasks generally belonged to one of three task types: As an example of the first type, users had to investigate the neighbors of a specific node which were difficult to identify as they were part of different clusters and hence distant from each other. They had to compare their attribute values and elaborate on their details (representing tasks: *select, connect/reconfigure, abstract&elaborate*). For the second type, users had to focus on smaller groups, such as family connections, where in one case they had to identify the older generation and add edges from the son's new fiancée to the graph (*encode/connect, filter, select from group, manipulate*). Finally tasks of the third type were of a more explorative nature focusing on larger clusters of nodes. In one example, the users had to discuss and explore the age distribution and patterns within a specified group of friends (*overview, explore/connect, filter*).

5.4.2 Observations and Results

Two of the researchers separately went through logged data to summarize and categorize behavior. The videos were used for confirmation where protocols were insufficient. As a result of the protocol analysis, videos were examined specifically to code participants' movement (type and amount) and observe their focus switches between devices. From the protocols, the video data, and the questionnaires, we extracted interesting observations and gained valuable insights during the study, also including minor user interface improvements and possible alternative interactions. In the following, we discuss selected insights focusing specifically on items concerning 1) the participants' distribution of focus between the individual displays, 2) the participants' workflow and sequence of actions, as well as 3) the individual techniques.

Distribution of Focus

Our study started with the use of the *details on demand* technique that provides further information on the mobile when selecting a node on the wall. We observed that there was an initial phase where users had to comprehend the decoupled input and output possibilities of the device combination. However, it took only a short time for them to understand the mobile device's role as a personal visualization and interaction tool that extends the capabilities of the large context display. Beside the wall selection, during our studies we learned that all participants liked the spatial pointing as an alternative selection technique. We observed very different styles of user movements, distances to the wall, and levels of focus and awareness concerning the mobile and large display. Based on our interviews, observations and video recordings, we identify two groups (G1 and G2) which were equally represented in our study and which is also reflected in their choice of selection technique. Some participants strongly focused on the combination between the wall and mobile and thereby switched their gaze and physical position frequently. This group (G1, $n = 4$) used both interactive displays equivalently and thereby seemed to have a strong overall awareness of the entire content at any time. They brought the mobile selection, detail view, or tool view into line with the large context visualization as they switched their focus and used the display wall for selection as well. Within G1, one participant (P1) showed unusual behavior in that he picked up content from the wall-sized display using touch and then often turned away from the display wall, temporarily focusing completely on the mobile display, but all in all switching his attention frequently between devices. In contrast, several other participants (G2, $n = 5$) were very focused on the mobile device. They used the large context visualization mainly as an overview from which they picked regions of interest. For selection they preferred pointing techniques and worked with content on their mobile in a more exclusive way while only sometimes looking up at the display wall for orientation and overview. We see that these participants typically interacted far away (2–3 m from the wall) and used the perspective pointing and mobile interactions more prominently (e.g., pointing, freezing, encircle selection on the mobile) instead of actively using physical navigation in front of the wall.

Workflow and Sequence of Actions

Participants were very successful in solving the given tasks and found diverse solutions and workflows. They often started out by selecting a region of interest (ROI), followed by further refinement and application of a tool or lens. Specifically, participants from G1 frequently used touch on the wall for data collection (tap or lasso), while participants from G2 primarily used pointing and optionally froze the movement before using either the rectangle, tap, or lasso selection on the mobile device. Freezing was very important to participants and was used constantly after pointing, likely to relax the posture and focus on the selected ROI. However, not all participants clearly separated freeze from rectangle selection. Except for the detail view and sieve filter tool, all other techniques can be applied continuously without prior content selection. We observed that for at least two participants this separation seemed to be a challenge and might have caused minor issues in the deactivation of tools. For instance, it seemed that after unfreezing when moving along to other regions, they were surprised by the Bring Neighbor lens still being active.

Participants resided at very different distances to the wall-sized display (with some relation to their association to G1 or G2) and hence moved very differently during the study. At the extremes, a participant (P8) from G2 was consistently at a distance of approx. 3 m central to the display wall and rarely moved from that position, using perspective pointing for all her interactions with the wall-sized display. On the other hand, two participants (P1, P9) were very active using criss-cross movements within the complete space in front of the display wall. However, the majority of our participants ($n = 6$) naturally positioned themselves at their individual neutral distance (approx. 1–2 m from the display wall, P7 at 3 m) which seemed to be their personal comfort position. They all moved strongly parallel to the wall at this personal distance, with participants from G1, who used touch on the wall most, returning to this distance before moving sideways again. To access content at the top of the display, all but one participant (P4, height: 1.88 m) switched to perspective pointing, instead of lifting their arms using orthogonal pointing. To summarize, the tool seemed to be flexible and could be used by all participants independent from position and movement in front of the display wall. We observed that it was very helpful to combine the different tools and techniques for solving tasks. This was done frequently and naturally by all participants and was not even mentioned as a feature.

Observations Concerning the Lens Techniques

The tangible graph lenses proved to be very flexible tools that were frequently combined with other techniques for solving tasks. A typical interaction flow often consisted of pointing towards a region of interest, freezing the content, applying a lens function, and finally selecting identified content using a lasso to gain further details. Surprisingly, this also meant that lenses were rarely used as active tools to be moved continuously around the data set to gain insights, but were rather used on already explicitly chosen areas using the freeze operation. Even in explorative tasks, participants often worked with samples from selected clusters instead of moving about the entire node-link representation. However, this may change with experience or may also be a result of the specific set of tasks used in this study and will have to be investigated further in future work. Regarding a specific lens instance, offloading the adjacency matrix onto the mobile display to simultaneously show both visual representations physically decoupled was well-accepted by all participants. Participants used the adjacency matrix frequently to identify connections and patterns while referring back to the node-link representation on the wall-sized display specifically when editing the selected sub-graph. There was some disagreement on whether the unlock button for edit operations was necessary. While some explicitly found it to be an unnecessary interaction step (P6) as they were holding the mobile with both hands while reading the matrix, others (P5, P8) used their forefinger strongly to follow rows and columns and hence found an unlock operation essential to prevent unintended manipulations.

5.5 Discussion and Summary

We identified challenges and advantages of the combination of mobile devices and a large display. It is a general limitation of visual data analysis that it is only feasible for a limited number of nodes, i.e., for larger graphs an initial query or filtering step is necessary before reasonable use in a visual exploration setup. The large display and multi-device setup can help distribute this content and organize data at different levels of detail. However, as a certain limitation this also requires more cognitive effort for the user in managing content on the different devices. The study has reinforced that with multiple displays there is the need for additional focus and attention switches which may lead to additional interaction efforts and increased cognitive load, e.g., for tracking changes on the wall. Clear visual indications of focus regions and selected objects on both devices and consistent feedback can reduce these efforts. In addition, we have also seen several advantages of this device combination as each user worked with the mobile as their own tangible personal tool, and seamlessly accessed content even from afar.

Because of our design, all interactions were feasible and executable on the mobile display while using the wall-sized display mainly for overview tasks. We specifically focused on this setup to support the use in multi-user scenarios. Taking our techniques as the basis, further investigations could enrich this scenario by allowing more diverse views and changes on the display wall. Our techniques utilize the mobile as a personal toolbox for graphs and thereby allow independent parallel work on the same context visualization without disturbing other collaborating users. After initial registration of devices, all mobile interactions are possible, and touch selections on the wall are automatically send to the closest device. However, the current prototype is focused on parallel, individual work and does not prevent editing conflicts so that extension is required for actual application to multi-user and collaborative scenarios.

In our use of physical metaphors, e.g., for perspective pointing and the sieve tool, we saw evidence of participants enjoying the use of physics for visual data analysis. However, this needs to be differentiated for beginners and experts. While slow animations and additional interactions help to develop a clear understanding of the principles and steps needed, we found the lack of efficiency in the sieve tool too severe for repeated use in professional contexts. Consequently, shortcuts, ways of personal configuration, or automated adaption to the user's experience level should be included in a future iteration of the system.

As the number of participants in our user study was small, no significant quantitative statements can be made. However, with our qualitative measures and observations we aimed to present tendencies and rise questions for future research studies and application designs. Even within our small group of participants, we saw that there is not just one single style of interaction. This confirms related work (e.g., [TBJ15]) where diverse exploration strategies for selected tasks have been observed. While we identified two general groups of focus behavior, even within these groups there were variations in terms of workflow and sequences of action (see 5.4.2). The GRASP techniques were flexible enough to allow for these variations including the different patterns of movement for navigation within the data space. However, more adaptations to the individual usage should be considered and could further personalize the system (e.g., unlock button for manipulation). All in all, we see the advantage and strength of productive solutions in the versatility and composition of tools and techniques to fit individual strategies and user preferences.

This chapter presented the GRASP system as a set of interactive techniques that combine a wall-sized display and mobile devices tracked in space for graph visualization and interaction. We showed that with the GRASP techniques the mobile as a lens representation can become a useful toolbox for exploration per user. The techniques addresses a large range of diverse graph visualization tasks from basic selections and details on demand to alternative representations, manipulations, tangible graph lenses, and diverse filtering techniques. In particular, we investigated physical metaphors as well as spatially-aware and body-relative interactions for selecting and filtering multivariate data items. In a qualitative study, we found workflows to be very diverse and our system well suited to handle a wide range of interaction sequences and combined techniques.

Our techniques support individual local interaction on each user's personal mobile device, which serves both as an additional, tangible visualization view that can be manipulated as well as a pointing device to interact with and coupled to the display wall. Therefore, the contributed techniques are designed to allow interacting in close proximity to the display wall as well as remotely from afar. In the study, we have seen that this flexible movement suits the users' individual preferences and workflows. However, we have also confirmed that the often regarded attention switches between decoupled views are an issue (as known from other studies of MDEs, e.g., [RNQ12b]). While we believe the toolbox to have a range of advantages, these attention switches are well-known as a reason for reduced efficiency. Regarding the advantages of the mobile with its interaction capabilities remote from the wall-sized display as well as these concerns of attention switches, the next chapter will investigate the use of mobile devices as controllers for lenses on the display wall, eliminating attention switches.

6 Mobile Devices as Remote Lens Controllers

Contents

In the last chapter, GRASP provided a personal toolbox which turned out to be suitable for individual work and may be applicable in a collaborative scenario. While this personal toolbox character is of advantage to focus interaction on the lens output, it also requires attention switches between wall-sized display and mobile device which is a known reason for reduced efficiency [RNQ12b]. However, we saw clear user-preferences in where users position themselves individually in front of the wall while using GRASP: Participants stood at a user-specific comfortable distance of a few meters (cf. section 5.4).

To enable flexible position and control from any location in front of the wall-sized display and at the same time eliminate attention switches, this chapter proposes an eyes-free controller for configuration of lenses on the display wall. For this work, we integrated graph visualization and lenses in a more complex data application presenting multiple coordinated views. Using this application, the chapter presents the DI.VI.CO approach that supports interactions both close to the display using touch and remotely from afar using a mobile device. The DI.VI.CO design focused on using a simple interaction vocabulary with consistent mappings between both interaction styles thereby designing for an improved workflow (cf. challenge C4, page 72). The mobile device is only used for its input capabilities. This includes both aspects of spatial interaction as well as touch interaction on the device. This duality between interaction styles was created to investigate users preferences in movement and positioning in a qualitative study. The evaluation is conducted with pairs of users collaborating to gather data and thereby investigates the use of lenses in closely coupled collaboration (cf. challenge C5).

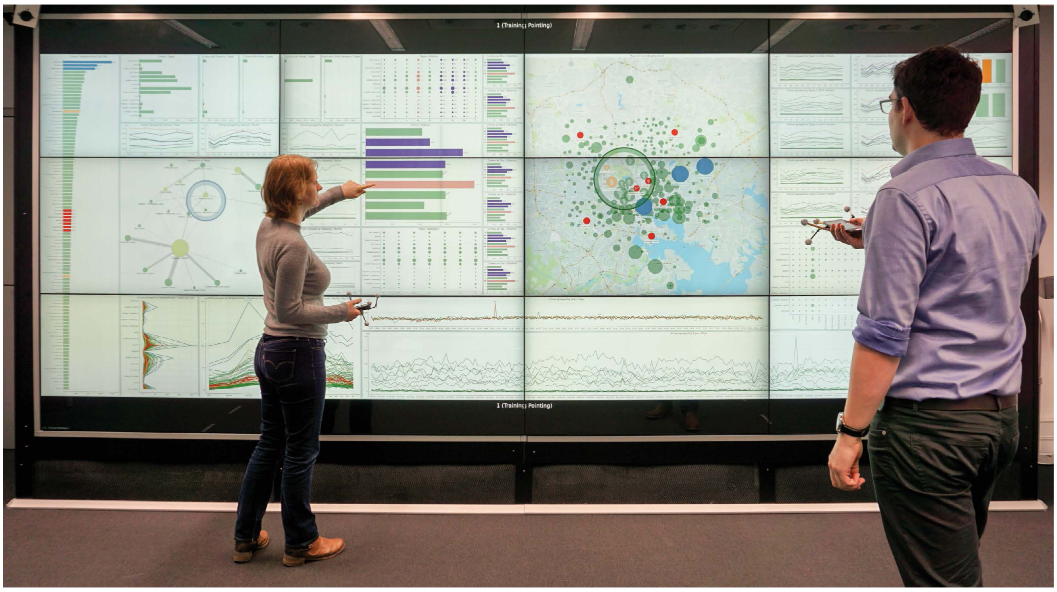


Fig. 6.1.: DI.VI.CO presents a system of multiple coordinated views at a wall-sized display. It supports data exploration through touch interaction on the display or by distant interaction using mobile devices.

6.1 Introduction

Multiple coordinated views (MCV) are an important solution to exploratory visualization [Rob07] where operations and interaction between multiple views are linked. MCV are of specific benefit for multivariate data sets since the diverse attributes of the data can be represented in separate views while maintaining their connection. The visualized views in MCV may include a variety of different visualization techniques, among them graph visualizations. To present these visualization views, the screen estate of large wall-sized displays is especially suited while at the same time supporting the exploration of data by multiple collaborating users (see Figure 6.1).

Furthermore, there clearly exists the need for movement both to get an overview of the data and view layout but also to identify and examine interesting views individually. Therefore, it is important to enable interaction from flexible positions in front of those views while at the same time supporting reachability of upper and lower areas of the display wall. As an alternative to touch interaction, Jakobsen et al. [Jak+15] compared mid-air gesture for manipulation from afar, but found it was too cumbersome and error-prone and was hence only used when moving around was entirely inconvenient. We argue that the control from afar can be increased by using smartphones as distant controller devices.

Previous works already introduced mobile devices as remote controllers for content on large displays for other application cases [SBR13; Zad+14] and specifically pointing towards the display [Bor+10; SBR13; Woź+14; Lan+16] (see also section 2.2.3). The following work builds on these works and applies them to data exploration and lens configuration. However, interaction should not be limited to pointing but integrated smoothly with the direct touch interaction on the wall-sized display. This results in challenges of consistency for interaction design between the two interaction styles. In the following, the DI.VI.CO approach is described in more detail. This application was designed to enable flexible positioning in front of the display to investigate behavior and positioning of collaborating pairs of users for data analysis tasks.

6.2 DI.VI.CO: Touch and Distant Interaction for Data Exploration

With our dual interaction visualization control approach "DI.VI.CO", we[1] propose a system that enables both direct touch as well as remote control for interaction with data visualization and lenses specifically. This is also where DI.VI.CO strongly differs from our GRASP principle. Instead of bringing parts of the view into the hand of the user, we aim to enable flexible movement by supporting lens interaction on the display wall from the distance (see Figure 6.2). The mobile device functions as an input device alone, with no additional output, and is used solely to interact with the content on the large display. At the same time, attention switches are eliminated as the mobile device is used eyes-free and not for output. As a result, DI.VI.CO uses the mobile device as a *lens controller* (cf. green in Table 5.1, page 99).

The following sections present the general principles of DI.VI.CO introducing the system design and selected interaction techniques regarding first basic interactions before focusing more specifically on lens interactions.

6.2.1 Basic Interactions for Multiple Coordinated Views at Wall-sized Displays

It was our goal to support interaction close to the display with touch as well as remote interaction in front of the display by using the position and orientation of a mobile device as a pointer towards the display and the additional touch capabilities of the device to trigger interactions. We aim to keep interactions as unrestrictive and easy to learn as possible and hence apply a very simple and basic interaction vocabulary of tap, hold, drag and double-tap with an additional swipe up or down gesture on the mobile and no complex multi-finger or bi-manual gestures. For

[1]'We' in this section relates to the author Ulrike Kister, as well as Ricardo Langner and Raimund Dachselt as co-contributors on this research.

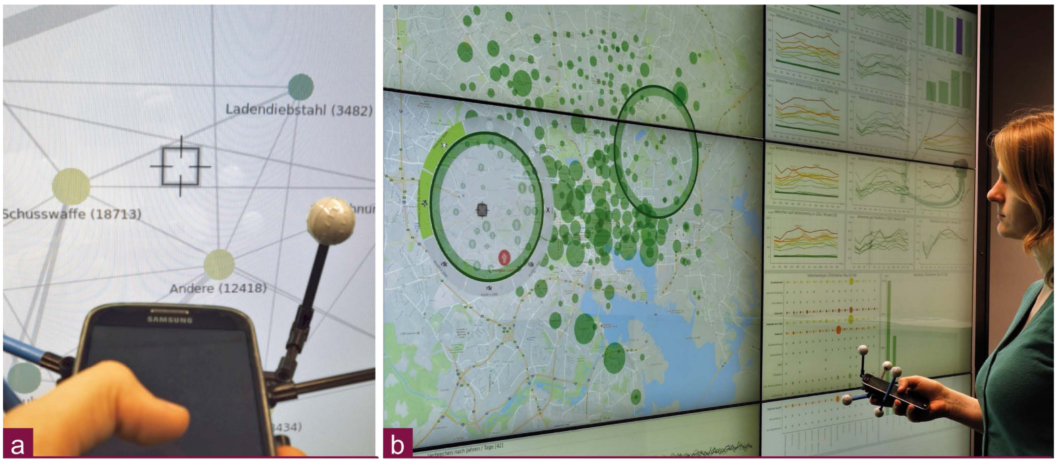


Fig. 6.2.: DI.VI.CO enables control of lenses with direct touch or remote interaction using mobile devices. The mobile functions as a pointer towards the display controlling a single cursor that can be used to 'touch' the device from afar.

enabling a rich repertoire of exploration features with these few simple interactions, we apply context-dependent interaction.

A major challenge for the combination of contact-based and remote interaction is finding a consistent interaction vocabulary to ease the transition between both types of interaction. As such, we aim to provide a joint mental model where pointing with the mobile device is only an extension of the user's arm and equal to touch, with only few additional extensions or shortcuts. By pointing the mobile device towards the display, a cursor is visible that visualizes this extension (see Figure 6.2a) and enables distant interaction. For interaction, a *tap* on the display wall surface invokes the same response as pointing with the device and tapping on the device surface (*point-tap*). The same is done for *double-tap* or *hold* interactions. However, double-tap or hold interactions on the mobile device might be problematic as the device pointer position moves while pointing due to the natural jitter of the hand [Mye+02]. This can be stabilized by considering the last positions of the device pointer when transferring the double-tap or hold to the wall. To support improved precision, we alternatively provide a *swipe up* gesture (towards the display wall) or *swipe down* gesture (away from display wall) on the mobile device to trigger interactions that would otherwise use a double-tap or hold. Finally, drag movements on the display wall can also be created through pointing by performing a *hold* on the device while moving it (*remote-drag*).

With this general interaction repertoire, DI.VI.CO supports a range of interaction features suitable for interaction with multiple coordinated views. The basic and most essential interaction with visual elements in the visualization views is **selection** which can be done by tapping individual elements or encircling them using drag (or *remote-drag* with the mobile device). In contrast to direct touch, the mobile device

Fig. 6.3.: Selections in any visualization view are propagated to other views that present the same data attribute (a). Exact values can be accessed using guides (b) or details on demand (c).

adds haptic feedback using vibration when hovering over individual items which aids in precision before tapping to select. Consistent with the principle of **linked brushing** [BC87; Koy+18], selection is always propagated to all other views that present the same data attribute. Therein, the color of selection and group selections are an important factor: We propose that taps in quick succession are considered part of the same selection group, i.e., receiving the same color, while lasso encircling is always considered a new selection group.

Depending on the provided visualization views within the MCV diverse alternative interactions can support exploration. For graph visualizations, **rearranging and moving graph nodes** can help identify connections. This can equally be done by either *dragging* a node using touch on the display wall or pointing with the mobile and using *remote-drag*. Furthermore, many visualizations allow providing **details on demand**. We suggest *double-tap* for activating and deactivating these details and add the possibility to toggle details either using pointing and *double-tap* on the mobile or *swipe up* to trigger details and *swipe down* to remove them from the wall-sized display.

The various views may further support individual **analysis tools** to ease exploration among them aid lines and magic lenses. As the focus of this thesis, interaction will be discussed in more detail in the following section. In DI.VI.CO, aid lines are guides that highlight data items which they cross including presenting labels with appropriate values (see Figure 6.3b) and are used to read and compare exact values. These guides can be created by a dragging gesture originating from the horizontal or vertical axes. For line charts both direction of guides are possible while bar charts logically only support guides perpendicular to their bars. An additional transparent width makes them easy to move through *drag* or *remote-drag* with the mobile device. They can be removed by moving them out of the view. To ease and speed up interaction, we also reserved a *swipe up* or *swipe down* on the background of each view to invoke a tool. For scatter plots, line or bar charts, aid lines are considered this default tool while graph and map-based visualizations are associated with magic lenses.

Beside these interactions, there could be further extensions regarding the *interactions with visualization views*, i.e., interactions that regard the management of entire views. As such, they include the manipulation of the grid that makes up the MCV, its layouting and the positioning of views, e.g., switch position between two views, collapse or maximize a view, or attach an additional view as an overlay for comparison tasks. Additionally, *cross-view interactions* other than linked brushing could enhance the MCV, e.g., filter tasks such as excluding data items selected in one view within specific other visualizations.

6.2.2 Distant Lens Controller

Beside the aforementioned guides for reading data labels and comparing values, lenses, as the focus of this thesis, are an additional analysis tool that we aim to support both on the display wall and by remote control through the device. Lens functions are data type and visualization specific: While some lens functions, e.g.,

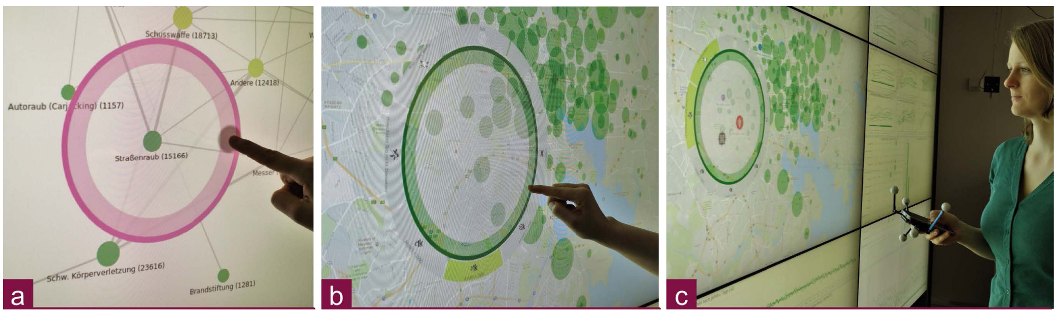


Fig. 6.4.: Within our DI.VI.CO prototype, lenses can be applied to the graph visualization (a) or the map-based visualization (b). Interaction is possible through touch on the display wall (a-b) or remotely using the mobile device (c).

attribute filter, can be applied on many visualizations removing the appropriate elements (or parts of the element), most lens functions are limited to a specific representation of data. For instance, Bring Neighbors or Adjacency Matrix functions are only suitable on visualizations with node-link relations. In DI.VI.CO's multiple coordinated views, lenses are hence restricted to individual visualization views.

To create a lens on the view, we apply the same principles as for details on demand and guide creation. A *hold* while touching on the background of graph or map-based views invokes a generic lens without active function (a default function per view could be set). With the mobile the same interaction extending the arm (*remote-drag*: holding while pointing without moving) can be done from a distance. To reduce the limitations of natural jitter while pointing, lenses can also be created and deleted from afar using *swipe up* or *swipe down*, respectively.

For moving the lens above the data, any drag-movement on the display wall can equally be performed from the mobile using *remote-drag*. However, the pointer has to be on the lens to do so which may be difficult from a distance when only the lens border is considered part of the lens. Differentiating interactions within the lens is important to enable touch through the lens to access interesting elements identified with the lens function (cf. MULTILENS, section 4.2). As a result DI.VI.CO lenses can be in one of two modes, the default mode, where the inner area is touch-through, or the *filled-mode*, where the inner area has a mostly-transparent overlay (see Figure 6.4b) that allows interaction with the lens and creates a larger target for pointing. *Tap* on the lens toggles these two modes.

To select and manipulate lens functions of the lens, a menu is placed around the lens. It is positioned outside the focus area to keep the region of interest in view similar to our previous MULTILENS approach. To reduce unnecessary occlusion at

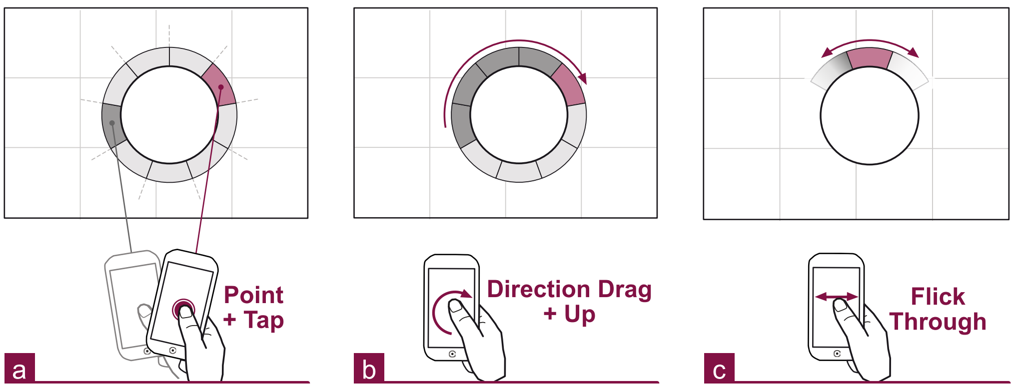


Fig. 6.5.: The lens menu can be controlled using the mobile device as an extension of the arm by point+tap (a). Alternatively, a drag motion on the device allows selection of menu items on lifting the finger (b). A visually minimized version supports flicking through the individual lens functions (c).

the transition of focus and context view, the menu can be toggled visible or hidden on *tap*. To remain consistent to touch interaction, *point-tap* (see Figure 6.5a) can be used on any lens menu button to select a lens function and toggle activation. The lens buttons' interaction area is extended towards the outside to ease pointing, however this still requires precision. In addition, a directional drag on the device can be used to highlight the individual menu items, and lifting the finger from the device toggles the selected lens function (see Figure 6.5b). This is especially suitable for frequent users that already know the position of lens functions within the menu and can hence quickly *drag* in the known direction similar to marking menus [KB94; KHA11]. Alternatively, we considered a visually minimized approach where the user *flicks* through the alternative lens functions (see Figure 6.5c). This supports exploring the lens functions for novice users and reduces occlusion of the context visualization but slows down interaction when lens functions are known and removes quick access possibilities to lens functions for expert users.

6.3 Technical Realization

DI.VI.CO uses the same technical setup as the previous GRASP prototype: The IMLD touch-enabled display wall functions as the foundation for the MCV visualization. The 3D tracking system OptiTrack [@Nat] is used to track the Samsung Galaxy S6 mobile devices which are used for distant interaction. Similar to GRASP, the *remote user interface* [Zad+14] is used to stream any touch interaction from the mobile to the main application on the wall-sized display.

We used crime data of the City of Baltimore publicly available through an open data project[2]. The data contains recorded victim-based crimes (more than 242k data items) that occurred over five years (2012 to 2016) with 15 attribute dimensions including date, time, location, crime type, weapon, district, neighborhood, and premise. The data is presented on the display wall in 47 visualization views including bar charts, line charts, scatter plots, a map view, and a graph visualization (see Figure 6.6).

Beside the existing graph representation from GRASP, various charts have been added and linked together with a *SelectionManager* that coordinates selection groups and highlighting colors between views. The device management is adapted from the GRASP prototype. It is extended by a *divico controller*[3] which interprets touch interactions streamed from the mobile display and its movement received from the

[2]BPD Part 1 Victim Based Crime Data by Baltimore Police Department, downloaded Aug 25, 2017 at https://data.baltimorecity.gov/

[3]The charts, layout, selection, and touch injection components as well as general integration of all modules were realized in part by Marc Satkowski in form of a student research project.

tracking system. After interpreting this input, it then injects the appropriate touch input on the wall-sized display. For example, a *hold* on the mobile display while moving the device will be interpreted and injected as a *drag* movement at the current position of the remote pointer.

The lenses and their configuration apply the GRASP lens components in their backend and were implemented for both map and graph visualization. For the map visualization, lens functions were added that highlight neighborhoods with specific attribute values, e.g., number of crimes higher than 3000, as well as lens functions that place additional textures on the neighborhood, e.g., an icon presenting the predominant weapon used for crimes within the neighborhood. The prototype uses the point and tap interaction in Figure 6.5a since it better integrates into the rest of our interaction concepts.

6.4 Qualitative Evaluation of Collaborative MCV Exploration

Using the presented prototype, we aimed to investigate collaborative user behavior and positioning while exploring data on large vertical displays. In particular, focus of this thesis specifically concerns the use of the lenses as well as aspects of flexible positioning and distance to the display. We conducted three pilot studies and seven actual study runs to investigate the users' behavior while collaboratively exploring the Baltimore crime data set in multiple coordinate views. Therefore, we recruited 20 participants (6 female, 14 male; one left handed) between the age of 20 and 27 years (avg.: 24 yrs) from the student population of our university including departments of computer science, psychology, and mechanical engineering. Six of those participants took part in the pilot runs and 14 were observed in the evaluation runs. None of the students had participated in the GRASP evaluation. All participants were studying for at least two years to assume a certain degree of experience. Their average body height (self-reported) was between 1.60 m and 1.98 m (avg. 1.78 m). Furthermore, they were recruited in pairs with the condition that team members had previously worked together.

6.4.1 Procedure and Tasks

Each session took approx. 95 min and began with a short introduction and general explanation of the Baltimore crime data set, its dimensions, and the basic layout of the views on the wall-sized display. This was the only time that a researcher was within the participants' interaction space. In all following phases, the experimenter

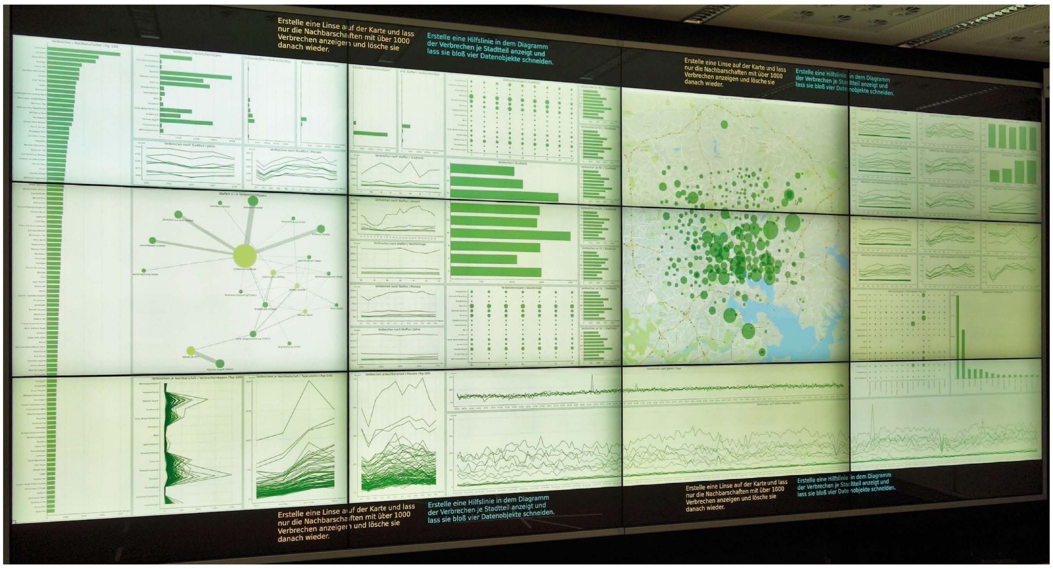


Fig. 6.6.: DI.VI.CO prototype and study setup with multiple coordinated view layout and training instructions (top and bottom).

and an additional person recording observations were sitting at tables at a distance from participants and display wall to avoid any influence on participants' movements. The following procedure was divided into three main interaction phases, starting with a training phase followed by two exploration phases.

The **training phase** (avg. 27 min) contained training for both touch interaction at the wall-sized display and distant interaction with the mobile device separately. The order of these modality trainings was counterbalanced among sessions. The initial training commands and descriptions were read by one of the experimenters. Participants were allowed to ask questions and individually iterate each interaction few times until they felt they had understood the action. Finally, each modality training ended with a set of commands where each participant was asked to perform four actions, e.g., "Select the neighborhood with the highest sum of crimes in Octobers.". These tasks were also used to validate that participants had understood the general layout of the views and could find the mentioned data items.

The **main exploration phase** (avg. 27 min) was designed to create an actual data exploration workflow. Participants were instructed to take the mobile device but could freely decide whether and when to directly touch the wall-sized display or use the mobile device for distant interaction. The phase included six question blocks with each three to five questions forming a logical workflow from initial facts to comparison. For example, 1. "How many crimes were committed with each of the given weapons?", 2. "Did the number of crimes with firearms increase over time?", 3. "Which crime types are committed using firearms?", and 4. "How do crimes of these crime types differ in terms of time of day they were committed?". The list

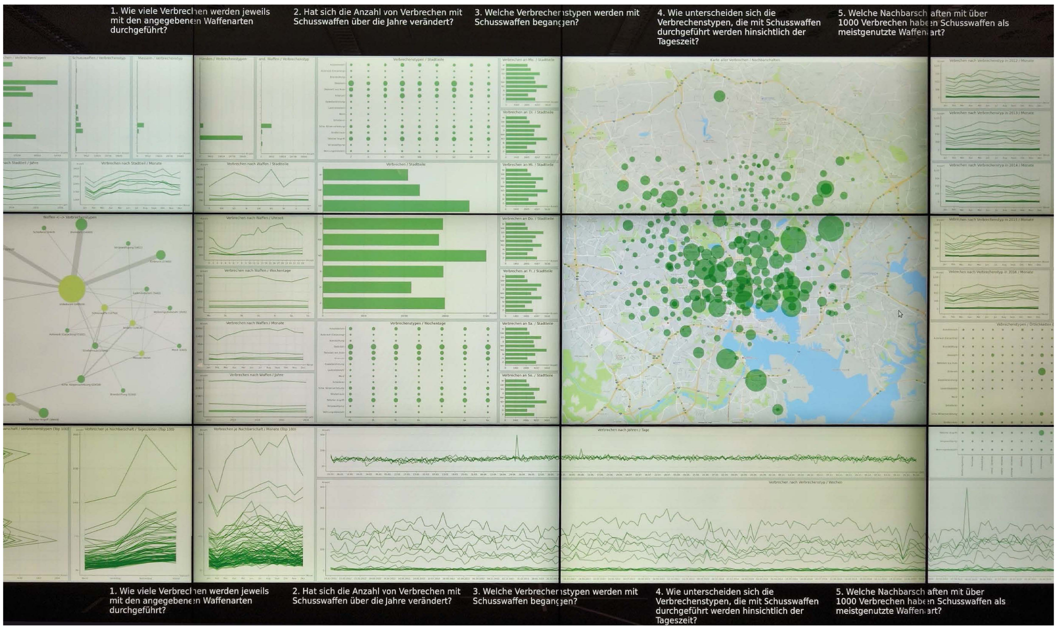


Fig. 6.7.: Instructions were presented at the top and bottom of the screen. They were grouped in logical blocks for the main exploration phase (cf. Figure 6.6 for color coded instructions in training phase).

of tasks can be found in Appendix A.4.1. Participants were instructed to clearly declare when they felt they could answer a question and pronounce the result. That is, answers were not registered according to specific selections or interactions, but could be discussed among the participants until they were sure to have found the result.

The second exploration phase, the **theses exploration phase** (avg. 10 min), consisted of four different hypotheses which were presented to the participants one by one. The users were encouraged to explore the presented data to find evidence that would give an indication if the hypothesis should be confirmed or rejected. Examples for presented theses are "A snow storm in January 2016 lead to a decrease of criminal activity" or "Neighborhoods with smaller number of crimes are relatively more affected by gun violence". To validate the theses, participants often had to regard multiple views and connect information overall, up to making assumptions to find relevant aspects and facts. All instructions within the study phases (training actions, question blocks, hypotheses) were presented both at the top and bottom of the wall-sized display (see Figure 6.7).

After an initial questionnaire concerning demographics and experience in the beginning of the study, participants filled out questionnaires after each training phase evaluating the interaction mapping, physical and mental demand and ease of use (Raw NASA TLX), and the perceived comfort of all possible interactions. A final questionnaire after the exploration phases allowed them to evaluate their preference between touch and pointing interactions, the general system evaluation (adapted from [Lew95]), as well as questions concerning their preferences regarding the use of the mobile device in general. The complete list of questions can be found in Appendix A.4.2. After the final questionnaire, a short debriefing and interview on additional comments was done with both participants together with specific focus on physical demand, additional feature wishes, and preferences.

Two of the **pilot studies** were conducted to test the appropriate mapping for the pointing cursor position. We first applied a distance-dependent pointing technique that supports both quick selection from a distance and more precise pointing when close to the display (adapted from [Lan+16]). However, this turned out to be counterproductive for users that were standing close to the wall and tried to reach upper parts of the display area. Since we did not want to force their movement, we decided against this advanced pointing technique and used perspective pointing with an additional 1 € filter [CRV12] for smoothing the output. The third pilot study was used to validate that all interactions were feasible using pointing only. The two participants were asked to use only the pointing in the exploration phases and it was shown that all interactions could be achieved within the average time frame.

6.4.2 Methodology, Data Collection, and Analysis Process

Protocols of the experiments were sorted and categorized according to four main categories regarding i) collaborative behavior or position, ii) workflow, iii) comments focusing on touch or pointing, and iv) individual system features (e.g., lens functions, aid lines, details on demand). We also collected participants' answers to the training and final evaluation questionnaires and added notes of individual participants during the debriefing of the users. Participants within a session were color coded as 'blue' and 'yellow' user according to the colored stripe on their mobile device (Px-c where x is the session-id and c represents colors B or Y for the specific user).

We video-recorded the sessions from three different camera angles, focused on participants, on their focus of the wall, as well as from the back. Additionally, logging was added to the application which documented the position of participants' joints (as defined by Kinect [@Mic13]), the devices' positions in space in front of the display wall, as well as its pointing position on the screen each at 12 Hz. Furthermore, we logged all touch interactions on wall-sized or mobile displays and any general application event. This data and the camera recordings were applied to an enhanced version of the group analysis toolkit *GIAnT* [ZD17] which was redesigned and extended with an additional set of views to evaluate our specific requirements for DI.VI.CO (see screenshot in Appendix A.4.3). It allowed us to simultaneously watch all three synchronized video streams and helped us identify and replay relevant scenes. Two of the experimenters separately coded the video data using GIAnT with focus on the exploration phases.

With the collected data, we also calculated a range of additional data including but not limited to the distances of each participant to the wall-sized display, the distances between participants (in 4 bins according to proxemics [Hal90]), the ratio of touches generated directly on the display wall in contrast to the number of interactions triggered through the mobile device, and the amount each participant walked during the study run.

6.4.3 Observations and Results

The following section describes our observations of users' behaviors and interactions during the study. Therefore, the results are categorized with focus on i) general movement and interaction behavior including observations of touch on the display wall and distant interactions, ii) collaboration aspects regarding the proximity of users and their collaborative behavior, and iii) lens interactions and workflows regarding this specific analysis tool.

General Movement and Interaction Behavior

On average, participants walked a distance of 436 m (SD=108 m) during the entire study session. Nevertheless, no participant complained about any fatigue or issues with standing or walking even when being asked explicitly during the debriefing. Even more so, physical demand was small M=3.36 in a scale of 10 (SD=1.44) while mental demand was at M=6.5 (SD=1.76) probably due to the number of views involved and the resulting complexity of the tasks.

During exploration phases, participants' average position was at 1.76m (SD=0.18m) from the display wall. Users spent only 0.3 % of the time (SD=0.6) in a distance of up to 0.46 cm of the display wall (cf. [JH14] where users spent 60 % of the time at this distance). Touch interaction was done at distances up to 0.8 m but users often only stepped closer to interact and then stepped back from the display wall afterwards (see Figure 6.8c). Table 6.1 presents the time spent at various distances

Distance to Display Wall	0.0 - 0.8 m	0.8 - 1.6 m	1.6 - 2.4 m	2.4 - 3.7 m
Mean	9.54 %	26.76 %	49.33 %	14.36 %
Standard Deviation	6.43 %	7.03 %	11.23 %	10.61 %

Tab. 6.1.: Time span that users spent at specific distances from the wall-sized display.

from the display wall. We defined these distance bins from our observations of where touch on the display occurred (below 0.8 m) and separated the remaining space accordingly. The table shows that participants spent most of their time few meters from the display wall. Independent of interaction style used, participants did not remain close to the display wall.

Touch on the display wall was the interaction used predominantly in terms of number of interactions for many of the participants (11/14). This preference also became clear in the answers in the final questionnaire regarding individual interaction tasks: Single selection (Wall-Touch: 10, Distant-Touch: 4) and guides (Wall-Touch: 11, Distant-Touch: 3) were clearly preferred using touch interaction on the wall-sized display, while selection of multiple elements (Wall-Touch: 8, Distant-Touch: 6) and lens interactions (Wall-Touch: 6, Distant-Touch: 7, no answer: 1) were balanced. Table 6.2 shows the percentage of interaction times conducted with either touching the display wall or interacting with the mobile device. Therein, we observed many continuous interactions with the distant interaction techniques using either encircling or lens drag movements which resulted in higher values for distant interaction.

The chosen interaction style also highly influenced the area covered by the participants through movement. For instance, participants in session 6 mainly used touch and hence were on average much closer to the display (see Figure 6.8a) while participants in session 4 predominantly used distant interaction and hence stepped much less towards the display wall (see Figure 6.8b). While interacting with the device, 5 participants used a single hand for interaction in a casual way, 6 used a second hand for stabilizing their input, and 3 frequently mixed between the two.

Interaction Modality	P0-B	P0-Y	P1-B	P1-Y	P2-B	P2-Y	P3-B	P3-Y	P4-B	P4-Y	P5-B	P5-Y	P6-B	P6-Y
Wall-Touch in %	58	99	68	68	69	57	81	28	10	49	25	53	90	79
Distant-Touch in %	42	1	32	32	31	43	19	72	90	51	75	47	10	21

Tab. 6.2.: Percentage of interaction time that each participant spent touching the display wall (Wall-Touch) or using the mobile device to interact (Distant-Touch). Values above 60 % and 80 % are highlighted to identify extremes.

Collaboration Aspects

Participants spent most of the time in closely coupled collaboration. As a result, we observed different coupling styles [Tan+06]: This involved mainly *same problem same area* or *same problem different area* (SPDA), as well as few situations of *view engaged* (VE) where one participant remained in a central position regarding the current interaction and possible changes in other views from afar. Team work was considered helpful for answering the questions (M=6.29 of 7, SD=0.96) and felt

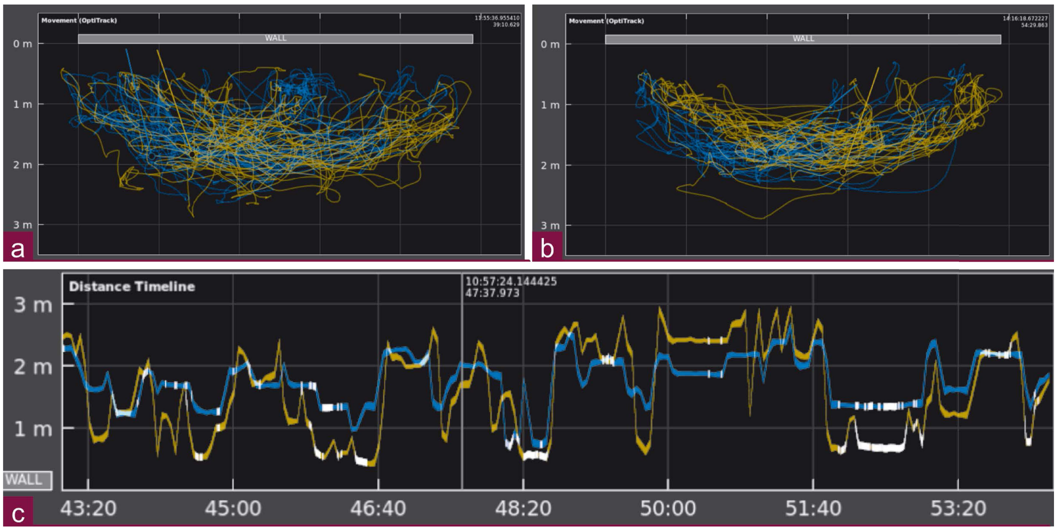


Fig. 6.8.: GIAnT: Movement at display wall for session 6 with predominantly touch interaction (a) and session 4 with predominantly distant interaction (b). Users only stepped close to the display wall to interact (c, interaction highlighted in white).

comfortable to team members (M=6.86, SD=0.35). Therefore, participants spent a lot of time in close proximity: Mean distance between users was on average 1.1 m (SD=0.14 m) which is considered the *far phase* of *personal distance* in proxemics [Hal90] (see Table 6.3 for times spent at specific proxemic distances). Team members also moved together in the majority of times. Times of separation were on average 26 s long (SD=16 s) but could be up to 4 m when participants were focusing on different views. These separations were sometimes triggered by i) searching for appropriate views to answer the question or more often ii) using a separate view for selection of aspects by one team member while the other identified different aspects in another chart. The later describes times that participants worked especially close together, despite the spatial distance. In contrast to the definition of SPDA (*same problem different areas*) by Tang et al. [Tan+06], we observed that conversation was very lively during these times. For instance, one member of the team selected crime types that occurred using guns in the graph visualization while the other compared the highlighted crime types regarding time of day in a line chart.

Users' Proximity	*intimate distance* 0.0 - 0.46 m	*personal distance – close* 0.46 - 0.76 m	*personal distance – far* 0.76 - 1.2 m	*social distance* 1.2 - 3.7 m
Mean	2.32 %	23.45 %	39.36 %	34.31 %
Standard Deviation	2.27 %	12.17 %	10.55 %	10.67 %

Tab. 6.3.: Time in percent that users in each session spent at specific distances from their team member (in distance bins according to proxemics [Hal90]).

Linking & brushing was an essential component used to answer the questions. It was used very frequently to find information and was commented on positively in the questionnaire's open questions section. Specifically, elements were deselected and selected to change their selection colors and compare them in subsequent tasks. However, since our DI.VI.CO system propagated changes to all other views, participants were also confused when things changed because the other team member selected elements, especially in the beginning (cf. 'visual footprint' [PBC17a]). On the other hand, this change in other views, even on the other side of the display wall, was also actively used to identify possibly relevant views during the process of solving a question.

Lens Interactions and Workflows

The lens interaction turned out to be very different from interactions with other features and tools. It was the one feature that was used more with distant interaction than touch on the wall-sized display (approx. 63 %), even by participants that would otherwise prefer direct touch. Participants (P1-Y, P2-B, P0-Y) also actively commented on their preference of pointing for lens movement, while creation and parametrization of the lens was frequently also done using touch. In the questionnaire, lens control was considered easiest in comparison to other features of the system and had very good results for both touch on the display and remote interaction. In contrast to other features, remote interaction was considered slightly easier than direct touch interaction for creation and positioning (see Table 6.4).

Due to their close collaboration, participants often created two equally configured lenses to explore the information space or created a single lens that both used for exploration. Even more, we repeatedly observed users collaborating actively with this one lens: One participant moved the lens using the mobile device while the other took responsibility for changing lens functions in the menu or touching through the lens to select the identified elements (P0-B and P0-Y, P4-B and P4-Y, P5-B and P5-Y). Participant P5-Y explicitly triggered this behavior after creating the lens himself telling P5-B: "Continue [moving the lens], I'll select them."

Ease of Use	lens creation	lens positioning	lens configuration
Wall-Touch	M=6.21, SD=1.01	M=6.29, SD=0.88	M=6.29, SD=1.10
Distant-Touch	M=6.38, SD=0.74	M=6.31, SD=0.72	M=6.15, SD=1.03

Tab. 6.4.: Participants' answers to the ease of use questions regarding lens interactions on a scale of 1 to 7.

6.5 Discussion and Summary

This chapter presented DI.VI.CO, an eyes-free remote controller for general interaction with visualizations in multiple coordinated views. Therein, we specifically focused on designing consistent interaction between touch on the wall-sized display and distant interaction in front of the display. The application supports lens positioning and manipulation both using touch directly on the lens or in its menu, as well as remotely from a distance and thereby enables control from any flexible position in front of the display. A qualitative evaluation was conducted with pairs of users to observe their behavior, positioning, and movement while exploring data. Where we found users standing relatively far away from the wall while at the same time stepping to the display and using touch predominantly though not complaining about any physical demand. However, the remote pointing technique was often used for lens positioning and larger continuous movements and showed to be suitable for lens interaction on both graph visualization and map-based data representations.

User Position and Movement The benefits of physical navigation (cf. section 2.2.2) were part of our motivation for the investigation into data exploration on wall-sized displays. As such, we base our work on the advantages of the user's movement in front of the display [BNB07; AN13] and the spatial orientation and memory of where interesting data items (or views) are placed. At the same time, it was important to us to enable the users' flexibility to position themselves where most appropriate, e.g., to remain in a group to discuss content instead of being forced to move around to accomplish a goal. Both GRASP and DI.VI.CO aim to ease this positioning by allowing both interaction at the display as well as remotely interacting (DI.VI.CO) or picking data from the display (GRASP) to explore it further.

The study by Jakobsen et al. [JH14] indicated users remaining close to the wall-sized display (< 76 cm) for more than 90 % of the time. This clearly is very task-dependent: Jakobsen et al. investigated tasks involving a lot of search and reading in news documents that could be freely arranged on the large display surface. For information visualization and multiple coordinated views in particular, we found very different behavior: Less than 10 % of the time were spent within 80 cm to the display and hardly ever as close as 46 cm. On the contrary, participants spent most of their time at a distance of around 2 m from the display wall, supposedly to keep multiple visualizations in focus instead of focusing on a single data item. As a result for future system designs, the supported tasks need to be examined to infer suitable distances and adapt both visualization and interaction accordingly. However, these aspects cannot easily be generalized. They could be influenced by the selected tasks as much as the layout of the visualization views as well as selected label size and dimensions of the views. Furthermore, though not explicitly mentioned by participants, the resolution of the IMLD display wall (12 FullHD displays at a size of 4.86 x 2.06 m) could be a reason for the users' distance. A higher resolution could trigger people to step closer to perceive more information.

Remote Interaction In their study on remote interaction using mid-air gestures for selection in comparison to touch on a display, Jakobsen et al. [Jak+15] indicated that users prefer direct touch interaction. Only if explicit movement cost was introduced, did users refrain from touching and used mid-air gestures instead. For DI.VI.CO we saw similar behavior for many of our participants when selecting content. Moving back and forth did not seem to be an issue, very little physical demand and no issues with fatigue could be observed. At the same time, we found specific tools to be very actively used with distant interaction: Lens positioning and multi-selection, where larger movements are required, were often accomplished using the remote interaction using mobile devices. Additionally, we saw very lazy behavior in terms of movement for some participants in the GRASP evaluation as all data could be extracted from the display wall and brought to the hands of the user due to the additional tablet device surface supporting additional interaction. As such, for future application of this work the possibility for remote interaction should be investigated focusing specifically on aspects of purposes and tasks that users want to accomplish from afar.

Mobile as Lens & Mobile as Controller GRASP and DI.VI.CO investigate the use of the mobile in very different roles: While GRASP is a separate toolbox presenting an input and output view, DI.VI.CO emphasizes the character of the controller with only input and very little output capabilities. Both approaches have clear advantages for slightly different interaction and collaboration styles in the same scenarios. DI.VI.CO remains close to the context display and actively supported collaboration between users, while GRASP collects interesting data from the context and supports individual exploration with the content. Our work has shown that separation has the advantage of very focused exploration of a specific subset of the data but it requires attention switches which may result in loss of efficiency and additional effort where the connection to context data is required. This raises the question of whether and how these two approaches can be combined? As well as, when is switching from remote controller (DI.VI.CO) to the actual representation (GRASP) worthwhile? The controller can be a valuable tool in identifying and exploring interesting parts of the context data and zooming into relevant aspects before focusing more closely on selected sets of data with the separate toolbox of lenses. As such, both the GRASP and DI.VI.CO approach are valuable in supporting the exploration of data. However, future work needs to focus on the combination of these two in one system. This requires discussions regarding issues of when to switch and how to ease the transition between using the mobile as controller to mobile as the lens and toolbox.

Body-controlled Magic Lenses

7

Contents

The previous chapters explored the advantages of making the lens graspable by moving the lens itself or the lens controller in the space in front of the wall-sized display. However, already in our DI.VI.CO approach, the device's output capabilities were unused and only its tangible characteristics were applied to manipulate the lens. This chapter explores further embodying the magic lens for exploration by eliminating this additional device. Hence, it investigates the possibilities of controlling the magic lens with the body. By creating interactive body-enhanced magic lenses, the knowledge of body-centric interactions [Sho+10; Jak+13], proxemic interactions [Gre+11] and the advantages of magic lenses (as described previously in chapter 3 and in related work [Bie+93; Tom+17]) are combined proposing BodyLenses, flexible work territories with various functions and tools. BodyLenses are body-controlled magic lenses that appear when users move in front of a display wall and which follow the users' movements, adapting their properties to the user's motion. They can additionally be adjusted through other interaction modalities, e.g., using touch on the display when in close proximity.

This chapter presents an exploration into using the body as a controller for magic lens manipulations, looking at the design space of what is possible with embodied magic lenses as well as the fundamental aspects and dimension that need to be considered when designing such lenses (7.3). These aspects include the appearance of such a lens, the mapping of the user's body to the lens, general interactions to manipulate it, and aspects of multi-user usage. Hence, this incorporates discussions of design alternatives, solutions for lens positioning, dynamic shape modifications, distance-based parameter mappings, and the idea of BodyLens tool belts. BodyLenses can be applied to a range of application cases (7.5). While the focus in this chapter will again be the exploration of information visualization in regard to graph exploration, it will also discuss the advantages of the concept as dynamic personal territories independent from an actual lens function, e.g., for awareness between users and user-specific workspaces and tools.

Research presented in this chapter has been extended and restructured from the work previously published in the following paper:

Ulrike Kister, Patrick Reipschläger, Fabrice Matulic, and Raimund Dachselt. 2015. BodyLenses: Embodied Magic Lenses and Personal Territories for Wall Displays. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces* (ITS '15). ACM, New York, NY, USA, pages 117-126.

7.1 Motivation and Related Work

Large displays are increasingly available and support both the visualization of large information spaces as well as multi-user scenarios. As known from literature (see chapter 2) and presented also specifically for data exploration tasks in the last chapters, it is our natural behavior to move around to see both details at close proximity or get an overview of the presented information space from afar. This entails that this natural movement has inherent meaning that can be analyzed to understand the user's intentions. As a consequence, the visualization may react to the users movement to support these intentions and thereby the associated goals and tasks. When tracking the user's movements, the interpretation of these **implicit** movements of the user may be joined by the use of **explicit** gestures and movement for function invocation. Researchers have analyzed mid-air gestures, body movement, and body-controlled manipulations [Jak+13; Sho+10] to enable and improve these interactions and investigate the differences as well as ways of allowing explicit triggers to distinguish implicit from explicit interactions. In the simplest case, the body can be used to control media on large displays, e.g., the position in space changes the playing speed of a video stream (as in industry marketing projects [@BF11]). While this is not necessarily a natural behavior to the user's movement, it allows users to apply their spatial memory to recall interesting positions in the video stream by stepping back to where they were when they saw them.

The body itself as an input controller has been an increasingly relevant subject in research, developed in conjunction with the ideas of ubiquitous computing and embodied virtuality [Wei91] as well as proxemic interaction [BMG10; Gre+11]. The general notion of *embodied interaction* as described by Dourish [Dou04] focuses on systems exploiting the fact that the user thinks and acts in a physical world. Within that approach also lies the idea of using the body's implicit movements as well as explicit body gestures to interact with a system. The implicit use of the user's body is supported by applying social behavior and psychological concepts to design new interaction principles that are less explicit and more integrated into our natural behavior. An example of this idea can be found in applying the principle of proxemics [Hal90] to develop proxemic interactions [BMG10; Gre+11]. An important aspects for the effective use of embodiment and body interaction are our body awareness, environment awareness, and social awareness and the skills that we as humans have in regard to these aspects which can be applied to human computer interaction [Jac+08].

For multi-user and collaboration scenarios, global interactions on the wall-sized display may lead to confusion of other users, e.g., as we have also seen for DI.VI.CO's

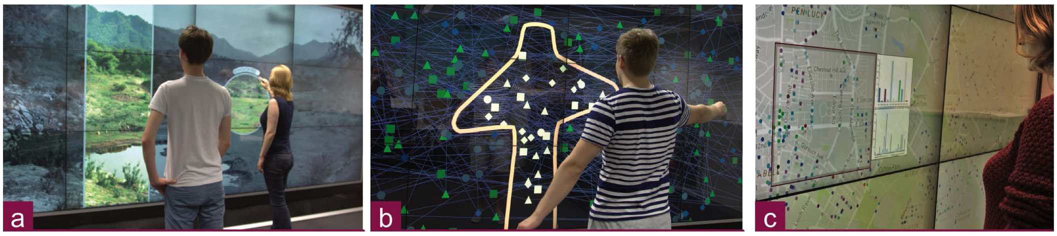


Fig. 7.1.: BodyLenses are body-controlled magic lenses that support parallel work in multi-user scenarios (a). They can be of various shapes, e.g., dynamic body-centric shapes modified by arm gestures (b), and provide additional tools and visualization views for data exploration (c).

global linking and brushing. Research has shown people's use of territories [SCI04] to separate the space in front of a wall-sized display [Aza+12] which however change with the users' movements [JH14]. To support this behavior and ease the movement of territories, a new version of flexible, dynamic territories should be considered. Using the advantages of the transient, spatially-restricted characteristics of magic lenses to manipulate the view in a local, user-defined – even user-specific – region lends itself to support this concept of dynamic working territories. Hence, we[1] investigate the use of body-controlled magic lenses to create an enhanced version of magic lens control but also a completely new concept of working with personal territories on wall-sized displays.

7.2 Definition and Characteristics of BodyLenses

This section will cover the basic principles of BodyLenses as a novel principle of body-controlled magic lenses and their resulting application as personal territories for in-parallel collaboration on wall-sized displays (see Figure 7.1). This will shortly define and characterize BodyLenses before placing the concept in relation to existing research principles and interaction concepts as well as discussing design space dimensions in the next section.

We define the term BodyLens as any type of personal, body-controlled magic lens used on a vertical display. In accordance with its definition, the magic lens applies a range of possible functions to a user-defined region of interest (cf. section 3.1) while '*body-controlled*' refers to i) any type of body movement with regard to the fixed display as well as ii) gestures of any body part – most prominently arms and hands, but also head, legs and feet – and any combinations thereof. Furthermore, provided the display is an interactive surface, iii) direct interaction *on* the display may also manipulate the lens, e.g., by means of multi-touch, pen, or tangible input.

[1] 'We' in this chapter relates to the author Ulrike Kister, as well as Fabrice Matulic, Patrick Reipschläger, Anke Lehmann, and Raimund Dachselt as co-contributors on this research.

While horizontal movements along the wall display typically influence the position of the lens, distal movements, body gestures, and direct interactions may influence any other parameter associated with the lens.

We argue that the combination of the human body and a magic lens is more than a simple mapping of novel means of interaction to well-known functions. A BodyLens is indeed not just a filter or magic lens, but also a personal tool and work territory. BodyLenses support effortless, *implicit* navigation within an information space as they automatically move with the user. They can also be adjusted *explicitly* using either direct contact-based manipulation, e.g., touch on the display, or mid-air gestures. For multi-user contexts, the lens function alters a local view into the data space, not disturbing other users. In regard to this aspect, the lens presents a personal territory that supports mutual awareness and fosters collaboration. Through personalization, BodyLenses also support individual tools along the lens, private annotations, and ownership of associated elements.

7.3 BodyLenses Design Space Dimensions

This section elaborates on the design space of BodyLenses. Both magic lenses and body-centric interaction have been extensively studied in the literature (cf. chapter 2 and 3). The design space for BodyLenses extends that prior work and establishes a general frame of reference for body-centric interactive lenses in a variety of contexts. The exploration of this design space is based on literature review, also including aspects of territoriality that we extend to vertical display spaces, and our own exploration and investigations.

The exploration of this space can be divided into four main dimensions: *appearance & shape, space mapping from user to lens, lens function*, and *multi-user (& multi-lens) aspects*. From these categories, the lens function is a very general aspect relevant for all magic lenses and will hence not be discussed again (see section 3.3 for categorization, example functions, etc.). The following sections will elaborate on all other aspects categorized as shown in Figure 7.2. For each of these aspects and dimensions, possible manifestations will be described and discussed. Where possible, we illustrate the design alternatives with related work or examples from our own investigations and implementations. We implemented a number of application prototypes that serve as examples for the feasibility of our concepts and demonstrate the potential of BodyLenses. Details on the technical aspects of the prototypes and the designed applications will follow in 7.4 and 7.5, respectively.

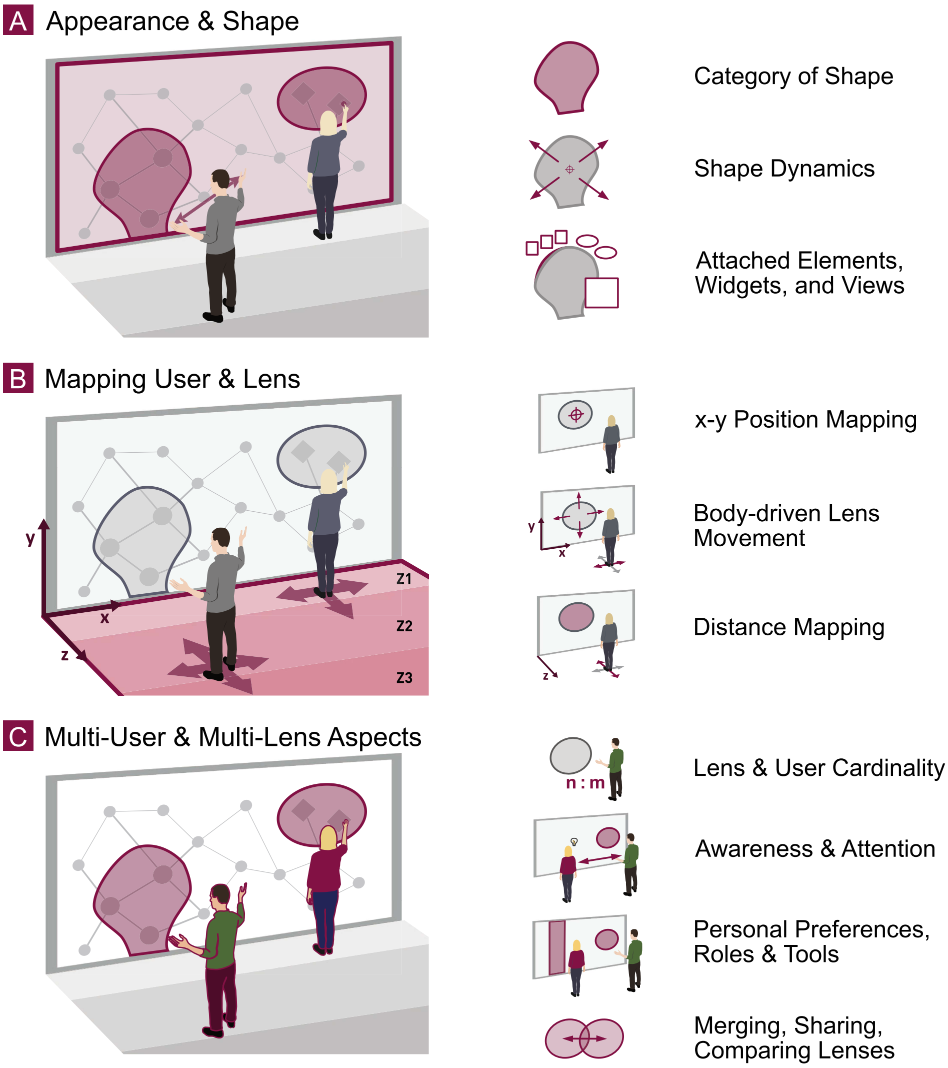


Fig. 7.2.: The BodyLenses design space elaborates on appearance and shape of the lens, the mapping of user position to lens position, and aspects of multi-user scenarios concerning multiple lenses.

Interaction As one of the most essential components of BodyLenses, we will first focus on general interaction aspects. Interaction will however also be discussed in all following design space dimensions focusing on its use to manipulate the BodyLenses and their properties, as well as the associated challenges and limitations. The BodyLenses design space offers a range of interaction possibilities to explore and manipulate the lens and the information space using a variety of modalities: While stepping closer to the display, basic principles of physical navigation and proxemics apply and are registered by the system. Small manipulations can already be performed from a distance using mid-air gestures and basic body movements. When stepping even closer, where contact with the wall-sized display is possible, more precise manipulations can be made using touch. In principle, we can hence differentiate between fine-grained, finger-based interaction *on* the screen (such as multi-touch or pen input) and coarse input through whole-body movements and arm or hand gestures *in front* of the display, with possible transitions between both types of input. More importantly, it is designed to support natural and fluent transitions between these modalities and interaction styles. Finally, we refrain from discussing interactions with the underlying data as this is task-specific. The design space of the former is however obviously influenced by the gestural vocabulary of the latter, as the two have to form a coherent and conflict-free whole in the final application.

7.3.1 A: Appearance and Shape

The appearance of a BodyLens refers to its characteristic visual features: Important aspects of this design dimension lie in the shape category (A1) and the shape's closeness to the dynamic body-behavior (A2). Furthermore, attachments to the visual lens components (A3) are part of this discussion.

A1 – Category of Shape

As discussed for any magic lens, the shape of the lens is one of the essential geometric properties that define the region of interest (cf. section 3.2). Generally, magic lenses often have the shape of geometric primitives such as circles or rectangles. Those types of shapes are of course also applicable to BodyLenses, where the contours can be dynamically modified using body gestures. However, the connection to the body further extends the shapes and forms to consider. BodyLenses can be regarded as kind of augmented shadows, with the lens contour more or less matching that of a virtual shadow cast by the body onto the display (e.g., [STB07]). Figure 7.3 proposes a number of typical BodyLens shapes that designers may choose to use in their applications, ranging from (a) conventional formats such as ellipses and rectangles to (b) more body-centric silhouettes such as convex shadows, mummies, stripes etc., while also including (c) data-driven, content-aware shapes. The appropriate shape

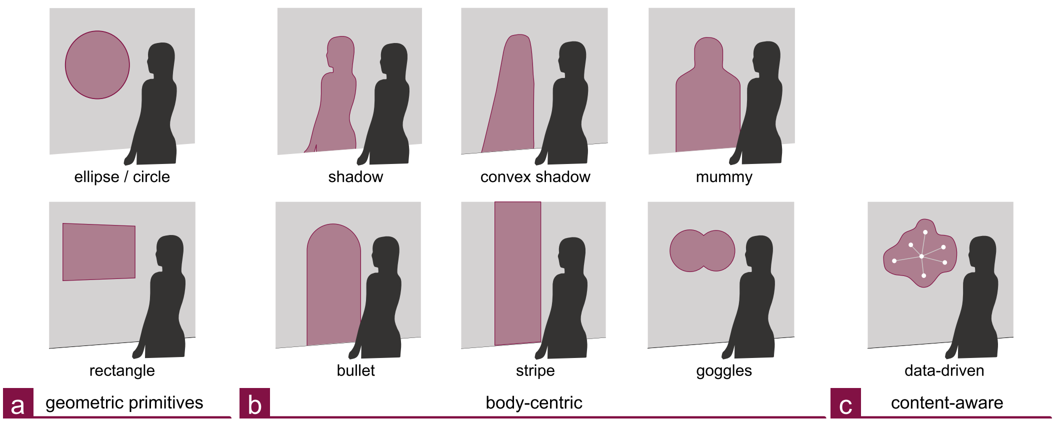


Fig. 7.3.: Selected examples of the three main shape categories of BodyLenses: geometric primitives (a), body-centric (b), and content-aware shapes (c).

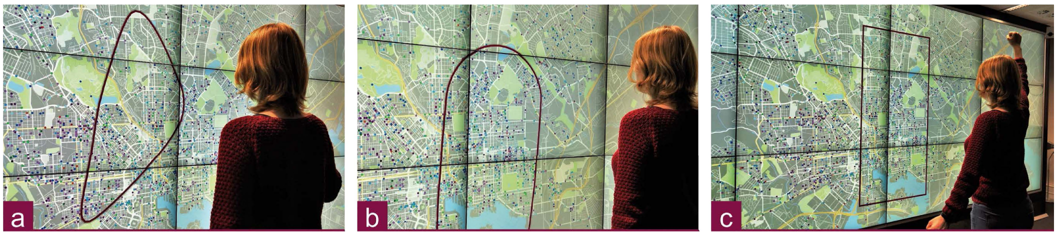


Fig. 7.4.: BODYLENSES can be used with dynamic shapes that implicitly change with the user's movement (a) including shapes that adapt only to the basic torso movement (b) or can be manipulated explicitly through interaction (c).

likely depends strongly on the application context. "True shadow" BODYLENSES, i.e., lenses whose outline exactly mirrors the contour of the user's body, presumably find more relevance in artistic installations or games [@Bro13; @CL13]. For co-located computing tasks, however, in particular information visualization applications, the projected shape of the human body may not constitute an adequate view space for magnified or filtered data. Thus, for those contexts, body-based lens shapes are likely to take more organic or geometric forms, e.g., vertical stripes or bands, whose width roughly corresponds to that of the body, to vertically elongated ellipses or other oblong shapes, or adapt to the form of the data [Pin+12].

Similar to traditional magic lenses, various rendering effects can be applied to the inner area as well as to the borders of lenses (cf. section 3.2). Borders, for instance, can be fully rendered or they can be implicit, i.e., the lens area is determined and visualized only by the filtered content within or outside its bounds. Similarly, the area within the lens can exhibit various degrees of opacity and filling patterns. In regard to BODYLENSES, this may very well be even more important than for general magic lenses as the lens may incorporate, for example, a translucent image of the user, depending on whether the lens should strictly display processed content or also fulfill other purposes (such as providing visual feedback for awareness).

A2 – Body-driven Shape Dynamics

Some of the presented shapes, such as the shadows or convex hull, continuously change by nature as they follow the user's movement. These shapes are adjusted **implicitly** by reacting according to the user's current pose. However, these constant movements may also distract users from the exploration of the information space. In our implementations, we strongly reduced those frequent lens shape movements by not considering the very active body parts, e.g., hands or fingers, but the more body-centric movements (the torso). This further supports the active movement of hands for touch interaction. Additionally, users can also **explicitly** change and adapt the lens shape to include or exclude data. For organic shapes, the arms can be used to punch bulges in the lens contour to stretch it in desired directions, similar to sculpting gestures used in BodyAvatar to grow virtual limbs [Zha+13]. If, for example, the user raises a hand to access some data, the shape may bulge outward at that location, mimicking the user's movement (see Figure 7.4a). One of the interactions that we propose for classic geometries and stripe-shaped lenses is to allow their width or height to be increased by stretching the arms sideways or diagonally, thereby expanding the area covered by their function To ensure that the lens size does not constantly change with any arm movement, an explicit hand pose can be required for shape-modifying interactions. For this, a grabbing gesture can be used, i.e., raising the arms and closing the hands in mid-air (see Figure 7.4c). To provide feedback, the shape becomes highlighted and can then be modified as the shape's border is "attached" to the user's fist. Embodied shapes can similarly be altered by bulging the contour when grabbing their border (as a handle) and moving the hands. Clutching is possible when the user re-opens, moves and re-forms the fist to resize the shape again.

A3 – Elements, Widgets, and Views Attached to the Lens

The BodyLens is not only a tool in itself, it can also provide additional tools depending on the current task and context. This requires additional elements and widgets to be connected to the lens for quick access. The selection of required tools is dependent on the role, current task, or general preference of the user. As a result the body-controlled lens can become a **personal tool belt** for its user. In accordance with the BodyLens concept, connected tool belts will move implicitly with the user. However, there are different ways to arrange and access these menus and tools.

In related work, individual tools were placed as semi-transparent overlays (as in the initial Toolglass interface [Bie+93] and body-centric tools [Elh+15; Sho+10]). As we apply this concept to magic lenses, this principle needs to be reconsidered as the content within, i.e., the focus area, is very important. Hence, in the BodyLenses concept, we consider placing menus and tools at the border along its edges to be much more beneficial as it enables the content of the lens to be entirely visible (see Figure 7.5), similar to the multi-touch magic lens solution presented in chapter 4. However, even then occlusion of data by these additional widgets is a concern to carefully access depending on context and application as the relation between focus and context area is a major advantage of the magic lens concept. One solution is to allow toggling of the visibility of additional menus and widgets. Depending on the interaction possibilities (touch, mid-air gestures, gaze, etc.), it might even be convenient to reduce occlusion by automatically fading out menus that cannot be accessed due to the position of the user (e.g., not in arm's reach for touch).

Tools and menus may exist of varying size: While menu buttons that simply manipulate the function of the lens may only be sized appropriate for interaction, e.g.,

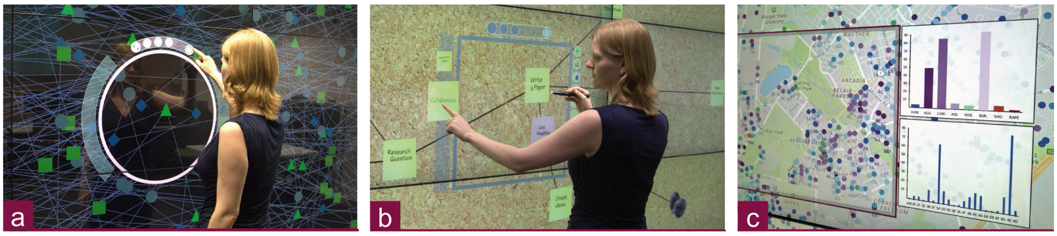


Fig. 7.5.: Paths for tool placement require definition according to the selected lens shape. Tools are placed along the lens border to not occlude the results of the lens function (a) and can incorporate menus and clipboards (b) or visualizations of aggregated data (c).

smaller button sizes for touch or larger areas for gesture-based interactions, other tools might depend on their size to provide additional information or storage areas. In our application use cases, we see the advantages of using clipboard areas along the lens to temporarily place elements to remain with the user while moving (see Figure 7.5b). Furthermore, tools may incorporate additional views for presenting details of multivariate data objects or aggregated data visualizations for the elements within the lens selection (see Figure 7.5c). This may especially be the case for lens functions that produce *separate views* (effect extent, cf. 3.3) that should not overlay the existing selection. While these separate lens function views relate to the focus area within the lens, views can also relate to the context visualization providing additional information for orientation and awareness. Off-view visualizations (similar to off-screen solutions [Gus+08; FD13; GST13]) can support navigation recommending and guiding the user where to go next within the visualization [Sha+17]. These additional views and elements may not only provide information on the content within the visualization but also information on the various users and their current interactions, e.g., rings of different colors around the lens dependent on the current state, and thereby support awareness between users.

Since the shape of a BodyLens can be very dynamic, the placement of elements like menus, clipboards, and additional views is not easy. We therefore suggest that for each shape certain parts of the border need to be defined as paths where the tools may be placed. This may include primary and secondary paths for different types of tools or sub-menus. In our prototypes, the circular lens defines a circular path for these additional items, while the rectangular lens defines straight paths (see Figure 7.5). For dynamic shapes, these paths need to either be updated accordingly or placed independent from the lens border. In any case, the design has to take into account the probable position of the user's arms. When positioning the menu at the top, the user might occlude the data within the lens when switching functions. Further research is required concerning this placement along dynamic body-centric lenses, since early experiments within our prototypes have revealed this to be a challenging topic.

Fig. 7.6.: While the default lens position is at the user's eye level (a), exploring the entire information space may require vertical movement, e.g., using the hand (b), or might be implicitly included through the selected shape, e.g., stripe (c).

7.3.2 B: Mapping User and Lens

A main advantage of BODYLENSES is their flexible, implicit positioning. In the following, we present the mapping of the user's movement in the three-dimensional space in front of the display to the lens' position in two-dimensional display space (B1) and its dynamic adjustment according to the user's body movement (B2). Furthermore, we discuss the distance to the display wall as a prominent dimension of body movement and its possible meaning for manipulation of the visualization and lens (B3). Considering proxemics [Hal90] and proxemic interaction [BMG10; Gre+11], the user's movement and distance-based BODYLENS interactions can be designed to mirror interpersonal engagement, based on familiar spatial relationships governing social interactions. As such, this section inherently includes the discussion of proxemic dimensions [Gre+11], such as *position, distance, orientation*, and *movement* (Note that *identity* will be addressed in the following multi-user section).

B1 – Position of User and Lens

The choice of the position of the lens on the display with respect to the user's body, i.e., the anchor point of the lens, is an important design aspect. Perhaps the most obvious materialization of a BODYLENS is a circle or a rectangle roughly centered at eye-level of the user (see Figure 7.6a). However, the lens can be attached to any trackable part of the body. The shape of the lens also influences this position mapping as it can be described by one or multiple control points. For instance, a circular lens might be positioned by attaching the user's head coordinates to the center of the circle. Body-shaped lenses can have multiple control points that are attached to multiple joints of the user such as shoulders, hands, and knees. A limitation of this mapping concerns the height of the lens: Other than the extremities which can actively be steered, body parts generally remain at the same height coordinate. As a result, the user's physical movement in front of the screen only moves the lens horizontally. This may result in problems of reachability of certain areas of the wall and especially concerns the upper areas, which might not be reachable because of the user's size, and the lower areas, as kneeling down might be uncomfortable or inappropriate. For example, the top of our wall display is at roughly 2.3 m, which can be unreachable even for people of average height. This problem can be avoided at the cost of additional space monopolized by the lens, if the latter is made to span the entire height of the display, e.g., using vertical stripes (see Figure 7.6c). Such kind of lenses might be particularly appropriate in multi-user scenarios, where each person working on the display occupies a dedicated personal territory delimited by such a band. Otherwise, additional means have to be devised to also allow the lens to move vertically and to enable 2D positioning in the whole screen space. In these cases, separate interaction is required to change the height of the lens.

B2 – Body-driven Lens Movement

The lens movement is closely coupled to the central body motion of the user (i.e., the torso) on all axes. When users are within the "touch-interaction zone", we believe it most suitable to freeze the lenses so that their position is no longer coupled with their movement. This enables working on details of the data and avoids jitter. Users may still drag the lens using touch while it is frozen.

Depending on the shape of the lens and the application requirements, a simple unidimensional mapping along the display may suffice, especially for basic entertainment purposes [@BF11; @CL13]. For data analysis with more densely populated information spaces, the lens however often needs to be steerable over the entire 2D coordinate space of the screen, as a result movement along the vertical axis also needs to be supported. We identify and implemented five alternative mappings for this axis that we found promising in addressing this problem: It is possible to 1) use the user's hand motion to move the lens up and down (see Figure 7.6b). The user can also 2) resize the lens to reach the upper or lower part of the display. Alternatively, it is possible to 3) set an offset by dragging the lens using touch, which is maintained upon moving in front of the display. Further, the height can be influenced by 4) the user's distance from the wall. Finally, 5) the (approximated) gaze direction can be used to manipulate the lens' height. These gaze or head-based manipulations for steering content or lenses have previously been investigated [ADS05; Leh+11], but are very dependent on the availability of additional tracking hardware. For all of these cases, depending on context and tasks it is likely necessary to incorporate a mode switch to trigger the explicit interaction to change the height of the lens.

B3 – Mapping the User's Distance to the Display Wall

The ability of users to move freely in the physical space in front of the wall display to control a BodyLens is at the heart of its interaction paradigm. We have seen that such control can be based on a simple unidimensional mapping along the width of the display or different body movements to move the lens vertically on the screen. However, as the user moves in a three-dimensional space there is the additional movement towards and back from the wall-sized display that can be used for additional input. For public displays alone, this movement has already been interpreted to convey a certain interest in content and has been used to define zones of specific engagement and interaction phases [VB04]. As part of the proxemic interaction principle [Gre+11], the orientation and distance of the user has further been evaluated to suggest interest towards content. Following this observation, BodyLenses can be made to progressively react to users nearing the display following a gradual engagement design pattern [Mar+12]. Perhaps the most straightforward realization of that pattern is a progressive fade-in of the lens as the user comes closer to the screen. When the lens is completely materialized, further distance-based interactions controlling a functional parameter can be performed. If the display detects contact input as well, a further “at-the-wall” stage might exist, where users can directly operate lenses and content on the screen using touch, pen, or tangible interactions.

In the following, we propose a number of promising distance mappings that we investigated and implemented within our application scenarios. As previously discussed, depending on the selected mapping and context, an explicit mode switch might be necessary to initiate the explicit change of parameters using the distance to the display wall. Furthermore, any mapping of the distance can either be considered **absolute** in space, i.e., the same position in space will always invoke the same value, or **relative** to the position of the user on activation. While the former supports use of the spatial memory of the user to recall the position of previously identified locations of interest, it may be unsuitable for use with an explicit activation trigger as this may result in very sudden and unexpected increase or decrease of the value on activation. On the other hand, the relative distance-mapping is very well suited for activation by mode switch and supports clutching values, i.e., activation and deactivation for multiple sequential adjustments to support further reach, but lacks the consistent positioning of states in space for improved recall.

We derive four categories of lens control based on distance-mapping: mapping to lens properties such as a) lens position or b) other geometric lens properties, as

Fig. 7.7.: The user’s distance to the display wall can be applied to aid problems of reachability by influencing the *vertical position* of the lens.

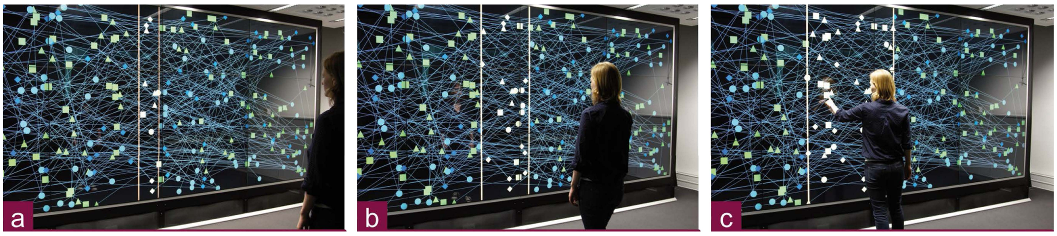


Fig. 7.8.: The user's distance to the display wall can be mapped to the size of the lens (*lens property*) so that the lens symbolizes the area currently occupied by the user.

well as mapping the distance to c) lens function parameters and d) visualization parameters. The suitability of those mappings depends on the application context and task as well as user preferences.

a) Lens Position The user's distance to the display can be used to change the *vertical position* of the lens, moving it up and down by stepping forward or backward (see Figure 7.7). This way, all areas of the display can contain data and can still be explored by the user through the lens. However, the mapping should always ensure that the lens is at eye-level when moving at arm distance to the wall, so that manipulations using touch are still possible. In our implementation, the user can move the lens up and down when farther from the wall. When stepping close to the display wall, the BodyLens returns to eye-level (see Figure 7.7c).

b) Geometric Lens Properties Different *lens properties* other than position can be controlled by changing the distance to the wall. Recall that we define a lens property as any parameter of the lens that is independent from its currently selected function or content (cf. 3.2). One example is the mapping of the distance to the *size* of the lens, i.e., the width and height of a geometric shape, the width of a stripe, or a factor to enlarge a body's mapped shadow. However, different mappings and factors to manipulate size are possible: 1) When stepping back from the wall, the lens becomes larger, thus increasing the spatial coverage of the lens function. This type of mapping coincides with the notion of considering the lens as a shadow with the light source behind the user [Sho+10]. 2) Alternatively, we can consider the fact that users standing in front occlude, and thereby occupy, more data than users in the back. Hence, when stepping back, the lens area becomes smaller (see Figure 7.8). This also coincides with the fact that by stepping back we assume that the user's intention is to get an overview of the data, focusing more on the context, not the lens and its focus area. As with the mapping type, however, its direction may be very dependent on task, context, and user preferences. Similarly, the distance can also be mapped on *shape*, e.g., extending the lens to full height (stripe shape) when stepping back, while only applying a circular lens when near the display wall. Again, this resembles and visualizes the user's focus area and what part of the data they may perceive and thereby occupy while they move around.

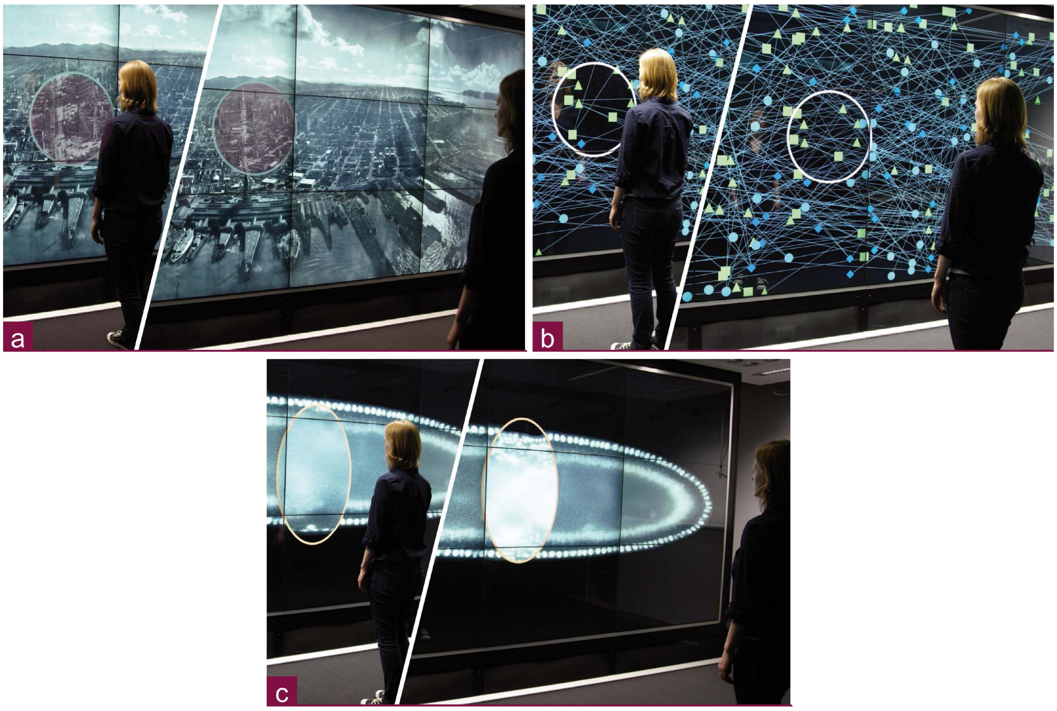


Fig. 7.9.: The user's distance to the display wall can manipulate *lens function parameter*, such as the zoom factor (a) or the transparency of edges (b), or *visualization parameter*, e.g., selecting a time slice (c).

c) Lens Function Parameters Depth movement can also be used to modify *parameters of the lens function*. While not explicitly making use of magic lenses, there are examples in the literature, in which user-to-display distance has been used to modify zoom [HD08] or levels of detail [Dos+14] and abstraction [Leh+11] of visualized data. Depending on the lens function and the parameter to modulate, the distance-to-variable mapping can be continuous or discrete. The former associates a variable with continuous values to the distance between the user and the screen (e.g., zoom factor or edge transparency as can be seen in Figure 7.9a-b). The latter often implies the partition of the physical space in front of the display into several zones corresponding to different discrete states of the variable (e.g., layer type). Another example would be the use of a semantic zooming lens in a data visualization application: By stepping closer to the display wall, data items expand and the user can see more details such as multivariate attribute data. Based on our implementation, we suggest that when the lens is used not only for exploration but also manipulating and editing purposes, it is important to define a distance at which the user can simply look at the result without setting a lens function parameter for manipulation.

d) Visualization Parameters Finally, general *visualization parameters* of the data can be influenced by the user's distance to the wall. One example is the mapping of the third dimension of the information space to the distance, so that movement from and towards the display actually maps to depth movement in the *3D virtual space*. This way, we propose the local exploration of arbitrary slices of volumetric or spatial data by using BODYLENSES. Another important dimension in today's data sets is the special property of *time*. Data such as the measurements of traffic behavior or the development of a tumor are often analyzed with respect to time. Comparing different time steps with each other and slicing through these time-dependent data sets are essential tasks during analysis. In our image analysis prototype, we map the user's distance from the wall to the time dimension (see Figure 7.9c). To the best of our knowledge, we are the first to propose this mapping of body movement to time. It allows the user to individually select a desired time frame, which can often be changed only globally. This further supports parallel work on the whole data set.

Challenges of Distance Mapping As becomes clear from the discussions above, there are certain challenges that arise when applying the user's distance to the wall-sized display due to natural limitations and properties of human perception and memory as well as the mental models of individual users. Noteworthy examples are i) the direction of mappings, ii) the support of comparison of content, and iii) the limitations of visual shrinking of objects when farther from the display wall.

i) In our observations, we found that even for the simple case of zooming, users were at odds on the **direction of the mapping**: Some users prefer having the detailed view close to the wall and the less detailed view from afar, supporting the assumption that stepping back equals focusing on the context. Others preferred seeing the details from afar, helping them to better compare that focus area to the context data. Hence, while there may certainly be a tendency of any application designer to make a decision depending on the tasks to support, the character of the lens as a personal embodiment practically encourages the support of individual preferences. Specifically, the direction of the mapping should be considered a setting on a per lens basis or can even be set automatically when including user identification.

ii) We described the concept of stacking different views onto the data in front of the wall so that users can step back and forth to change the view. In that context, it is a very common task to **compare different slices or views** onto the data. For example, when time is mapped to the distance to the wall, the user can compare different time cuts by moving towards and away from the display. However, while temporal separation may highlight certain changes, it may be difficult to recall a list of changes while only seeing one view at a time. There are a range of possibilities to extend the BODYLENSES to further support this task, e.g. freezing or decoupling the lens to create a new one for comparison, presenting additional views with difference visualizations along its border after selecting a reference slice, or allowing multiple

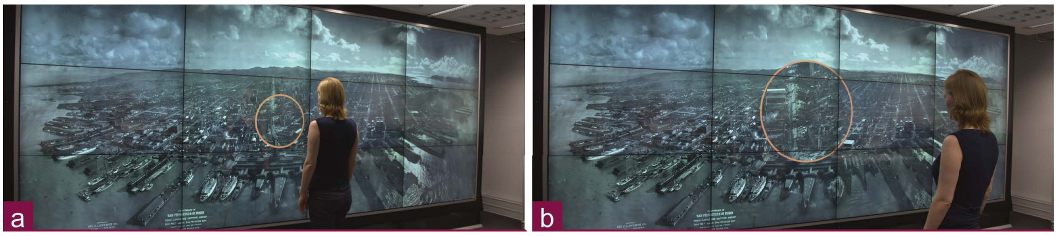


Fig. 7.10.: To compensate for limitations of perception from afar, the lens should apply additional zooming to support the same visual size when manipulating other parameters using the distance.

users to work cooperatively. Some of these examples will be discussed in more detail in the section on multi-user and multi-lens aspects (see section 7.3.3).

iii) Another challenge when comparing different slices of the visualization by stepping back and forth arises from the limitations of human perception, i.e., **visual shrinking of content at larger distance**. When stepping back from the wall-sized display, e.g., slicing through time, the elements visualized on the wall-sized display are at a larger distance and hence appear smaller from afar. This makes detailed comparison difficult. To be able to still properly compare these layers, it is important that users see the same amount of detail and content that they saw before when being close. The BodyLenses may counteract these limitations: We propose the concept of automatically enlarging the BodyLens and the elements within to compensate for the user's movement. Thus the user *perceives* the content to be always the same size (see Figure 7.10) regardless of the distance, which is used to control another parameter like time.

7.3.3 C: Multi-User & Multi-Lens Aspects

Another important dimension to consider is the number of lenses and users interacting simultaneously at the wall-sized display. We can differentiate between single-user and multi-user scenarios whereas the scenario with multiple users may include times of close and loose collaboration (e.g., [Ise+12]), i.e., both parallel individual exploration as well as joint collaborative work. In both cases, the **cardinality of lenses and users** strongly influences the possibilities of lens usage: While in the general case of the BodyLenses concept each user is associated with his or her own personal lens tool which remains coupled over the course of interaction, the scenario can be extended to a single user working with multiple lenses, placing them at regions of interest for a certain time to pick them up later or to look and compare different parts of the information space using a lens for each focus point. This scenario already entails the need for novel concepts to support visual comparison tasks, e.g., transferring settings and lens function parameters between lenses to keep them consistent for comparison. Finally, for multiple user scenarios the 1 : 1 user to lens association can be dissolved by merging or sharing lenses. In the following, we first discuss aspects considering the "simple" case of one lens per user in multiple user scenarios and later discuss chances and challenges arising by decoupling the lens from the user and supporting multiple lenses each.

C1 – Mutual Awareness and Focus of Attention

A fundamental advantage of BodyLenses in multi-user contexts is their ability to provide mutual awareness (see Figure 7.11b). Much like digital shadows, which are known to possess that property [STB07], BodyLenses that follow users continuously indicate their current locus of attention (i.e., "I am currently working on the content in that lens"). Those moving lenses are noticed by the other collaborators who can then adapt their behavior accordingly. This can further be extended by providing additional feedback, e.g., by color encoding the lens border, to indicate what type of interaction and task is currently performed by the user. As with shadows and visual feedback reflecting users' motions and postures, we imagine that the awareness benefits of BodyLenses particularly manifest themselves in co-located situations as users move around separately in front of the same display wall. This is likely true to an even greater extent in remote collaboration conditions where the lack of physical co-presence of other participants needs to be further compensated (similar to what virtual embodiments achieve for remote collaborative whiteboard applications [Zil+14]). In such settings, the lens would therefore function both as a tool to interact with the data as well as a mechanism to maintain mutual awareness between workers (co-located or remote). Finally, we want to mention a particular category of multi-user scenarios in which the user's body plays a role not only as input to control a BodyLens, but also as a physical obstacle in front of the display. Those cases occur particularly in situations when privacy needs to be safeguarded in (semi-)public environments. Specifically, the body blocks access to other people in the vicinity, while the lens creates a confined view of the private content that can only be visualized by the owner [Bru+14]. However, this restriction of access is not the main concern of BodyLenses as we see their potential mostly in supporting collaboration phases at large vertical displays.

C2 – Personal Preferences and Personal Tools

If the system supports user identification keeping the user's id consistent within a session or even over time (e.g., using additional Kinect sensors [Zad+16] or accessing hand contour [SCG10] or finger print [HB13] on contact), each lens can be associated with a specific user. As a result, it is possible to customize the lens to the users, their individual roles, and their preferences. Default lens functions and predefined lens function combinations can hence be offered to identified users according to their usual actions and goals. Similarly, the menus and tools can be adapted to fit the user's preferences enhancing the user's **personal tool belt** by adding user specific configurations. For instance, the exploration of a machine's construction plans, different technicians may have different responsibilities and goals when looking at the design. While one user may want to examine heat propagation within the machine and requires lens functions regarding these aspects, another may focus on electrical wiring and cable routing. Both users have to easily access the tools their individual profession and responsibility entails. As another example, we designed a mind-map application enhanced with BodyLenses (see applications section 7.5). This application requires tools for creating and manipulating items. We suggest personal tool belts as a means to arrange and access these menus and tools attached to a lens. Thereby the lens becomes a user's personal territory for work on the mind-map. Parts of these tools on BodyLenses can also function as individual storage containers with access to personal data, which can be accessed on the tool belt of the personal lens. Within this mind-map application, images or previously prepared sketches can be added to the mind-map through the lens. To categorize, sort and manage these items, a part of the lens can function as a clipboard to which items can be attached (see lens attachments, Figure 7.5 on page 152).

In addition to these personal lens adaptations, content created or manipulated within the lens can be tagged with this user information and as a consequence can be highlighted accordingly or even locked to prevent changes from other users. This further supports removing a user's personal content from the display if they decide to no longer participate in the collaboration.

C3 – Merging, Sharing, and Comparing Lenses

When multiple lenses are active in a common data exploration application, they support loose collaboration scenarios where multiple users work independently. Beside their local manipulation and representation of the user's personal territory in these scenarios, BodyLenses may further become a tool supporting exchange between users in a variety of ways. For example, next to the clipboard areas that we

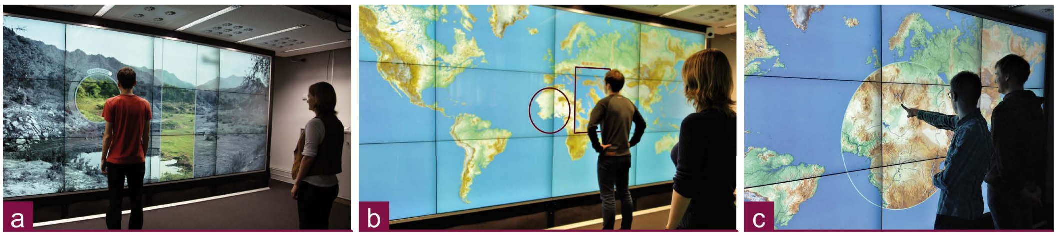


Fig. 7.11.: Lenses can be adapted by their individual users to conform to their requirements, preferences, and goals, e.g., lens shapes (a). The lens can also function as an awareness tool especially of users standing behind (b) and can be merged for closely coupled collaboration (c, mock-up).

described for individual use, the BodyLenses might access joined storage areas that all lenses may access so that exchange of information (e.g. individual data points, extracted information, or identified results) between users becomes possible.

Multiple lenses can also be applied to close collaboration phases where users step closer together overlapping their respective lens territories. As has been discussed for the general concept of magic lenses, the merge for composition of lenses and combining lens functions can be worthwhile for data exploration (see section 3.3.3). Further, in case of personal lenses with user-specific IDs, this inherently results in sharing of private spaces [SCG10; Sho+10]. Adapted to BodyLenses, those applications may rely on multi-user embodied interactions to collaboratively create compound lens functions and grouped spaces (see Figure 7.11c). Thus, BodyLenses that completely or partially merge through concerted user actions can form common embodied territories with shared properties and elements, e.g., a coalesced convex shadow lens within which users work together on a common task. This can be viewed as an application of F-formations [Ken90; MHG12] and more generally of proxemics [Hal90; Gre+11]. Thanks to the very dynamic nature of BodyLenses, these ad hoc groups with temporarily joint territories can be quickly formed and split as users move towards and away from each other and may ease the transition between loose and close collaborations.

Even multiple non-overlapping lenses have notable benefits when data needs to be viewed side by side for comparison, as has been shown for magic lenses [BHR14]. This has also been investigated for body-based data views (albeit not as lenses per se) by SpiderEyes [Dos+14] where parallel, user-associated stripes allow selection of varying datasets and details. Extended to flexible and versatile BodyLenses, this paradigm allows groups of people to collaboratively examine multiple arbitrary facets of a common dataset through freely moved, scaled, and adapted lenses using natural proxemic and body-driven interaction. Considering this visual comparison task [Gle+11; TFJ12] makes clear that the use of multiple lenses may improve side-by-side comparison even for an individual single user. As such, it may become necessary to decouple and place the current BodyLens to be able to create additional lenses that can then be refined. By decoupling and re-coupling lenses to the body, the BodyLenses concept can be extended to support the flexible exchange of lens tools between users.

7.4 Technical Realization

The BodyLenses prototype builds on the same basis for lens and graph implementation as the GraSp prototype and was extended with components for map coordinate management (regarding the placement of data objects in latitude and longitude coordinates) which have also been used for DI.VI.CO. Furthermore, the extension of the graph prototype for use as a mind-mapping tool was created as well as components to manage and present image stacks on the wall-sized display. The prototypes were implemented in python using libavg [@lib03]. Consistent with the previous prototypes, the lenses are managed by a *lens control* while each contains a set of lens functions as well as a shape component that is responsible for assessing if elements are within or outside the lens.

To control the lenses, a Kinect sensor [@Mic13] for tracking the users' body movements was added (see Figure 7.12). This required the implementation of an additional skeleton manager which handles the creation, update, and deletion of skeleton objects depending on the received OSC stream from a separate computer connected to the Kinect sensor (see Figure 7.13). It is placed opposed to the wall-sized display and thereby tracks the users from behind. The *SkeletonLensManager* matches and organizes the skeletons from that skeleton manager with the lenses to create the BODYLENSES using a set of possible mappings (defined as *SkeletonToLensMapper*). In the simplest case, the mapper (*BasicPositionMapper*) converts the position of a single skeleton joint to display wall coordinates to manipulate the position of the lens. More advanced mappers apply a range of transformations and combinations of skeleton joints to form a body-centric shape, e.g., a shadow of the user, and as a result manipulate the polygon shape of the BODYLENS. These mappers may incorporate different states depending on the user's current hand state (e.g., fist) to adjust their mapping and allow explicit dynamic shape manipulations by the user.

The *BodyLens* (subclass of the original *Lens*) references an instance of *Person* which contains information on the position of the user and is equally updated by the selected mapper. Some of the existing lens functions have been extended to update depending on this position information to support parametrization of the lens

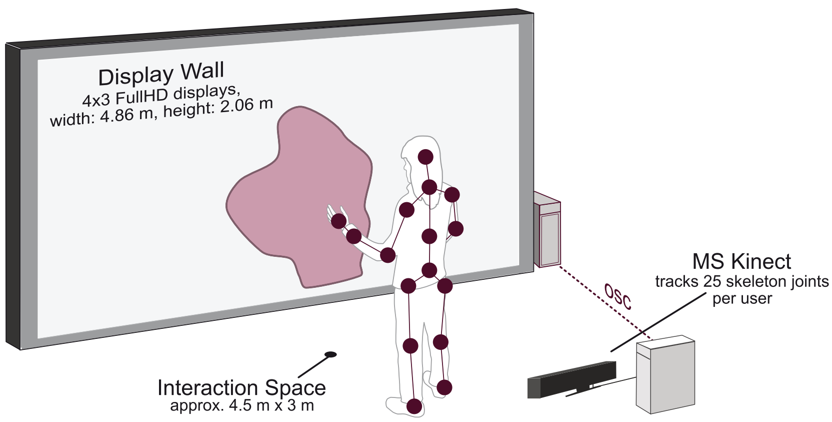


Fig. 7.12.: Setup of the BODYLENSES prototype

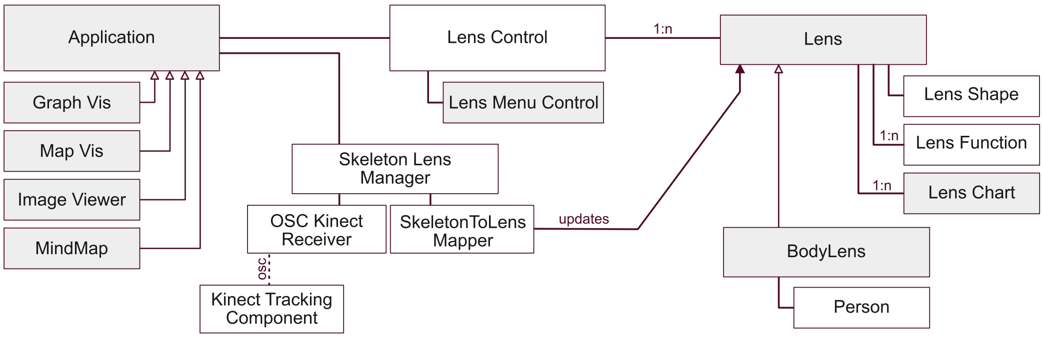


Fig. 7.13.: The BodyLenses prototype consists of an interchangeable application component that contains both the *lens control*, which manages the manipulation of the content through lenses, and a *skeleton lens manager* which responds to the Kinect data and triggers any updates of the lens and its subclass *body lens*.

function according to the user's distance to the display wall. Additionally, the *LensViews* have been updated with varying components for managing additional lens menus (*LensMenuControl*) or lens widgets, e.g., additional visualization charts (*LensChartControl*) as needed for the application cases.

7.5 Example Applications

We implemented a variety of application use cases which were used for our practical investigation into the potential and possible challenges of the BodyLenses concept. These application cases, the basics of their interaction vocabularies, and their use for explorations into selected dimensions and BodyLens principles will be summarized in the following sections. However, this description will focus on the *graph explorer* and the *map-based data analysis tool* as they are directly relevant for the interactive exploration of information visualization. Minor applications, such as our *image explorer* which includes exploration of time-dependent image stacks, the *mind-mapping tool*, and the small *artistic application* will be addressed in short summaries focusing on individual BodyLenses properties and dimensions that were explored with them.

7.5.1 BodyLenses for Graph Analysis

For the graph explorer, we used the one user – one lens association enabling the exploration of node-link-diagrams for multiple users where a lens is implicitly created once a user steps closer to the display wall (threshold: 2.5 m). Upon creation a lens is immediately coupled to its user and remains so until deletion when the user steps away from the display wall. An overview of lens interactions is given in Table 7.1. There are various data sets that have been used with the graph explorer, mainly a

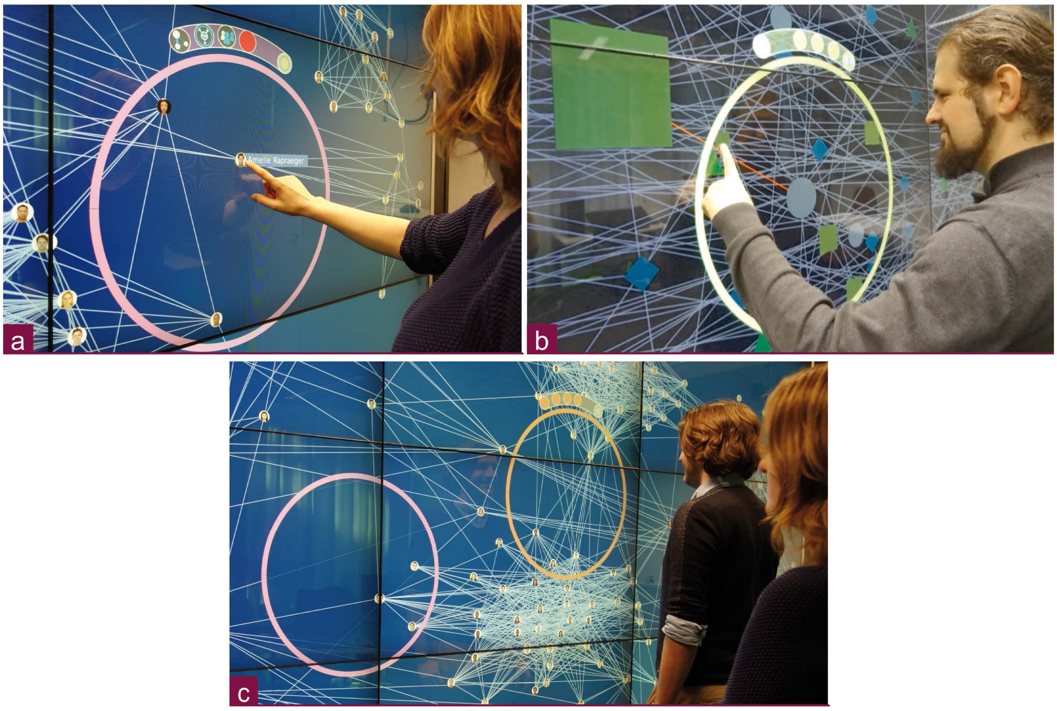


Fig. 7.14.: Our main BodyLenses application focuses on graph visualization (a) supporting the multi-user exploration of graph data (c). Therefore, we also adapted lens functions for the use within a lens territory, e.g., using proxies within the lens connected to neighboring nodes outside of the lens border (b).

social network data set that was extracted using the Facebook Graph API [@Fac15], the human disease network [Goh+07], and VIS publication information (authors and/or papers) from the vispubdata.org project [Ise+17]. However, any data converted to graphml file format can be loaded into the application.

We implemented a set of known graph lens functions (cf. section 3.3.2) and extended them for use at a wall-sized display considering both the possible use of distance and collaboration. Proxemic-dependent version of Local Edge and Fisheye lens functions support the manipulation of their primary parameter, transparency of edges and zoom factor respectively, through the user's distance from the display wall. Furthermore, the Bring Neighbors Lens [Tom+06] was revised to create and move proxy elements instead of the original neighboring nodes to counteract the limitations of the global effect of this function which could confuse other users as nodes would be pulled away from them. The proxies which are smaller representations of the actual node appear at the border of the lens in the direction of the actual neighbor and allow highlighting of the edge to the neighbor on touch interaction (see Figure 7.14b). These functions can be selected using a menu at the border of the lens.

The graph application was also used to explore the different shapes and manipulation of shapes (e.g., Figure 7.1b, page 146). We used the fist gesture as a trigger to allow gestural manipulation of the shape. For primitive shapes, we support increase and

create lens	stepping closer ($< 2.5\ m$)	Local Edge Lens: transparency	distance to display wall (after activation)
freeze lens movement	stepping close to display ($< 0.5\ m$)	Fisheye Lens: zoom factor	distance to display wall (after activation)
delete lens	moving away ($> 2.5\ m$)	manipulate height of lens	varying exploration triggered by fist closed • y-position of highest hand • rel. movement (step) to or from display • abs. distance to display
move lens relative to user	touch-drag to set offset		
toggle lens functions	function menu		
switch lens shapes	shape menu		

Tab. 7.1.: Interaction mapping of BodyLenses graph analysis tool focusing on basic lens manipulations (not including interaction with nodes and links).

decrease of either width or height or both simultaneously. For body-centric shapes, we tested different mappings and assessed whether and how strongly dependent movement of the shape should be for interaction. Finally, we also explored different mappings for the control of reachability of all areas of the information space. The application supports setting a relative offset of the lens from the body using touch which remains consistent during movement. It is also possible to temporarily connect the lens to the hand to steer the lens to the top or bottom of the screen. Beside parameterizing the lens functions, we also tested the application of the user's distance to steering the lens' height (see Figure 7.7, page 155) which however heavily restricts the user's movement by forcing a specific behavior and hence was evidently unsuitable for this kind of graph explorations.

7.5.2 BodyLenses for Map-based Visualization

For the map-based data visualizations, we extracted European power plant information from Enipedia [@Dav+15] to visualize the various power plants, their category of energy, CO_2 emissions for 2000, 2007, and 2020 as well as capacity and intensity of energy production[2]. Furthermore, we also applied this prototype for the crimes data set of Baltimore (cf. DI.VI.CO prototype data, page 131).

In this application BodyLenses are not strictly coupled to a single user but can be put down on the visualization to be picked up later. This way users may create multiple lenses for themselves creating new ones where needed. Upon creation, the lens is only attached to the user if no other lens is currently coupled. An overview of lens interactions is given in Table 7.2. For this application we applied attribute filtering lenses that reduce the number of data objects shown within the user's focus view. They can be configured to completely remove items within the lens, slightly fading

[2] The data extraction and integration from Enipedia was part of the student project of Marc Satkowski.

Fig. 7.15.: The implementation of BodyLenses for map-based visualization explores the use of lens functions with separate views that are attached to the lens and move along with the user (a) as well as attribute filter functions that reduce the clutter of items within the BodyLens and highlight nodes with selected properties (b).

them, or replacing their texture with additional information, e.g., showing a knife, firearms, or fist texture depending on which weapon type has been used for a crime. This application helped us investigate the use of additional visualizations coupled with the BodyLens. By adding specific chart lens functions, the lens provides additional views attached to the lens border that provide aggregated information on the data selected by the lens, e.g., number of crimes per type (see Figure 7.15a). By adding multiple lenses onto the visualization, these additional charts support comparison of different regions.

create lens	• touch-hold on background for default size • encircle data points
(de-)couple lens to/from user	double-tap lens border
delete lens	trigger close button in menu
move uncoupled lens	drag on lens border
toggle lens functions	function menu

Tab. 7.2.: Interaction mapping of BodyLenses map visualization focusing on lens manipulations (not including interaction with data objects).

7.5.3 Further Applications for BodyLens Explorations

Three smaller example applications have been used to explore varying properties, dimensions, and principles of BodyLenses including the use of distance to the display wall, additional widgets and menus, and manipulations of the content.

Image Analysis / Time Stack The image analysis tool and time-based image stack exploration helped us investigate the basics of the BodyLenses principle through a range of simple to grasp use cases and includes different mappings of the user's distance to the display wall. To that purpose, it contains a variety of small application cases that have been simplified to image exploration for quick prototyping and testing. We implemented a range of lens functions starting from i) basic image magnification

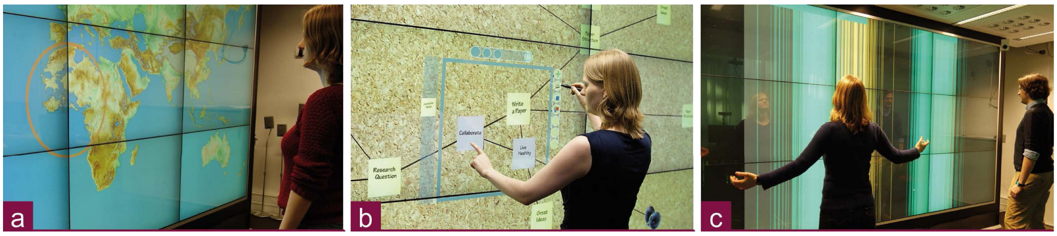


Fig. 7.16.: Other example applications were designed to explore the potential of BodyLenses: The image analysis prototype supports exploration of the earth's relief (a). The mind-mapping tool enables creation and collection of notes (b). A small additional application explores body-movement for artistic expression (c).

as a core concept of magic lenses in general, but also ii) manipulating saturation or brightness values to present the principle of lenses and focus and context in an easy to understand way, and adding iii) the selection of a specific slice of an image stack to emulate a range of parameter adjustment, e.g., selecting a time step.

We used these lens functions to explore the variety of possibilities BodyLenses offer through the user's movement to and from the display wall. The prototype supports the use of this distance to increase a zoom factor of the magnification lens function, zooming into high-resolution images to explore their details (e.g., San Francisco after the 1906 earthquake[3]). By adding a stack of images of the world at different levels of water rising[4], we allow stepping to and from the display wall to explore the world's relief surface (see Figure 7.16a). Additionally, the time-based photographs of the development of a drosophila (i.e., fruit fly)[5] support stepping through time to identify its different steps of evolution (see Figure 7.9c, page 157).

Mind Mapping The mind mapping application[6] was designed to investigate the concept of BodyLenses in regard to their use as personal lens territories providing personal menus and tools. As such, a range of widgets and tools are attached to the mind-map BodyLenses which only appear when close to the display wall, i.e., in range for touch interaction, to prevent occlusion of content while the users discuss. The menus include tools for adding either existing image notes or clean notes for writing own content using a digital pen (see Figure 7.16b). These notes can be connected to one another using the pen or sequential tapping for edge connection. Furthermore, the lenses contain clipboards to temporarily store notes and items for later use. Next to presenting a personal set of tool menus and clipboards, the mind-map BodyLenses can be used with a range of lens functions, which we tested at different stages of the prototype. It is possible to present additional details of a

[3]Photograph of San Fransisco in ruins openly accessible at http://hdl.loc.gov/loc.pnp/ppmsca.07823
[4]Water rising images are free to use from http://www.reliefs.ch/meeresanstieg/weltanstieg.htm.
[5]Courtesy of Max Planck Institute of Molecular Cell Biology and Genetics
[6]The mind-mapping prototype was implemented onto the base application as part of the Bachelor's thesis of Maximilian Gräf and extended afterwards.

note (e.g., author, category, person responsible) or expand collapsed sets of notes. Furthermore, we applied an association to individual users as owners to notes to test locking of individual notes to prevent manipulation by other users. These associations can be made visible by highlighting the notes with the users' colors. This could also be used to assign notes, e.g., certain tasks, to specific users at the end of the mind-mapping process.

Artistic Application In our small artistic installation, we use BodyLenses to paint and manipulate color stripes on the screen. Here, the users' distances to the wall-sized display changes the color of their individual stripe. Furthermore, the stretch of their arms influences the size of the lens and thereby the size of their current manipulation effect (see Figure 7.16c). Based on our implementation, we strongly suggest that, when the lens is used not only for exploring but also to manipulate content, there needs to be a comfortable distance at which the user can simply look at the result without changing it. We additionally use a dwell time before any stripes are painted onto the display to support a conscious manipulation process and prevent unintentional painting. Different alternatives have been implemented for the distribution of resulting lines within a BodyLens stripe, e.g., Gaussian, equal, or random distribution, to try-out different types of results. While not necessarily a productive work tool, the artistic installation supports the joy of easily creating joined art with multiple users.

7.6 Discussion and Summary

This chapter presented an exploration into the design space of body-controlled magic lenses. It describes and discusses the various possibilities that emerge by creating BodyLenses including the appearance and shape of lenses, the mapping of user position and movement to lens properties, as well as various possibilities for mapping the distance to the wall-sized display to configure the lens. Finally, aspects of multi-user support are discussed aiming at making the lens a personal tool and toolbox. A majority of these aspects were implemented in a range of different application prototypes where we investigated specific aspects and possibilities of the BodyLens principle. This design space and the experiments with our prototypes show the potential of BodyLenses, but they also identify a range of challenges and open questions that will need to be investigated in future research. Specific aspects in the following discussion are the use of implicit interaction that could trigger unintentional reactions by the system and the advantages of relying on physical navigation at the risk of enforcing movement for interaction. Finally, we consider the learnability of interaction gestures for the actual real-world application of BodyLenses.

A part of the appeal of BodyLenses is their inherent combination of **implicit and explicit interactions**. This is ingrained in the basic principles of BodyLens interaction as the simple movement towards and along the wall-sized display already implicitly manipulates the view while explicit mid-air gestures and active touch interaction are used for fine-grained, specific adjustments of the tool. This transition between implicitly stepping closer to explicit manipulation is at the core of BodyLens interaction. However, it also raises questions regarding any mapping of interactions and effect depending on what is required and appropriate for the specific application case and context. One major aspect is the selection of suitable features for implicit movement. This especially requires careful consideration as unintentional manipulations could be invoked and thereby undermine the users' trust in their control of the system. In the presented examples, we suggested the benefit of providing more details to the user or slightly increasing or decreasing a parameter value. Inherent in that is the fact that implicit changes should not trigger very prominent manipulations or should in that case only be applied in specific conditions or modes that can be activated by explicit interactions. This aspect equally concerns the mapping of features to the distance of the user to the wall-sized display.

In chapter 5 and 6, we found participants of our studies to move back and forth from the display repeatedly and discussed the benefit of seeing details when close to the display while also focusing on getting an overview and orienting oneself when stepping back. This **flexible movement** of users was considered beneficial. In addition, we already discussed that physical navigation aids users in spatial orientation and memory [BNB07; AN13] (see also chapter 2). The concept of BodyLenses strongly relies on this ability and aims to apply it for exploring information spaces. However, with mapping this movement, specifically the distance to the display wall, to BodyLens' features and manipulations, BodyLenses **force certain movements** of the user to reach their goals. That is, to set certain lens parameters or change the lens size users will have to step back or forth to work with the lens. While the advantages of remembering one's position remain intact even in this kind of guided movement, especially as left and right movement is not affected, other advantages of physical navigation might be hindered and flexible positioning (which we focused on in GraSp and di.vi.co) might be reduced or, in the worst case, eliminated.

Furthermore, we know that not all users apply physical navigation when giving the chance: In our GraSp and di.vi.co evaluations (see section 5.4 and 6.4), we observed some participants who were lazy in terms of movement and used the advantages of the remote systems to eliminate having to physically move. The BodyLenses system however applies this movement and makes it necessary, not only to configure the lens but to explore the entire information space. It should be investigated further if the actual movement of the body is worthwhile or if the sufficient display space (cf. space to think [AEN10]) alone can be beneficial enough for improved data exploration. This would hence regard the question if we could increase the efficiency of these users by encouraging them to move as is the case with BodyLenses or whether tools should be designed that rely on their preference and do not require them to move, as has been applied in GraSp and di.vi.co.

Another important aspect when considering the actual application of BodyLenses to real-world scenarios is its intuitiveness and **learnability**. With its implicit movement, we argue that certain parts of the interaction are intuitive and easily understood. This especially concerns increasing the degree of an already existing function inherent in perception: Stepping closer to the wall-sized display provides more information, more detail, and more focused content. Other interactions become apparent through this natural movement where the body-centric shape of the lens automatically adapts to the current user position and hence, naturally, deforms when the user stretches. These types of interaction are learned by simply trying them out and moving to observe the resulting change. However, we also argued for an explicit trigger for radical shape changes and adaptations, i.e., we described the fist as a possible gesture for invocation. How can a user identify both the possible functionality of resizing the lens as well as its required gesture? These additional manipulation gestures require recall. Since BodyLenses resemble shadows and hence could function as avatars, a tutorial stage of the application could use this characteristic to introduce gesture and movements that users can perceive, follow, and copy to learn interaction. In the current version, all essential features of the lens for data exploration, e.g., activating lens functions, are further integrated into the touch-enabled menu around the lens and can be triggered on recognition alone. However, for application to real-world scenarios this learnability of body movement and gestures needs to be investigated further.

8 Discussion, Contributions, and Directions for Future Work

At the beginning of this thesis, I argued for the need of tool development for data analysis and visualization in novel display environments. The increased availability and reduced cost of display technology enable multi-display spaces and wall-sized displays as a possibility for data analysis. This development also resulted in commercial solutions that incorporate the novel technologies to adapt to specific requirements in business, e.g., the MS Surface Hub [@Mic15] is a 84″ touch and pen-enabled display space for presentation, video communication, and meeting scenarios. At the same time, the evolution of smart mobile devices and their wide-spread use increase the potential of multi-device environments while also making users familiar with novel interaction technologies like touch and pen. This trend makes development of natural user interfaces not only an interesting possibility but a necessity for future work environments.

For the specific case of visualization for data analysis and exploration, I identified magic lenses as suitable tools that support temporary local manipulation of the view. Magic lenses have been shown to be flexible in terms of usage tasks and a variety of solutions have been provided by means of diverse lens functions in research. With the categorization and analysis of lens functions, I have shown that they have been useful for exploration of graph visualizations (e.g., [SB92; HW04; Tom+06; LAM10; Gla+14] but also for a range of other visualization techniques and data types [Tom+17]. I have identified limitations of the current application of the tool for visualization and have refined research goals that were successfully addressed in this thesis. This includes

- the transformation of magic lenses into flexible tools for graph analysis through **functional extension** including their configuration to fit user-specific needs,
- enabling graph exploration at large display spaces by creating **interaction design** with novel interaction modalities to support lens usage,
- enabling fluid, effortless interactions to manipulate and configure lenses using alternative **interaction styles** adapted to users' preferences and experiences, and

- the development of magic lenses as personalized tools for multi-user scenarios and according **empirical investigations** into user behavior for data exploration with lenses.

8.1 Overview of Insights and Contributions

The following gives an overview of the contribution of this thesis with regard to the identified challenges and thesis goals for magic lenses.

8.1.1 The Systematic Analysis of the Magic Lens Concept

This thesis pursued the exploration and extension of magic lenses in terms of both function and interaction. As a basis for this endeavor, the principles, concepts, and existing use of magic lenses had to be well understood. Therefore, I presented a survey of research work on magic lenses and lens-like principles.

In the analysis of the magic lens concept, I initially defined the concept, examining its initial description and related techniques and as a consequence discussed its relation to the visualization pipeline [CMS99]. As a foundation for later configuration of the tool, I identified lens properties describing the lens selection and essential parameters characterizing the lens function. Chapter 3 presented an overview of existing lens functions from related work focusing also specifically on graph lens functions and categorizing them by user task, effect class, and effect extent. This general categorization of lens functions can help support the description of past and future lens publications. In preparation for making the lens a flexible tool, the chapter also discussed issues and strategies for lens function combination including the challenges of lenses with multiple flexible functions as well as the possibility of multiple, diverse lenses on the same context visualization. Finally, an overview of existing interaction principles for lenses was presented where they were categorized by interaction modality used. Within this overview, limitations and open research questions became clear.

A major issue identified was the inflexibility in using magic lenses. This is to say that previous research focused on lenses with a single lens function proposing highly relevant and novel concepts for temporary manipulations. However, they have not focused on the possibilities of these lens functions within a more complex workflow and the varying needs and tasks of the user. Other research work, focusing on the interactive properties of the lens, often regarded magnification as the only essential feature of magic lenses and hence applied the lens to improve imprecise selections or simple overview and detail views without considering their rich potential for data exploration. While many of these issues are addressed in the contributions of this thesis, the analysis alone enables a more thorough understanding of the advantages of magic lenses as well as the potential and currently open issues within magic lens research.

8.1.2 Flexible, User-configurable Lens Tool

A core contribution of this dissertation is the extension of magic lenses from rather static single-purpose instruments to flexible, configurable multi-functional tools integrated into the users' workflows and adaptable to their tasks. This incorporated developing strategies for **functional extensions** regarding the interactive manipulation of the lens. In the most basic case, this covers changing lens properties, e.g., interactively refining the lens' size and shape to fit it to the current region of interest. More challenging however, this also includes supporting the manipulation of lens functions, e.g., selecting one or multiple lens functions in a single lens and supporting multiple lenses on the same context visualization. To make this possible, strategies for managing the iterative modifications of individual lens functions had to be designed and implemented. Extending this further, the flexible configuration of lenses includes setting the various lens function parameters. For instance, this can be defining a single value describing the zoom factor for magnification lenses but also configuring multivariate attribute properties for filter functions.

The functional extension was initially developed for multi-touch interaction (MULTILENS, chapter 4) and the concepts have been adopted in all following interaction designs of this thesis. While the functional strategies for lens combination and lens function application remain the same, the interactive possibilities of changing functions and parameters have been iterated throughout the diverse interaction modalities and according requirements. In all of the presented interaction designs, the parametrization and configuration of the lens is handled in-place at the lens. This is essential for large wall-sized displays to reduce global menus that would require additional movements and create possible conflicts between users. Furthermore, the presented approaches focused on enabling alternative, flexible, and **user-specific interaction styles** to manipulate lenses. These styles are created to adapt to the users' experiences or preferences. MULTILENS presented both widget-based and gesture-based interactions where the widget-based approach has been specifically designed to ease users from simple tap interactions to more fluent interaction flows using the drag-snap slider technique. With DI.VI.CO (chapter 6) and GRASP (chapter 5), interactions both at the wall-sized display as well as distant interactions can equally be performed. BODYLENSES extend these works by allowing additional mid-air gestures and body movement for lens configuration and manipulation.

8.1.3 Personal Lens Tool and Territory

By enabling these individual configurations of the lens with diverse interaction styles and user-selectable properties, this thesis introduced lenses as a personal, user-specific tool. Thereby, the lens can be adapted to each user, their preferences, and their current needs. As an extension of that, the lens has even been personalized to follow the shape of the user's body (BodyLenses, chapter 7) becoming a shadow-like lens that is more than a tool for its user but also a factor of awareness for others. As a result the lenses become both **personal tools and territories** for the users. This personal territory character naturally arises as the users take their personal lens with them, either by carrying it as a mobile device (GraSp, chapter 5) or as it follows the user automatically (BodyLenses). In both cases, the actual lens functionality is integrated into a more extensive tool that supports attaching other views, personal clipboards, or menus, either at the border of the lens or on the mobile device. Due to the personal association, the tool highlights the current awareness of the users and what they associate as 'their' content for the moment. For mobile lenses, this content is specifically copied from the large display wall onto the personal display for individual exploration and only the feedback of where a user is focused on remains.

8.1.4 Embodying of Lens Representation and Controller

To enable the previously discussed configuration of magic lenses and investigate the suitability and usability of the tool for data exploration, this thesis proposed a set of interaction techniques to magic lenses for data exploration. Only due to this iterative **interaction design** for the varying interaction modalities, the lens developed to become a flexible tool adapting natural interactions for data exploration on large displays. That process comprised the design of interaction concepts as well as the realization and refinement of prototype development. Multiple iterations and evaluations were made to improve and refine the presented interaction techniques and investigate user behavior and positioning.

Initial interaction design focused on modalities closely coupled to the interactive surface. We applied different styles of multi-touch interaction including the continuous gestures and the widget-based approach for fluid configuration of the lens (MultiLens, chapter 4). Building on these initial functional extensions and configuration possibilities, we addressed the specific requirements and advantages of wall-sized display environments focusing on the interaction space in front of the display. Therefore, we designed a mobile lens concept that placed the lens into the hands of the users while being able to move and flexibly position themselves in front of the context visualization (GraSp, chapter 5). However, this also separated the focus

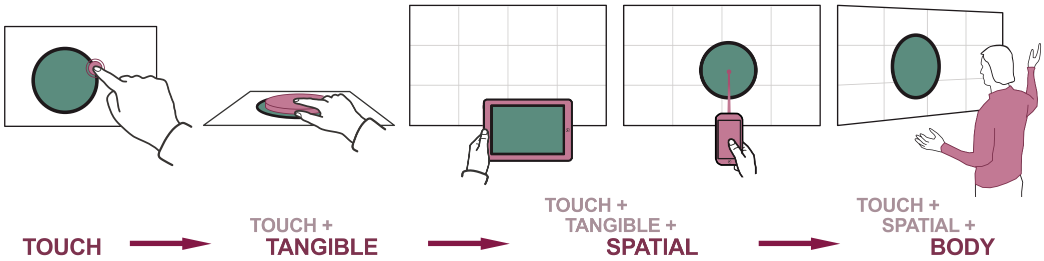


Fig. 8.1.: The proposed interaction modalities and device setups for magic lens application increased the stage of embodiment. However, they also result in a separation of lens and controller.

region defined by the lens from the context visualization. As a result, we reduced the required attention switches by using the mobile device only as a controller for the lens on the wall-sized display (DI.VI.CO, chapter 6). Finally, we designed and developed the concept of BODYLENSES – removing the additional device and using the body as a controller to manipulate the lens on the display (chapter 7).

Within this process and with each new enhancement of the setup, the stage of **embodiment** of lens interaction was increased (see Figure 8.1). We started by making the lens itself tangible and then increased the reachability of the user's arm through pointing until finally using the body as a controller. Each work created a physical representation of the lens or its controller closer to the body for improved manipulation of the tool. This improves interaction as it builds on the advantages of our body, environment, and social awareness and skills [Jac+08]. At the same time, this development increased the size of individual movement, relying more strongly on proprioception, and focusing on an increased movement of the body in general. This activity of the body can take advantage of human perception and spatial memory while exploring visualizations. While research still has to consider the disadvantages of prolonged activity resulting in fatigue and high physical demand (studies have been restricted to limited time), an enhancement towards a physically more active work environment might also be of great benefit for health and productivity (cf. active office [Pro+12]).

8.1.5 Flexible Positioning and Remote Control at Large Vertical Displays

In conjunction with the embodiment and increased physical sense of lens and controller, this thesis also considered the necessity for flexible movement of the users in front of the vertical display. This includes the possibility of walking and moving around as much as the decision to keep standing or sitting in a preferred position. **Flexible positioning** can especially be of advantage when a group of users stand together for close collaboration phases. Here, staying together helps support collaboration [Haw+05] and we also observed this behavior as being naturally preferred by users when working together (cf. DI.VI.CO study, section 6.4). As a result, we aimed to give control to the user from any position in front of the display. GRASP allows this by collecting data from the display wall and presenting it on the mobile device close to the user. While this enables very individual exploration with a personal toolbox, it also reduces the connection between focus and context view and thereby the advantages of the lens principle. DI.VI.CO and BODYLENSES allow **remote interaction** with the lens on the display wall. Here, the mobile device's position and touch surface or body movement and mid-air gestures were used in addition to touch interaction on the wall-sized display. However, within this process of enabling flexible position and supporting remote interaction, we started spatially separating the lens from its controller (see Figure 8.1).

While this separation of lens and controller may seem like a disconnection, it is yet unclear if there is a disadvantage considering the benefit of the control from any flexible position and the strengthened connection between focus and context. Furthermore, while this separation may appear from an outside view, there is a strong coupling between controller and lens during interaction. BODYLENSES specifically create this association by presenting a strong synchronization of body movement and lens movement and thereby clearly seem connected. The entire principle of **body-controlled magic lenses** is an innovative concept that builds on this connection. In this thesis, I presented our investigation into the possibilities of body-controlling of tools and designed a range of alternative interaction techniques to foster this connection to the user's body. I proposed a design space and a discussion of these possibilities and challenges and realized prototypical implementation to evaluate these principles. I believe there is lots of potential in using the natural body movements to enhance interaction. However, further development and extensions of concrete application cases would be a future step for evaluating the concept of BODYLENSES for specific use cases.

8.2 Discussion and Critical Reflection

The following section discusses the limitations of the presented work focusing on general limitations originating due to the dissertation scope as well as limitations of the techniques and studies presented in this thesis. As a result of these limitations, open questions and possible future directions are identified and shortly discussed.

8.2.1 Limitations

Influence of Device Size and Setup In this thesis, I focused on large vertical display environments and specifically wall-sized displays that support multiple users that interact flexibly in front of the display and require physical navigation for data exploration. All prototypes were designed for this display setup and studies were conducted using a display wall of approx. 4.9 m x 2 m with a resolution of 7680 x 3250 pixels. While some of the presented concepts are clearly independent from display size, it is yet unclear if the observed results would have equally been visible in much larger or smaller context display setups with higher or lower resolution capabilities. The presented concepts were also evaluated specifically in situations of standing participants, moving around in front of the display. While some of the presented concepts (namely GRASP and DI.VI.CO) could similarly function with users in sitting positions, additional studies and possible extension would be required to confirm and evaluate their use in those alternative scenarios.

Laboratory Studies The presented evaluations and empirical investigations were conducted with university students and university personnel in controlled laboratory setups. Data sets were selected appropriate for this focus group and to ease the introduction of participants to the data and their possible goals. However, it would be important to make larger evaluations of the concept with data analysts and their own domain-specific data to distinguish differences in behavior. This would enable a clearer identification of where the proposed techniques enable improved data exploration and where aspects of the approaches need to be reconsidered and iterated to fit individual needs and requirements of specific use cases and domain-specific applications.

Development of Lens Functions This thesis surveyed and categorized a wide range of existing lens functions, also specifically addressing graph lens functions. Within this research, we also inherently applied and adapted these lens functions in ways that could be considered novel lens functions by i) flexibly combining multiple lens functions creating novel, more focused lens output, ii) adapting existing non-lens graph solutions to lens application, and iii) considering the requirements of multi-user interaction, e.g., adding proxies instead of moving graph nodes outside the lens border. However, we did not focus on these specific lens functions and did not specifically evaluate their suitability as this was not within the scope of this dissertation. As such, many more appropriate lens functions could be designed to fit user-specific tasks regarding graph exploration and manipulation.

Dynamic Graph Exploration This thesis focused on static graph data. However, it already discussed time-dependent data visualization and the potential of navigating through time using BodyLenses. Considering the application of our principles to dynamic graphs [Bec+17], we would have to consider the movement of individual nodes and clusters over time. Novel concepts regarding automatic lens update or guidance of where to move the lens would be required to follow elements in focus. Additionally, lens functions would be required that specifically present information on changes that occurred in relation to previous or future time steps.

Multimodality The presented research approaches focus on alternative interaction modalities to invoke the same features and functions: For instance, BodyLenses support touch on the display, mid-air gestures, and physical movement of the body to manipulate the lens. Similarly, both GraSp (in part) and di.vi.co support touch interactions on the wall-sized display and at the same time spatial input for distant interaction as alternatives to accomplish results. Further research should widen this approach to care for the diversity of users and their preferences. In particular, an increased focus on transitions between alternative interaction modalities and synergistic effects of those alternatives should be investigated further.

Co-located Collaboration The research approaches presented in this thesis provide individual, personal lens tools to each user that can be configured and manipulated with user-preferred interaction styles. We further discuss possibilities to apply magic lenses to multi-user scenarios for parallel work (GraSp and BodyLenses) and investigate lens usage as part of the di.vi.co study. In particular, we saw interesting behavior regarding the remote control of lenses on large displays as well as surprising collaborative behavior of pairs of users using one and the same lens. However, there are many more possibilities to extend lenses to closely coupled collaboration especially in regard to the territory-character of the lenses, e.g., by sharing content between lenses or focusing on additional awareness aspects. While this work introduces initial concepts in this regard, there is potential for further exploration of explicit collaboration techniques using lens territories and empirical investigations of how lenses could provide this collaborative support.

8.2.2 Further Reflections

More Than Magic Lenses

Our definition of lenses at the beginning of chapter 3 was very concrete focusing on the characteristics of lenses as transient, temporary, and locally restricted manipulations of the visual representation of the data. While all presented concepts were introduced as extensions of the Magic Lens principle [Bie+93], the proposed extensions and additions to the concept highly relate to the original idea of Toolglass widgets [Bie+93]. Magic lenses are a part of this Toolglass principle which describes see-through interfaces that support the permanent manipulation of the content underneath. By temporarily changing the view, magic lenses ease the selection of visual elements through the tool so that the user can manipulate content and apply properties with the Toolglass. As such, some of our proposed principles are part of the Toolglass concept, e.g., the permanent manipulation of edges within a temporarily changed encoding of the representation (cf. adjacency matrix lens of GRASP, page 113). However, many of the proposed additions to the lens are not merely focused on manipulation. The extensions contain additional exploration tools, configuration menus, and attached views related to the lens function. As a result, the presented concepts in part extend their purpose from a magic lens with additional Toolglass features to a personal, user-defined toolbox for data exploration.

Looking into related work developed in recent years, the term lens has further been used for other tools. For instance, Badam et al. [Bad+16] recently proposed a movable view in multiple coordinated views as a lens: They investigate views that move with the user similar to BODYLENSES and can be controlled via explicit hand gestures. In addition, generic hand-held displays in augmented reality settings have been called lenses [Bar+12]. As described above, GRASP equally merged concepts of magic lenses with a general toolbox showing manipulations of the view that extend lenses. Similarly, any use of mobile devices in connection to large displays may share some traits and characteristics of magic lens principles, showing a manipulation of the view or a separate view that relates to the specific content on the large display. While the relation and connection to the context view may be one essential aspect relevant for distinguishing these concepts from magic lenses, it becomes clear that these extensions open up a wide range of possibilities with a broader usage of the term lens. With that in mind, the proposed concepts for magic lens positioning, manipulation, and personalization can be considered for these extensions as well. As a result, I am confident that this dissertation's interaction concepts and their principles for controlling and configuring lens properties and parameters are general enough to be applicable to many other tools.

Contribution to NUI for InfoVis

The importance of interaction as part of visualization research is well-known to increase the understanding of data [Fek+08; Spe14; Tom15]. Further research has identified the advantages of novel display environments for data exploration, making a case to use these benefits for visualization "beyond the desktop" [Lee+12; JD13; Rob+14]. In that spirit, the contributions of this thesis make considerable progress towards enhancing the use of magic lenses for data exploration. As a result, they clearly add to the development of effective information visualization on wall-sized displays. By investigating and designing the use of magic lenses in this context, this dissertation increases the possible use of interactive tools for data exploration in novel display environments. The proposed principles do not only apply to graph visualization. Some of the proposed interaction concepts have already been shown to be applicable to other visualization techniques as well, e.g., map-based visualization for the DI.VI.CO and BODYLENSES approaches. As I have presented in chapter 3, lenses can be applied to a multitude of data types and visualizations. Since the presented approaches do not depend on specific properties of the visualization or lens functions, they can be applied independent of visualization type. For instance, a Regression Lens [Sha+17], showing the regression model of selected points, could be moved and positioned on a scatter plot using the remote interactions of DI.VI.CO. A Color Lens [EDF11], adjusting the local color scale of map-based visualizations, could be extracted from the context visualization onto a mobile toolbox like GRASP and thereby support parallel exploration of the data space by individual users. This is to say that the advantages of the presented techniques can be transferred to other lens functions and contexts, as much as to more generic types of tools. It's the principles of flexible configurations (MULTILENS), the automatic user-dependent positioning and parametrization (BODYLENSES), the graspable tool in your hand (GRASP) mixed with the flexible remote positioning in front of large displays (GRASP + DI.VI.CO) that create a powerful advancement to data analysis tools and information visualization with natural user interfaces.

Use in Real-World Applications

Interactive wall-sized displays, though not ubiquitous yet, are increasingly available. While first not always interactive in terms of direct touch capabilities, these displays support data visualization due to their resolution and size. Combining this type of display with mobile devices, the principles discussed with GRASP and DI.VI.CO are likely to be realizable in real-world applications in the very near future. With additional steps in regard to sensor technology for localization of the mobile devices in space, the interaction concepts are flexible enough to support the movement and control of tools for data exploration. As part of this work, this thesis contributes principles that show the benefits of this device combination for data analysis, adding to the development of natural user interfaces for information visualization. It is our goal as researchers to investigate these novel concepts to better understand the user's behavior and tools to guide future developments for productive use. As such, we focused on user behavior, general movement patterns, and workflows with the presented tools to increase our knowledge of user's needs and procedures that we aim to support with novel technologies and concepts.

In contrast to GRASP and DI.VI.CO, our exploration on BODYLENSES may seem further from actual application in real-world scenarios. However, BODYLENSES are an exploration into the possibilities of bringing data analysis tools closer to the body. They may well require further iteration and clarification for mappings before being applied to professional use. However, the general principles of this approach and their discussion are of benefit for many application cases using large vertical display: Specifically, the issues presented in this dissertation regarding implicit and explicit interaction, the use of relative and absolute movements for parameter adjustment, and aspects of forcing specific distances to the display are important principles that will need to be considered for a wide range of novel interface applications regarding large vertical displays.

8.3 Future Work

By discussing the limitations and scope of this thesis in the previous section, I already addressed aspects of future work. In the following, I will pick up and extend some of these arguments and will also elaborate on further aspects and principles that require future research.

Extensions Beyond Visualization This thesis focused mainly on graph visualization and extended the presented principles to further types of visualization in the process. As previously described, magic lenses and consequently the proposed principles can be applied to other magic lens functions and visualization techniques. Even more, the concept of magic lenses exists in principles outside of visualization. As a next step, it would be worthwhile to see how the presented techniques may serve as a basis for further lens development outside of graph visualization or even visualization.

Collaboration While the presented techniques were designed with multi-user scenarios in mind, further iterations of the work should be considered for a more extensive focus in co-located collaboration. We saw lenses designed to be very user specific and in contrast collaborative use of one lens when in closely coupled collaboration. While magic lenses already provide the territory-character for individual users, combining the presented on-display lenses together with the tangible lens approach in GRASP could provide a more enhanced separation of personal and public views. Future work should further investigate into how and when lenses could provide either a tool for active collaboration or separate territories for individual work.

Meeting Space for Data Exploration This dissertation contributed to the advancement of data analysis on wall-sized displays. However, it only considered a subset of interactions possible on and around large displays focusing on magic lenses as an important tool. Future work should widen this scope regarding the investigation of the entire process of data exploration in collaborative analysis spaces at display walls. In particular, it should be our goal to combine the presented variety of interaction styles in a more complex, feature-rich setting where actual users could meet to discuss their own domain-specific data.

Ad-hoc Combination of Devices Our current prototypes often require preparation of the used mobile devices. In particular, they depend on previously installed applications and an additional optical tracking system to identify the position of mobile devices in front of the wall-sized display. To enable ad-hoc, easy, and effortless use in novel display environments, future work has to improve integrated spatial tracking. Furthermore, simple connection, communication, and exchange between devices needs to be enabled both technologically but also in terms of interaction design to enable easy transfer of content between devices. For visualization this particularly requires consistent, responsive, and scalable visualization that adapt to the various display sizes and device capabilities.

Evolution of Mobile Devices An issue perceivable in our GRASP prototype is the decoupling of focus and context visualization due to attention switches. This coupling however is especially important for use of the mobile device as a magic lens. As the focus of the GRASP approach tended towards a personal, flexible toolbox, we used DI.VI.CO to eliminate this problem. However, the development and evolution of transparent mobile devices (e.g., our own work [Hin+14a; Hin+14b]) could take GRASP-like magic lens interaction to another level connecting the two devices by allowing seeing through the magic lens towards the content. As a result, aspects of augmented reality are relevant for enhancing this combination and supporting the connection between both views and should be investigated further.

Closing Remarks

To conclude, there is great potential for data exploration in novel display environments. These setups enable a new quality of data analysis and visualization by supporting exploration at large scale due to increased display space but also more flexible, natural interactions and support of collaboration with multiple users. This dissertation contributed to the development of natural user interfaces for information visualization by presenting a variety of interaction techniques for tool manipulation in visualization. This included investigations on user behavior and movement for data exploration at large vertical displays. The work focused on magic lenses as a specific, versatile analysis tool that supports diverse interaction tasks. At the same time, it investigated a variety of interaction modalities from touch to spatial interaction and body-centric interactions that contribute to future development of tools for visualization and data analysis.

Appendix

A

Contents

A.1 Appendix: Overview of Referenced Task Categories

E – Exploration Tasks

For general exploration task, this thesis refers to the following tasks from the categorization of Yi et al. [Yi+07]:

- E [SELECT]: mark something as interesting
- E [EXPLORE]: show me something else
- E [RECONFIGURE]: show me a different arrangement
- E [ENCODE]: show me a different representation
- E [ABSTRACT & ELABORATE]: show me more or less detail
- E [FILTER]: show me something conditionally
- E [CONNECT]: show me related items

G – Graph-specific Task Refinement

To refine the general exploration tasks with graph-related details, I use the task taxonomy by Lee et al. [Lee+06].

- Topolgy-based Tasks
  - G [ADJACENCY]: identify direct connections
  - G [ACCESSIBILITY]: identify adjacency of multiple levels
  - G [COMMON CONNECTION]: find common node adjencencies
  - G [CONNECTIVITY]: identify properties regarding adjacency, e.g., paths, clusters, bridges etc.
- G [ATTRIBUTE-BASED TASKS]: identify, filter, or compute values from node or edge attributes
- G [BROWSING TASKS]: tasks of following paths or revisiting nodes
- G [OVERVIEW TASKS]: tasks regarding a wider set of nodes and edges, estimating values, and properties

M – Graph Manipulation Tasks

For manipulation tasks, I apply a selected set of graph editing tasks as categorized by Gladisch et al. [Gla+15b].

- M [ADD NODES/EDGES]: creating or inserting nodes or edges
- M [ADD ATTRIBUTES]: insert an additional attribute dimension to nodes or edges
- M [UPDATE ATTRIBUTES]: change attribute value of nodes or edges
- M [DELETE NODES/EDGES]: remove nodes or edges from the graph
- M [DELETE ATTRIBUTES]: remove an attribute value or dimension from nodes or edges

A.2 Appendix for MultiLens Evaluations

A.2.1 Initial User Feedback Questionnaire (Translated)

This section contains a translated version of the questionnaire used for the initial user feedback discussing the MultiLens menu and gesture set.

Date:

General Information

Age: ____ Years

◯ male ◯ female

◯ dominant left hand ◯ dominant right hand

Please indicate how strongly you agree with the following claims.

	Never		Occasionally		Daily	Prefer not to say
I use mobile multi-touch devices (Smartphones, iPod, etc.)	◯	◯	◯	◯	◯	◯
I use larger multi-touch devices (tabletops, multi-touch display wall, etc.)	◯	◯	◯	◯	◯	◯
If yes, which devices?	____________________					

I use the following software	Never		Occasionally		Daily	Prefer not to say
Design tools (Photoshop, InDesign, Corel, etc.)	◯	◯	◯	◯	◯	◯
Graph Visualization Tools (Gephi, CGV, etc.)	◯	◯	◯	◯	◯	◯
Diagram Editors (z.B. Visio, UML-Editoren...)	◯	◯	◯	◯	◯	◯
If yes, which ones?	____________________					

General

	Disagree				Agree	
It was clearly visible how to move and scale the graph lenses.	◯	◯	◯	◯	◯	
I could reach my goals and answer the questions with the help of the graph lenses.	◯	◯	◯	◯	◯	
I could easily identify and change the parameters of the individual lens functions.	◯	◯	◯	◯	◯	
I could imagine the graph lenses in a real application to analyze graphs.	◯	◯	◯	◯	◯	

Radial Graph Lens Menu

	Disagree				Agree	
It was easy to invoke the radial graph lens menu.	◯	◯	◯	◯	◯	
It was easy to select lens functions using the radial menu.	◯	◯	◯	◯	◯	
It was easy to change lens function parameters in the radial menu.	◯	◯	◯	◯	◯	
It was clear how to use the lens menu.	◯	◯	◯	◯	◯	
I could easily answer the questions with the help of the menu.	◯	◯	◯	◯	◯	

Lens Gestures

	Disagree				Agree	
The gestures to trigger the lens functions were easy to under-stand.	◯	◯	◯	◯	◯	
It was easy to select lens functions using the gestures.	◯	◯	◯	◯	◯	
It was easy to change lens function parameters with the gestures.	◯	◯	◯	◯	◯	
I could easily answer the questions with the help of the gestures.	◯	◯	◯	◯	◯	
With the gestures, I could change lens functions and parameters quicker than with the radial menu.	◯	◯	◯	◯	◯	

Additional comments, feature wishes etc.

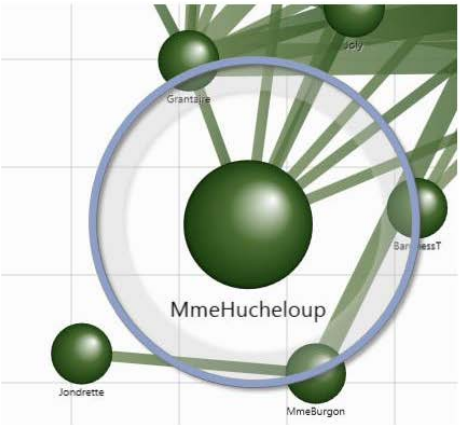
A.2.2 Comparative Evaluation

This section contains information on the first explanation of lens functions given to the users as well as an example set of process tasks (German), and the translated questionnaire.

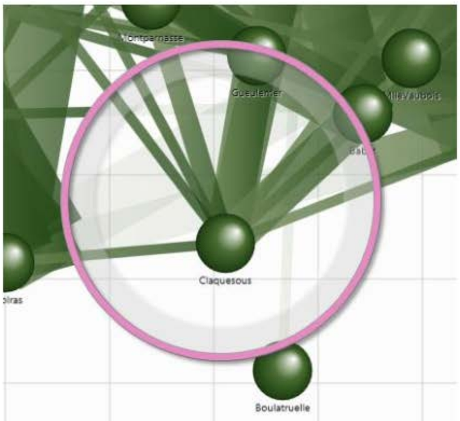
Lens Function Introduction Sheet

To make participants familiar with the lens functions before the introduction to the prototype, they were given an explanation of each function using the following paper sheet (translated from German) showing their effect.

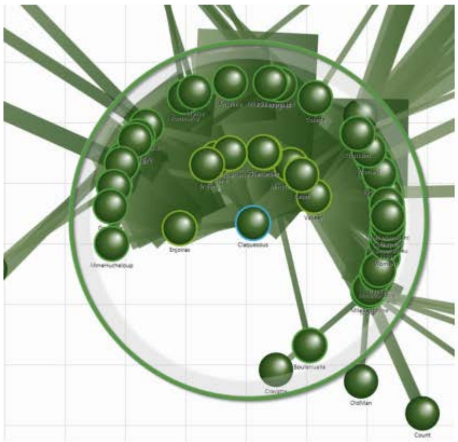
Fish-Eye: Magnifying Nodes



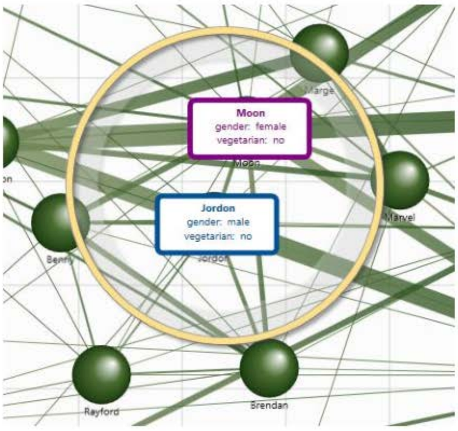
Local-Edge : Filtering Edges



BringNeighbors: Pulling-in Neighbors



DetailView: Showing detailed information



Set of Process Task Examples (Translated)

1. Move the lens onto the node highlighted in red.
2. Change the radius of the lens to 200 pixels.
3. Position the lens to also include the node hightlighted in blue.
4. Activate the lens function "fisheye".

1. Move the lens onto the node highlighted in red.
2. Activate the lens function "local edge" and
3. Change the function parameter "edge opacity" to 0.1

1. Activate the lens function "fisheye".
2. Change the function parameter "node zoom" to 2.0
3. Change the function parameter "repulsion" to 3.5 and
4. Move the lens onto the node highlighted in blue.

1. Change the radius of the lens to 230 pixels.
2. Activate the lens function "bring neighbors".
3. Change the function parameter "degree of neighborhood" to 2 and
4. Move the lens onto the node highlighted in blue.

1. Activate the lens functions "fisheye" and "local edge".
2. Change the function parameters "edge weight" to 3.0 and "edge opacity" to 0.2
3. Move the lens onto the node highlighted in blue.

1. Move each of the lenses onto one of the nodes highlighted in red.
2. Activate lens function "fisheye" for one lens and "detail view" for the other.
3. For the fisheye-lens, change the function parameter "node zoom" to 3.

Questionnaire (Translated)

Date: User-ID:

General Information

Age: ______ years

Gender: ◯ male ◯ female ◯ other

Dominant hand: ◯ left ◯ right

Experience

	Never		Occasionally		Daily	Prefer not to say
I use mobile multi-touch devices (Smartphones, iPod, etc.)	◯	◯	◯	◯	◯	◯
I use larger multi-touch devices (tabletops, multi-touch display wall, etc.)	◯	◯	◯	◯	◯	◯
If yes, which devices?	______					

	Never		Occasionally		Daily	Prefer not to say
I work with graph data, networks, or node-link diagrams.	◯	◯	◯	◯	◯	◯
I know the meaning of the term magic lenses.	◯	◯	◯	◯	◯	◯
If yes, in what context have you used or learned about magic lenses?	______					

for every condition:

Date: User-ID:

Questionnaire for condition: ___________________________

	Very low						Very high
How mentally demanding was the task?	○	○	○	○	○	○	○
How physically demanding was the task?	○	○	○	○	○	○	○

	Very low						Very high
How hurried or rushed was the pace of the task?	○	○	○	○	○	○	○
How successful were you in accomplishing what you were asked to do?	○	○	○	○	○	○	○

	Very little						Very much
How hard did you have to work to accomplish your level of performance?	○	○	○	○	○	○	○
How insecure, irritated, stressed (versus sure, satisfied, relaxed) were you?	○	○	○	○	○	○	○

	Disagree						Agree
It was easy to select lens functions.	○	○	○	○	○	○	○
It was easy to change lens function parameters.	○	○	○	○	○	○	○

Date: User-ID:

Final Questionnaire

	Touch with menu at display border	Touch with menu at lens	Mouse with menu at display border	Prefer not to answer
Which version was easiest to understand?	◯	◯	◯	◯
Which version did support you the best?	◯	◯	◯	◯
Which version did support you the least?	◯	◯	◯	◯
Which version was fastest for interacting with the lens?	◯	◯	◯	◯
Which version was most comfortable for interacting with the lens?	◯	◯	◯	◯
Which version could you imagine working with in everyday work?	◯	◯	◯	◯

What did you like?

____

____

____

What didn't you like?

____

____

____

Further comments.

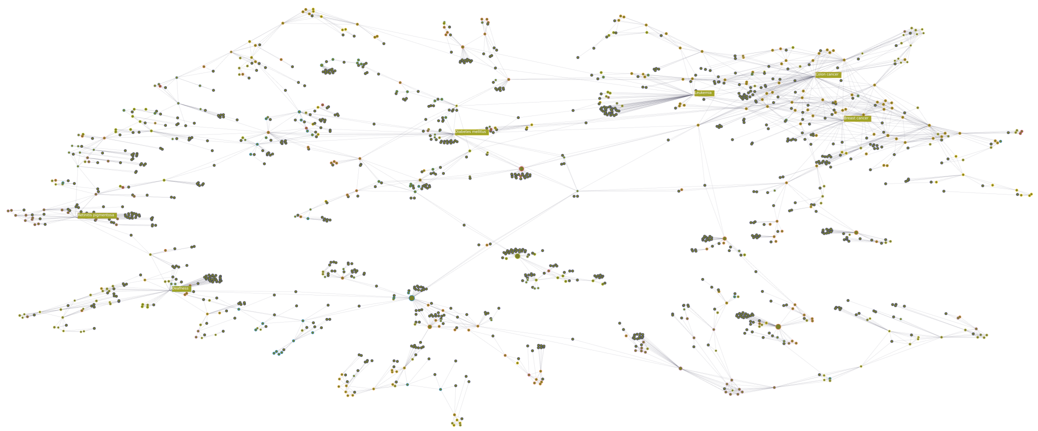
____

____

A.3 Appendix for GRASP Evaluation

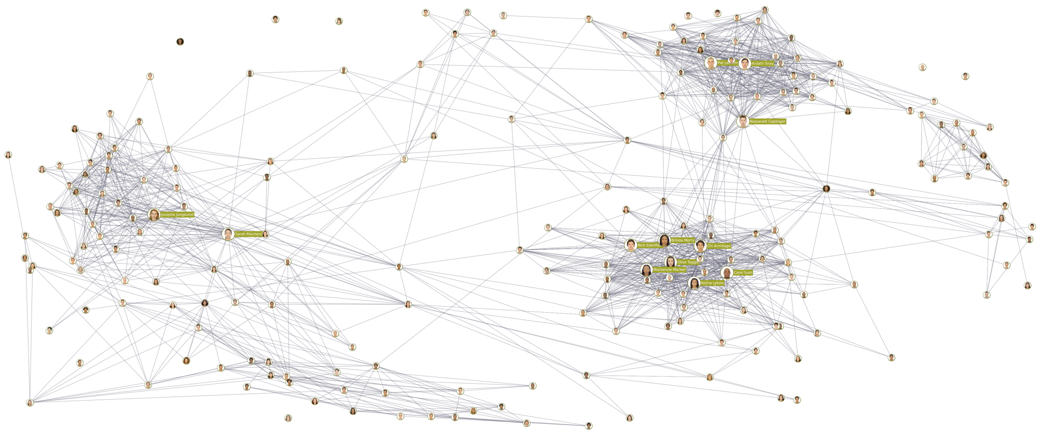
The following graphs were used with the GRASP prototype:

Human Disease Network



Data for this graph originated from the human disease network published by Goh et al. [Goh+07]. It contains 1419 nodes presenting genes and diseases/disorders (including their disease class) as well as 2738 edges describing gene connections.

Facebook Social Network



Data for this graph was extracted from the authors personal facebook account (state of 2014) and anonymized using random name generators for German, English, and Hispanic names (depending on facebook user group based on location) and extended with images from the Chicago Face Database [MCW15]. It contains 218 people and 1530 friend relations. Beside the full name, nodes contained information on the user's home location, age, gender (if given), associated facebook group, and a randomly assigned set of hobbies.

A.4 Appendix for DI.VI.CO Evaluation

A.4.1 DI.VI.CO Exploration Tasks (Translated)

Basic Exploration

Districts

1. Name the two districts in which most crimes are committed.
2. Are those two districts consistently the ones with most crimes up to 2016?
3. Is the order of these two districts the same for every day of the week?
4. What district has the least number of crimes on Thursdays?
5. In which month(s) does this district not have the lowest number of crimes?

Crime Types

1. Which crime type had a distinct increase in committed crimes in April 2015?
2. Which crime type displays all in all the most crimes?
3. How do these two crime types differ in regard to times of day crimes are committed?

Neighborhoods I

1. How many crimes are committed in each of the three neighborhoods with most crimes?
2. To which districts do these three neighborhoods belong to?
3. During which times of day do the number of crimes differ strongly in these three neighborhoods?
4. With what crime types are these neighborhoods especially affected?

Neighborhoods II

1. Which four neighborhoods with over 3000 crimes do not have robbery as the highest crime type?
2. What are the highest crime types for each of these neighborhoods?
3. To which district does the neighborhood 'INNER HARBOR' belong?
4. The neighborhood 'INNER HARBOR' has a strong increase in crime rate in the afternoon. Due to which crime type(s)?

Weapons

1. How many crimes were committed with the mentioned weapons each?
2. Did the number of crimes using firearms increase over time?
3. Which crime types are committed with firearms?
4. How do crime types committed with firearms differ in regard to time of day?
5. Which neighborhood with number of crimes higher than 1000 has firearms as the most often used weapon?

Years

1. Which year generally had the lowest amount of crimes?
2. Exactly how many crimes occurred in June of 2014?
3. Were there more crimes on Thursdays than on Mondays in 2014?
4. Which year had a distinct increase in crime at the end of April?

Thesis Verification Tasks

- The western part of Baltimore is more dangerous than the eastern part.
- Due to a snow storm there was a decrease of crimes in January 2016.
- Gun shootings occur predominantly at night.
- In neighborhoods with generally lower crime rates there are relatively more crimes involving firearms.

A.4.2 Questionnaires (Translated)

The following questions were presented as an electronic self-hosted LimeSurvey[1] questionnaire on separate computers for each participant. All questions were given in German and are translated here for the reader's convenience.

Demographic Information

1. How old are you?
2. What is your gender?
3. Which is your dominant hand?
4. Please give your body height in cm.
5. Do you have any physical handicaps, constraints, or disabilities...
   - ... regarding visual impairments (glasses or contact lenses)?
   - ... regarding hands or arms (pains or jitter)?
   - ... regarding standing or walking?
   - ... others?
6. What is your profession?
7. If you are part of this university: To which department do you belong?
8. From where do you know your team member?
9. Have you ever worked together with your team member?
10. In what way did you work together?

Experience

1. Please mark how often you use the following devices?
   (scale from 5-daily to 1-never + do not know)
   - Mobile touch devices (Smartphones, iPod, Tablet, etc.)?
   - Touch-enabled laptop or PC monitors?
   - Larger touch-enabled devices (interactive whiteboards, display walls, etc.)?

   - Laser pointer for pointing at slides during presentations?
   - Controller of a Nintendo Wii or similar for pointing in gaming?
   - TV remote control for pointing on the TV screen?
2. Have you ever used a mobile device (e.g., phone or tablet) to remote control another device (e.g., TV, drone) or invoke a function on another remote device?

[1] https://www.limesurvey.org/

3. Mark the principles you are familiar with and hence know the meaning of:
    - Information Visualization
    - Visual Analytics
    - None of the above
4. Which programs for visual data analysis have you used?
    - Never used any programm for visual data analysis.
    - Tableau
    - Spotfire
    - MS Excel
    - others
5. Which visualization techniques to visually present data are you familiar with?
    - Bar chart
    - Line chart
    - Scatter plot
    - Graph visualization, networks
    - Parallel coordinates plots
    - Tree map
    - Timeline
    - Star plot or radar plot
6. The concept of multiple coordinated views is familiar to me and I know its meaning.
7. Classify your experience with visual data analysis (scale from novice to expert)?

Training Questionnaire

The following questions were presented after each training condition (both touch and distant interaction).

1. How well do the following statements describe your opinion? [1 to 7 scale + do not know]
    - I felt secure in using the system.
    - It was simple to use this system
    - I was completely in control of the system at any time.
    - I could interact precisely with the system at any time.
    - After the training, I feel comfortable using the system.
2. How well do the following statements describe your opinion? [1 to 7 scale + do not know]
   It was easy...
    - to select small individual elements.
    - to select big individual elements.
    - to select multiple elements.
    - to access detailed information on individual elements.
    - to create and delete guides.
    - to move guides onto a specific position.
    - to create and delete lenses.
    - to move lenses onto a specific position.
    - to configure and switch lens functions.

Final Questionnaire

The following questionnaire was given after both exploration phases.

1. Evaluate the following aspects on a scale of 1 (low) to 10 (high).
    - How mentally demanding was the task?
    - How physically demanding was the task?
    - How successful were you in accomplishing what you were asked to do?
    - How hard did you have to work to accomplish you level of performance?

2. How well do the following statements apply to you? [1 to 7 scale + do not know]
    - I worked more with touch at the display wall than with the mobile device.
    - I had to walk a lot to answer the questions.
    - I often stood close to the display wall.
    - I often stood a few meters from the display wall.
    - Pointing with the device was easy.
    - Interaction on the display wall was easy.

    - Interaction with my team member was comfortable.
    - Because of the team work we could answer more questions that I would have been able working alone.
    - We had no conflicts while interacting.
    - We often stood closely together to get the results.
    - Activities between us were evenly balanced.

3. Which interaction modality (touch or remote) do you prefer for the following activities?
    - Single item selection
    - Multi item selection
    - Control of guides
    - Control of lenses

4. How much do the following statements describe your opinion?
    - I would not have needed the mobile device and would have liked to put it away.
    - This system does not have all the functions and capabilities I expect it to have.
    - It was hard to learn to use this system.
    - I did not have control over the system at all times.
    - I was very insecure and imprecise while using the system.

5. Open Questions
    - What did you like about the system, the interaction, or visualization?
    - What didn't you like about the system, the interaction, or visualization?
    - What features did you miss while interacting with the system?
    - Other comments or opinions.

A.4.3 Extension of GIAnT

To analyze the movement and interaction data collected within the DI.VI.CO study, we used GIAnT [ZD17], the Group Interaction Analysis Toolkit developed by Ulrich von Zadow and Raimund Dachselt in our research group. To apply our data and use this tool, we manipulated the existing visualization views and the GIAnT plugin itself. We further added specific views regarding device use to the visualization toolkit. The final analysis toolkit used for study evaluation can be seen in the following screenshot.

A Buttons to jump to specific DI.VI.CO study phases.

B Statistic view showing number of touches, distances walked etc. for selected time span per user (blue and yellow) and in sum.

C Device Touch presenting location of touch interaction on the devices (per user)

D App Events including beginning and end of task blocks as well as invoked functionality such as lenses and guides/aid lines.

E Movement of users (blue and yellow) over time where left-right position is in regard to wall-sized display and distance is width of line. Interactions are highlighted in white. Green and red lines indicate the start and end of DI.VI.CO study phases, respectively.

F Interactions over time per user and non-associated (in grey) where each user is encoded with two colors for either direct touch interaction at the display wall or remote interactions using the mobile device.

G Distance timeline showing each users' distance to the wall-sized display. White highlights indicate interaction with the display.

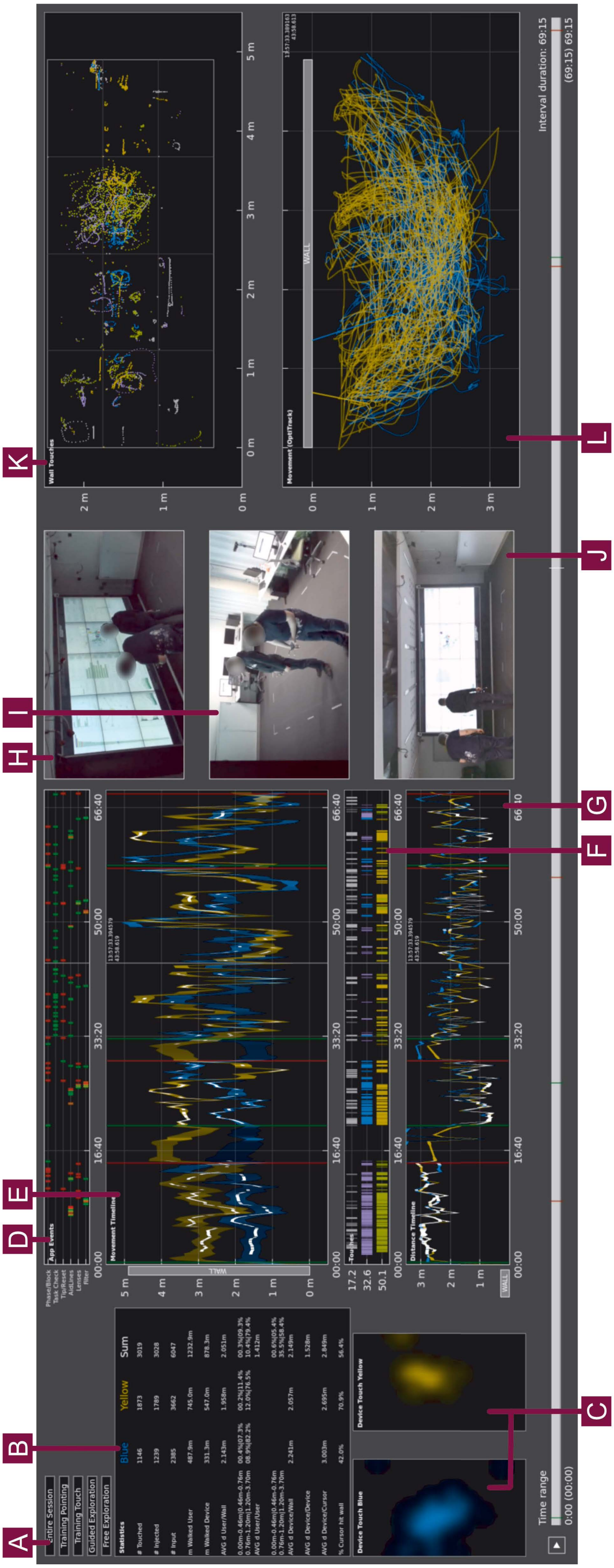
H CamBack – Camera with focus on the display wall showing interaction results.

I CamFront – Camera with focus on participants' interactions.

J KinectRGB – Kinect camera recording of users from behind.

K Wall touches showing all interactions and touch events on the wall-sized display encoded with two colors per user equal to F.

L Top view of movement in front of display wall showing either device positions (Optitrack) or body positions (Kinect) toggled by keyboard button.


A
B
C
D
E
F
G
H
I
J
K
L
Entire Session
Training Pointing
Training Touch
Guided Exploration
Free Exploration
Statistics
Blue
Yellow
Sum
App Events
Movement Timeline
Distance Timeline
Wall Touches
Movement (OptiTrack)
Device Touch Blue
Device Touch Yellow
Time range
Interval duration: 69:15

Bibliography

[ACP10] C. Appert, O. Chapuis, and E. Pietriga. „High-precision Magnification Lenses“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’10. Atlanta, Georgia, USA: ACM, 2010, pp. 273–282 (cit. on pp. 43, 47, 63, 64).

[ADS05] M. Ashmore, A. T. Duchowski, and G. Shoemaker. „Efficient Eye Pointing with a Fisheye Lens“. In: *Proc. of Graphics Interface*. GI ’05. Victoria, British Columbia: Canadian Human-Computer Communications Society, 2005, pp. 203–210 (cit. on pp. 69, 73, 154).

[AEN10] C. Andrews, A. Endert, and C. North. „Space to Think: Large High-resolution Displays for Sensemaking“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’10. Atlanta, Georgia, USA: ACM, 2010, pp. 55–64 (cit. on pp. 3, 20, 170).

[AES05] R. Amar, J. Eagan, and J. Stasko. „Low-Level Components of Analytic Activity in Information Visualization“. In: *Proc. IEEE Symposium on Information Visualization*. INFOVIS ’05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 15– (cit. on pp. 6, 35–37).

[Alm+12] R. A. de Almeida, C. Pillias, E. Pietriga, and P. Cubaud. „Looking Behind Bezels: French Windows for Wall Displays“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’12. Capri Island, Italy: ACM, 2012, pp. 124–131 (cit. on p. 27).

[Alv+14] J. Alvina, C. Appert, O. Chapuis, and E. Pietriga. „RouteLens: Easy Route Following for Map Applications“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’14. Como, Italy: ACM, 2014, pp. 125–128 (cit. on pp. 43, 47, 64).

[AN13] C. Andrews and C. North. „The Impact of Physical Navigation on Spatial Organization for Sensemaking“. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (Dec. 2013), pp. 2207–2216 (cit. on pp. 3, 20, 103, 140, 170).

[Aza+12] A. Azad, J. Ruiz, D. Vogel, M. Hancock, and E. Lank. „Territoriality and Behaviour on and Around Large Vertical Publicly-shared Displays“. In: *Proc. Designing Interactive Systems Conference*. DIS ’12. Newcastle Upon Tyne, United Kingdom: ACM, 2012, pp. 468–477 (cit. on pp. 21, 30, 146).

[Bad+16] S. K. Badam, F. Amini, N. Elmqvist, and P. Irani. „Supporting visual exploration for multiple users in large display environments“. In: *IEEE Conference on Visual Analytics Science and Technology (VAST)*. Oct. 2016, pp. 1–10 (cit. on pp. 69, 70, 181).

[Bak+02] C. A. H. Baker, M. S. T. Carpendale, P. Prusinkiewicz, and M. G. Surette. „GeneVis: Visualization Tools for Genetic Regulatory Network Dynamics“. In: *Proc. IEEE Conference on Visualization*. VIS ’02. Boston, Massachusetts: IEEE Computer Society, 2002, pp. 243–250 (cit. on pp. 54, 55, 63).

[Bar+12] D. Baričević, C. Lee, M. Turk, T. Höllerer, and D. A. Bowman. „A Hand-Held AR Magic Lens with User-Perspective Rendering“. In: *IEEE International Symposium on Mixed and Augmented Reality*. IEEE Computer Society, 2012, pp. 197–206 (cit. on pp. 68, 181).

[Bau+02] P. Baudisch, N. Good, V. Bellotti, and P. Schraedley. „Keeping Things in Context: A Comparative Evaluation of Focus Plus Context Screens, Overviews, and Zooming“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’02. Minneapolis, Minnesota, USA: ACM, 2002, pp. 259–266 (cit. on p. 43).

[BBR10] P. Baudisch, T. Becker, and F. Rudeck. „Lumino: Tangible Building Blocks Based on Glass Fiber Bundles“. In: *Proc. of ACM SIGGRAPH*. SIGGRAPH ’10. Los Angeles, California: ACM, 2010, 16:1–16:1 (cit. on p. 18).

[BC87] R. A. Becker and W. S. Cleveland. „Brushing Scatterplots“. In: *Technometrics* 29.2 (1987), pp. 127–142 (cit. on p. 128).

[BE16] S. K. Badam and N. Elmqvist. „Visfer: Camera-based visual data transfer for cross-device visualization“. In: *Information Visualization* (2016) (cit. on pp. 28, 29).

[Bec+17] F. Beck, M. Burch, S. Diehl, and D. Weiskopf. „A Taxonomy and Survey of Dynamic Graph Visualization“. In: *Computer Graphics Forum* 36.1 (2017), pp. 133–159 (cit. on pp. 32, 180).

[Ber+13] J. Bergstrom-Lehtovirta, T. Eklund, A. Jylhä, K. Kuikkaniemi, C. An, and G. Jacucci. „BubblesDial: Exploring Large Display Content Graphs on Small Devices“. In: *Proc. International Conference on Mobile and Ubiquitous Multimedia*. MUM ’13. Luleå, Sweden: ACM, 2013, 1:1–1:10 (cit. on pp. 17, 24).

[Ber83] J. Bertin. *Semiology of graphics: diagrams, networks, maps*. University of Wisconsin press, 1983 (cit. on p. 1).

[Bes+17] L. Besançon, P. Issartel, M. Ammi, and T. Isenberg. „Hybrid Tactile/Tangible Interaction for 3D Data Exploration“. In: *IEEE Transactions on Visualization and Computer Graphics* 23.1 (Jan. 2017), pp. 881–890 (cit. on p. 29).

[BFE15] S. K. Badam, E. Fischer, and N. Elmqvist. „Munin: A Peer-to-Peer Middleware for Ubiquitous Analytics and Visualization Spaces“. In: *IEEE Transactions on Visualization and Computer Graphics* 21.2 (2015), pp. 215–228 (cit. on pp. 19, 103).

[BH99] R. Balakrishnan and K. Hinckley. „The Role of Kinesthetic Reference Frames in Two-handed Input Performance“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’99. Asheville, North Carolina, USA: ACM, 1999, pp. 171–178 (cit. on pp. 17, 83).

[BHR14] S. Butscher, K. Hornbæk, and H. Reiterer. „SpaceFold and PhysicLenses: Simultaneous Multifocus Navigation on Touch Surfaces“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’14. Como, Italy: ACM, 2014, pp. 209–216 (cit. on pp. 46, 65, 73, 77, 162).

[BI12] A. Bezerianos and P. Isenberg. „Perception of Visual Variables on Tiled Wall-Sized Displays for Information Visualization Applications“. In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (Dec. 2012), pp. 2516–2525 (cit. on pp. 23, 27).

[Bie+93] E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose. „Toolglass and Magic Lenses: The See-through Interface“. In: *Proc. 20th Annual Conference on Computer Graphics and Interactive Techniques*. ACM, 1993, pp. 73–80 (cit. on pp. 2, 41, 44–46, 48, 49, 60, 62, 113, 143, 151, 180).

[Bie+94] E. A. Bier, M. C. Stone, K. Fishkin, W. Buxton, and T. Baudel. „A Taxonomy of See-through Tools“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’94. Boston, Massachusetts, USA: ACM, 1994, pp. 358–364 (cit. on pp. 41, 44, 49, 51, 60).

[BLC12] D. Baur, B. Lee, and S. Carpendale. „TouchWave: Kinetic Multi-touch Manipulation for Hierarchical Stacked Graphs“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’12. Cambridge, Massachusetts, USA: ACM, 2012, pp. 255–264 (cit. on pp. 23–25).

[BM13] M. Brehmer and T. Munzner. „A Multi-Level Typology of Abstract Visualization Tasks“. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (Dec. 2013), pp. 2376–2385 (cit. on pp. 2, 6, 35–37).

[BMG10] T. Ballendat, N. Marquardt, and S. Greenberg. „Proxemic Interaction: Designing for a Proximity and Orientation-aware Environment“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’10. Saarbrücken, Germany: ACM, 2010, pp. 121–130 (cit. on pp. 21, 145, 153).

[BNB07] R. Ball, C. North, and D. A. Bowman. „Move to Improve: Promoting Physical Navigation to Increase User Performance with Large Displays“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’07. San Jose, California, USA: ACM, 2007, pp. 191–200 (cit. on pp. 3, 5, 20, 97, 104, 140, 170).

[Bor+10] S. Boring, D. Baur, A. Butz, S. Gustafson, and P. Baudisch. „Touch Projector: Mobile Interaction Through Video“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’10. Atlanta, Georgia, USA: ACM, 2010, pp. 2287–2296 (cit. on pp. 19, 125).

[Bor+14] C. Bortolaso, M. Oskamp, G. Phillips, C. Gutwin, and T. N. Graham. „The Effect of View Techniques on Collaboration and Awareness in Tabletop Map-Based Tasks“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’14. Dresden, Germany: ACM, 2014, pp. 79–88 (cit. on pp. 65, 73).

[Bra+11] A. Bragdon, R. DeLine, K. Hinckley, and M. R. Morris. „Code Space: Touch + Air Gesture Hybrid Interactions for Supporting Developer Meetings“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’11. Kobe, Japan: ACM, 2011, pp. 212–221 (cit. on p. 31).

[BRL09] E. Bertini, M. Rigamonti, and D. Lalanne. „Extended Excentric Labeling“. In: *Proc. Eurographics / IEEE - VGTC conference on Visualization*. EuroVis’09. Berlin, Germany: The Eurographs Association, 2009, pp. 927–934 (cit. on pp. 48, 54, 56, 64).

[Bro+13] J. Brosz, M. A. Nacenta, R. Pusch, S. Carpendale, and C. Hurter. „Transmogrification: Causal Manipulation of Visualizations“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’13. St. Andrews, Scotland, United Kingdom: ACM, 2013, pp. 97–106 (cit. on pp. 48, 66, 73).

[Bru+14] F. Brudy, D. Ledo, S. Greenberg, and A. Butz. „Is Anyone Looking? Mitigating Shoulder Surfing on Public Displays Through Awareness and Protection“. In: *Proc. International Symposium on Pervasive Displays*. PerDis ’14. Copenhagen, Denmark: ACM, 2014, 1:1–1:6 (cit. on pp. 21, 160).

[Büs+14] W. Büschel, U. Kister, M. Frisch, and R. Dachselt. „T4 - Transparent and Translucent Tangibles on Tabletops“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’14. Como, Italy: ACM, 2014, pp. 81–88 (cit. on p. 102).

[Büs+17] W. Büschel, P. Reipschläger, R. Langner, and R. Dachselt. „Investigating the Use of Spatial Interaction for 3D Data Visualization on Mobile Devices“. In: *Proc. ACM International Conference on Interactive Surfaces and Spaces*. ISS ’17. Brighton, United Kingdom: ACM, Oct. 2017, pp. 62–71 (cit. on pp. 7, 68).

[Bux95] W. A. S. Buxton. „Human-computer Interaction“. In: ed. by R. M. Baecker, J. Grudin, W. A. S. Buxton, and S. Greenberg. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1995. Chap. Chunking and Phrasing and the Design of Human-computer Dialogues, pp. 494–499 (cit. on pp. 16, 83, 84).

[BWB06] H. Benko, A. D. Wilson, and P. Baudisch. „Precise Selection Techniques for Multi-touch Screens“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’06. Montréal, Québec, Canada: ACM, 2006, pp. 1263–1272 (cit. on pp. 17, 18, 64, 77).

[CB14] D. Chattopadhyay and D. Bolchini. „Touchless Circular Menus: Toward an Intuitive UI for Touchless Interactions with Large Displays“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’14. Como, Italy: ACM, 2014, pp. 33–40 (cit. on pp. 21, 22).

[CBF14] O. Chapuis, A. Bezerianos, and S. Frantzeskakis. „Smarties: An Input System for Wall Display Development“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: ACM, 2014, pp. 2763–2772 (cit. on pp. 20, 65, 73, 79).

[Cha+12] L. Chan, S. Müller, A. Roudaut, and P. Baudisch. „CapStones and ZebraWidgets: Sensing Stacks of Building Blocks, Dials and Sliders on Capacitive Touch Screens“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’12. Austin, Texas, USA: ACM, 2012, pp. 2189–2192 (cit. on p. 18).

[Chi06] L. Chittaro. „Visualizing information on mobile devices“. In: *Computer* 39.3 (2006), pp. 40–45 (cit. on pp. 17, 24, 103).

[Chu+14] H. Chung, C. North, J. Z. Self, S. Chu, and F. Quek. „VisPorter: Facilitating Information Sharing for Collaborative Sensemaking on Multiple Displays“. In: *Personal Ubiquitous Comput.* 18.5 (June 2014), pp. 1169–1186 (cit. on p. 20).

[CKB09] A. Cockburn, A. Karlson, and B. B. Bederson. „A Review of Overview+Detail, Zooming, and Focus+Context Interfaces“. In: *ACM Computing Surveys* 41.1 (Jan. 2009), 2:1–2:31 (cit. on pp. 43, 44).

[CKW09] G. Chin Jr., O. A. Kuchar, and K. E. Wolf. „Exploring the Analytical Processes of Intelligence Analysts“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’09. Boston, MA, USA: ACM, 2009, pp. 11–20 (cit. on p. 30).

[CLM12] K. Cheng, J. Li, and C. Müller-Tomfelde. „Supporting Interaction and Collaboration on Large Displays Using Tablet Devices“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’12. Capri Island, Italy: ACM, 2012, pp. 774–775 (cit. on pp. 3, 5, 19, 39, 40).

[CLP04] S. Carpendale, J. Ligh, and E. Pattison. „Achieving higher magnification in context“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’04. Santa Fe, NM, USA: ACM, 2004, pp. 71–80 (cit. on pp. 43, 48).

[CM01] M. S. T. Carpendale and C. Montagnese. „A framework for unifying presentation space“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’01. Orlando, Florida: ACM, 2001, pp. 61–70 (cit. on p. 55).

[CMS94] J. Christensen, J. Marks, and S. Shieber. „Placing text labels on maps and diagrams“. In: *Graphic Gems* 4 (1994), pp. 497–504 (cit. on p. 56).

[CMS99] S. Card, J. D. Mackinlay, and B. Shneiderman. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann, 1999 (cit. on pp. 22, 43, 45, 174).

[CRV12] G. Casiez, N. Roussel, and D. Vogel. „1 € Filter: A Simple Speed-based Low-pass Filter for Noisy Input in Interactive Systems“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’12. Austin, Texas, USA: ACM, 2012, pp. 2527–2530 (cit. on p. 135).

[Cyp+17] M. A. Cypko, J. Wojdziak, M. Stoehr, B. Kirchner, B. Preim, A. Dietz, H. U. Lemke, and S. Oeltze-Jafra. „Visual Verification of Cancer Staging for Therapy Decision Support". In: *Computer Graphics Forum* 36.3 (2017), pp. 109–120 (cit. on p. 2).

[Die05] R. Diestel. *Graph Theory*. Vol. 173. Graduate Texts in Mathematics. Berlin Heidelberg: Springer-Verlag, 2005 (cit. on pp. 1, 32, 34).

[Dör+12] M. Dörk, N. H. Riche, G. Ramos, and S. Dumais. „PivotPaths: Strolling through Faceted Information Spaces". In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (Dec. 2012), pp. 2709–2718 (cit. on p. 38).

[Dos+14] J. Dostal, U. Hinrichs, P. O. Kristensson, and A. Quigley. „SpiderEyes: Designing Attention- and Proximity-aware Collaborative Interfaces for Wall-sized Displays". In: *Proc. International Conference on Intelligent User Interfaces*. ACM, 2014, pp. 143–152 (cit. on pp. 20, 27, 28, 157, 162).

[Dou04] P. Dourish. *Where the action is: the foundations of embodied interaction*. MIT press, 2004 (cit. on p. 145).

[Dru+13] S. M. Drucker, D. Fisher, R. Sadana, J. Herron, and m. schraefel m.c. „TouchViz: A Case Study Comparing Two Interfaces for Data Analytics on Tablets". In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI '13. Paris, France: ACM, 2013, pp. 2301–2310 (cit. on pp. 22–26, 77).

[EBD05] G. Ellis, E. Bertini, and A. Dix. „The sampling lens: making sense of saturated visualisations". In: *Extended Abstracts on Human Factors in Computing Systems*. CHI EA '05. Portland, OR, USA: ACM, 2005, pp. 1351–1354 (cit. on pp. 52, 53, 64).

[Ebe+13] A. Ebert, C. Weber, D. Cernea, and S. Petsch. „TangibleRings: Nestable Circular Tangibles". In: *Extended Abstracts on Human Factors in Computing Systems*. CHI EA '13. Paris, France: ACM, 2013, pp. 1617–1622 (cit. on pp. 68, 73, 100, 101).

[ED06] G. Ellis and A. Dix. „Enabling Automatic Clutter Reduction in Parallel Coordinate Plots". In: *IEEE Transactions on Visualization and Computer Graphics* 12.5 (Sept. 2006), pp. 717–724 (cit. on p. 63).

[EDF11] N. Elmqvist, P. Dragicevic, and J.-D. Fekete. „Color Lens: Adaptive Color Scale Optimization for Visual Exploration". In: *IEEE Transactions on Visualization and Computer Graphics* 17.6 (June 2011), pp. 795–807 (cit. on pp. 2, 4, 52, 71, 182).

[Elh+15] I. Elhart, F. Scacchi, E. Niforatos, and M. Langheinrich. „ShadowTouch: A Multi-user Application Selection Interface for Interactive Public Displays". In: *Proc. International Symposium on Pervasive Displays*. PerDis '15. Saarbruecken, Germany: ACM, 2015, pp. 209–216 (cit. on pp. 21, 22, 151).

[Elm+11] N. Elmqvist, A. Vande Moere, H.-C. Jetter, D. Cernea, H. Reiterer, and T. J. Jankun-Kelly. „Fluid interaction for information visualization". In: *Information Visualization* 10.4 (Oct. 2011), pp. 327–340 (cit. on pp. 3, 16, 22, 35).

[End+11] A. Endert, C. Andrews, Y. H. Lee, and C. North. „Visual Encodings That Support Physical Navigation on Large Displays". In: *Proc. of Graphics Interface 2011*. GI '11. St. John's, Newfoundland, Canada: Canadian Human-Computer Communications Society, 2011, pp. 103–110 (cit. on p. 27).

[EWE09] M. Eissele, D. Weiskopf, and T. Ertl. „Interactive Context-Aware Visualization for Mobile Devices". In: *Proc. International Symposium on Smart Graphics (SG)*. Ed. by A. Butz, B. Fisher, M. Christie, A. Krüger, P. Olivier, and R. Therón. Springer Berlin Heidelberg, 2009, pp. 167–178 (cit. on pp. 17, 24).

[FD13] M. Frisch and R. Dachselt. „Visualizing Off-Screen Elements of Node-Link Diagrams". In: *Information Visualization* 12(2) (Apr. 2013), pp. 133–162 (cit. on pp. 38, 152).

[Fek+08] J.-D. Fekete, J. J. Wijk, J. T. Stasko, and C. North. „Information Visualization". In: ed. by A. Kerren, J. T. Stasko, J.-D. Fekete, and C. North. Berlin, Heidelberg: Springer-Verlag, 2008. Chap. The Value of Information Visualization, pp. 1–18 (cit. on pp. 1, 22, 181).

[FG98] A. Fuhrmann and E. Gröller. „Real-time Techniques for 3D Flow Visualization". In: *Proc. IEEE Conference on Visualization*. IEEE Computer Society, 1998, pp. 305–312 (cit. on p. 46).

[FHD09] M. Frisch, J. Heydekorn, and R. Dachselt. „Investigating Multi-touch and Pen Gestures for Diagram Editing on Interactive Surfaces". In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS '09. Banff, Alberta, Canada: ACM, 2009, pp. 149–156 (cit. on pp. 23, 25, 38).

[FHD10] M. Frisch, J. Heydekorn, and R. Dachselt. „Diagram Editing on Interactive Displays Using Multi-Touch and Pen Gestures". In: *Proc. 6th International Conference on Diagrammatic Representation and Inference*. Diagrams '10. Portland, Oregon: Springer, Aug. 2010, pp. 182–196 (cit. on pp. 25, 38, 39, 85, 112).

[FIB95] G. W. Fitzmaurice, H. Ishii, and W. A. S. Buxton. „Bricks: Laying the Foundations for Graspable User Interfaces". In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI '95. Denver, Colorado, USA: ACM, 1995, pp. 442–449 (cit. on p. 18).

[Fit93] G. W. Fitzmaurice. „Situated Information Spaces and Spatially Aware Palmtop Computers". In: *Communications of the ACM* 36.7 (July 1993), pp. 39–49 (cit. on p. 19).

[FLD11] M. Frisch, R. Langner, and R. Dachselt. „Neat: a set of flexible tools and gestures for layout tasks on interactive displays". In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS '11. Kobe, Japan: ACM, 2011, pp. 1–10 (cit. on pp. 77, 84).

[Fox98] D. Fox. „Composing Magic Lenses". In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI '98. Los Angeles, California, USA: ACM, 1998, pp. 519–525 (cit. on pp. 46, 62).

[FP99] J.-D. Fekete and C. Plaisant. „Excentric Labeling: Dynamic Neighborhood Labeling for Data Visualization". In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI '99. Pittsburgh, Pennsylvania, USA: ACM, 1999, pp. 512–519 (cit. on pp. 54–56).

[Fri+13] M. Frisch, U. Kister, W. Büschel, R. Langner, and R. Dachselt. „Translucent Tangibles on Tabletops: Exploring the Design Space". In: *Workshop on Blended Interaction at SIGCHI Conference on Human Factors in Computing System*. Paris, France, Apr. 2013.

[FS05] C. Forlines and C. Shen. „DTLens: Multi-user Tabletop Spatial Data Exploration". In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST '05. Seattle, WA, USA: ACM, 2005, pp. 119–122 (cit. on pp. 65, 73).

[Gas+11] 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". In: *IEEE Transactions on Visualization and Computer Graphics* 17.12 (Dec. 2011), pp. 2183–2192 (cit. on p. 48).

[GCR14] J. Gilliot, G. Casiez, and N. Roussel. „Direct and Indirect Multi-touch Interaction on a Wall Display". In: *Proc. 26th Conference on L'Interaction Homme-Machine*. IHM '14. Villeneuve d'Ascq, France: ACM, 2014, pp. 147–152 (cit. on p. 20).

[GFC05] M. Ghoniem, J.-D. Fekete, and P. Castagliola. „On the Readability of Graphs Using Node-link and Matrix-based Representations: A Controlled Experiment and Statistical Analysis". In: *Information Visualization* 4.2 (July 2005), pp. 114–135 (cit. on pp. 34, 35, 56, 112).

[Gla+14] S. Gladisch, H. Schumann, M. Ernst, G. Füllen, and C. Tominski. „Semi-Automatic Editing of Graphs with Customized Layouts“. In: *Computer Graphics Forum* 33.3 (June 2014), pp. 381–390 (cit. on pp. 54, 59, 71, 173).

[Gla+15a] S. Gladisch, U. Kister, C. Tominski, R. Dachselt, and H. Schumann. „Mapping Tasks to Interactions for Graph Exploration and Editing“. In: *Poster Program of the IEEE Conference on Information Visualization*. InfoVis’15. Chicago, USA: IEEE, Oct. 2015 (cit. on p. 38).

[Gla+15b] S. Gladisch, U. Kister, C. Tominski, R. Dachselt, and H. Schumann. „Mapping Tasks to Interactions for Graph Exploration and Graph Editing on Interactive Surfaces“. In: *Technical Report – arXiv:1504.07844* (Apr. 2015) (cit. on pp. 38, 104, 188).

[Gla+15c] S. Gladisch, H. Schumann, M. Loboschik, and C. Tominski. „Toward Using Matrix Visualizations for Graph Editing“. In: *Poster Program of the IEEE Conference on Information Visualization*. InfoVis’15. Chicago, USA: IEEE, Oct. 2015 (cit. on pp. 39, 113).

[Gle+11] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, and J. C. Roberts. „Visual Comparison for Information Visualization“. In: *Information Visualization* 10.4 (Oct. 2011), pp. 289–309 (cit. on p. 162).

[GM16] S. Gupta and M. J. McGuffin. „Multitouch Radial Menu Integrating Command Selection and Control of Arguments with Up to 4 Degrees of Freedom“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’16. Bari, Italy: ACM, 2016, pp. 256–263 (cit. on p. 83).

[Goh+07] K.-I. Goh, M. E. Cusick, D. Valle, B. Childs, M. Vidal, and A.-L. Barabási. „The human disease network“. In: *Proc. National Academy of Sciences* 104.21 (2007), pp. 8685–8690 (cit. on pp. 2, 32, 33, 85, 115, 165, 197).

[Gre+11] S. Greenberg, N. Marquardt, T. Ballendat, R. Diaz-Marino, and M. Wang. „Proxemic Interactions: The New Ubicomp?“ In: *interactions* 18.1 (Jan. 2011), pp. 42–50 (cit. on pp. 21, 106, 143, 145, 153, 155, 162).

[GST13] S. Gladisch, H. Schumann, and C. Tominski. „Navigation Recommendations for Exploring Hierarchical Graphs“. In: *International Symposium on Visual Computing*. ISVC ’13. Rethymnon, Crete, Greece: Springer Berlin Heidelberg, 2013, pp. 36–47 (cit. on p. 152).

[GT14] S. Gladisch and C. Tominski. „Toward Integrated Exploration and Manipulation of Data Attributes in Graphs“. In: *Poster Program of the IEEE Conference on Information Visualization*. InfoVis’14. Paris, France: IEEE, Oct. 2014 (cit. on pp. 37–39).

[Gus+08] S. Gustafson, P. Baudisch, C. Gutwin, and P. Irani. „Wedge: Clutter-free Visualization of Off-screen Locations“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’08. Florence, Italy: ACM, 2008, pp. 787–796 (cit. on p. 152).

[Hal+10] M. Haller, J. Leitner, T. Seifried, J. R. Wallace, S. D. Scott, C. Richter, P. Brandl, A. Gokcezade, and S. Hunter. „The NiCE Discussion Room: Integrating Paper and Digital Media to Support Co-Located Group Meetings“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’10. Atlanta, Georgia, USA: ACM, 2010, pp. 609–618 (cit. on p. 31).

[Hal90] E. T. Hall. *The Hidden Dimension*. A Doubleday anchor book. Anchor Books, 1990 (cit. on pp. 21, 30, 136, 138, 145, 153, 162).

[Haw+05] K. Hawkey, M. Kellar, D. Reilly, T. Whalen, and K. M. Inkpen. „The Proximity Factor: Impact of Distance on Co-located Collaboration“. In: *Proc. International ACM SIGGROUP Conference on Supporting Group Work*. GROUP ’05. Sanibel Island, Florida, USA: ACM, 2005, pp. 31–40 (cit. on pp. 21, 30, 178).

[HB13] C. Holz and P. Baudisch. „Fiberio: A Touchscreen That Senses Fingerprints“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’13. St. Andrews, Scotland, United Kingdom: ACM, 2013, pp. 41–50 (cit. on p. 160).

[HBP02] K. Hornbæk, B. B. Bederson, and C. Plaisant. „Navigation Patterns and Usability of Zoomable User Interfaces with and Without an Overview“. In: *ACM Transactions on Computer-Human Interaction* 9.4 (Dec. 2002), pp. 362–389 (cit. on p. 43).

[HD08] C. Harrison and A. K. Dey. „Lean and Zoom: Proximity-aware User Interface and Content Magnification“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’08. Florence, Italy: ACM, 2008, pp. 507–510 (cit. on pp. 21, 157).

[Hei+11] P. Heim, S. Lohmann, D. Tsendragchaa, and T. Ertl. „SemLens: visual analysis of semantic data with scatter plots and semantic lenses“. In: *Proc. International Conference on Semantic Systems*. I-Semantics ’11. Graz, Austria: ACM, 2011, pp. 175–178 (cit. on pp. 2, 48).

[HFM07] N. Henry, J.-D. Fekete, and M. J. McGuffin. „NodeTrix: a Hybrid Visualization of Social Networks“. In: *IEEE Transactions on Visualization and Computer Graphics* 13.6 (Nov. 2007), pp. 1302–1309 (cit. on pp. 34, 35, 54, 59, 112).

[Hin+10] K. Hinckley, K. Yatani, M. Pahud, N. Coddington, J. Rodenhouse, A. Wilson, H. Benko, and B. Buxton. „Pen + Touch = New Tools“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’10. New York, New York, USA: ACM, 2010, pp. 27–36 (cit. on p. 18).

[Hin+14a] J. D. Hincapié-Ramos, S. Roscher, W. Büschel, U. Kister, R. Dachselt, and P. Irani. „cAR: Contact Augmented Reality with Transparent-Display Mobile Devices“. In: *Proc. International Symposium on Pervasive Displays*. PerDis ’14. Copenhagen, Denmark: ACM, 2014, 80:80–80:85 (cit. on pp. 99, 184).

[Hin+14b] J. D. Hincapié-Ramos, S. Roscher, W. Büschel, U. Kister, R. Dachselt, and P. Irani. „tPad: Designing Transparent-display Mobile Interactions“. In: *Proc. Designing Interactive Systems Conference*. DIS ’14. Vancouver, BC, Canada: ACM, 2014, pp. 161–170 (cit. on p. 184).

[HKD17] T. Horak, U. Kister, and R. Dachselt. „Improving Value Driver Trees to Enhance Business Data Analysis“. In: *Poster Program of the IEEE Conference on Information Visualization*. InfoVis’17. Phoenix, Arizona, USA, Oct. 2017 (cit. on p. 32).

[Hor+18] T. Horak, S. K. Badam, N. Elmqvist, and R. Dachselt. „When David Meets Goliath: Combining Smartwatches with a Large Vertical Display for Visual Data Exploration“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’18. Montréal, Québec, Canada: ACM, Apr. 2018 (cit. on p. 29).

[HSS08] A. A. Hagberg, D. A. Schult, and P. J. Swart. „Exploring network structure, dynamics, and function using NetworkX“. In: *Proc. 7th Python in Science Conference (SciPy2008)*. Pasadena, CA USA, Aug. 2008, pp. 11–15 (cit. on p. 115).

[HSS15] S. Hadlak, H. Schumann, and H.-J. Schulz. „A Survey of Multi-faceted Graph Visualization“. In: *Eurographics Conference on Visualization (EuroVis) - STARs*. The Eurographics Association, 2015 (cit. on pp. 32, 34).

[HTE11] C. Hurter, A. Telea, and O. Ersoy. „MoleView: An Attribute and Structure-Based Semantic Lens for Large Element-Based Plots.“ In: *IEEE Transactions on Visualization and Computer Graphics* 17.12 (2011), pp. 2600–2609 (cit. on pp. 54, 57, 64).

[HVF13] S. Huron, R. Vuillemot, and J.-D. Fekete. „Visual Sedimentation“. In: *IEEE Transactions on Visualization and Computer Graphics* (Nov. 2013) (cit. on p. 115).

[HW04] F. van Ham and J. J. van Wijk. „Interactive Visualization of Small World Graphs“. In: *IEEE Symposium on Information Visualization*. INFOVIS ’04. 2004, pp. 199–206 (cit. on pp. 32, 54–56, 173).

[IF09] P. Isenberg and D. Fisher. „Collaborative Brushing and Linking for Co-located Visual Analytics of Document Collections“. In: *Computer Graphics Forum* 28.3 (June 2009), pp. 1031–1038 (cit. on p. 31).

[II13] P. Isenberg and T. Isenberg. „Visualization on Interactive Surfaces: A Research Overview“. In: *i-com* 12.3 (Nov. 2013), pp. 10–17 (cit. on p. 23).

[Ins85] A. Inselberg. „The plane with parallel coordinates“. In: *The Visual Computer* 1.2 (Aug. 1985), pp. 69–91 (cit. on p. 26).

[Ion+13] A. Ion, Y.-L. B. Chang, M. Haller, M. Hancock, and S. D. Scott. „Canyon: Providing Location Awareness of Multiple Moving Objects in a Detail View on Large Displays“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’13. Paris, France: ACM, 2013, pp. 3149–3158 (cit. on p. 48).

[Ise+09] P. Isenberg, S. Carpendale, A. Bezerianos, N. Henry, and J.-D. Fekete. „CoCoNutTrix: Collaborative Retrofitting for Information Visualization“. In: *IEEE Computer Graphics and Applications* 29.5 (Sept. 2009), pp. 44–57 (cit. on p. 31).

[Ise+10] P. Isenberg, U. Hinrichs, M. Hancock, and S. Carpendale. „Digital Tables for Collaborative Information Exploration“. In: *Tabletops—Horizontal Interactive Displays*. Ed. by C. Mueller-Tomfelde. Human-Computer Interaction Series. Springer Verlag, 2010, pp. 387–406 (cit. on p. 30).

[Ise+12] P. Isenberg, D. Fisher, S. A. Paul, M. Ringel Morris, K. Inkpen, and M. Czerwinski. „Co-Located Collaborative Visual Analytics Around a Tabletop Display“. In: *IEEE Transactions on Visualization and Computer Graphics* 18.5 (May 2012), pp. 689–702 (cit. on pp. 4, 5, 30, 159).

[Ise+13] P. Isenberg, P. Dragicevic, W. Willett, A. Bezerianos, and J.-D. Fekete. „Hybrid-Image Visualization for Large Viewing Environments“. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (Dec. 2013), pp. 2346–2355 (cit. on pp. 23, 27).

[Ise+17] P. Isenberg, F. Heimerl, S. Koch, T. Isenberg, P. Xu, C. D. Stolper, M. M. Sedlmair, J. Chen, T. Möller, and J. Stasko. „vispubdata.org: A Metadata Collection about IEEE Visualization (VIS) Publications“. In: *IEEE Transactions on Visualization and Computer Graphics* 23 (2017) (cit. on p. 165).

[Ish08] H. Ishii. „Tangible Bits: Beyond Pixels“. In: *Proc. International Conference on Tangible and Embedded Interaction*. TEI ’08. Bonn, Germany: ACM, 2008, pp. xv–xxv (cit. on p. 18).

[IU97] H. Ishii and B. Ullmer. „Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’97. Atlanta, Georgia, USA: ACM, 1997, pp. 234–241 (cit. on pp. 18, 99, 100).

[Jac+08] R. J. Jacob, A. Girouard, L. M. Hirshfield, M. S. Horn, O. Shaer, E. T. Solovey, and J. Zigelbaum. „Reality-based Interaction: A Framework for post-WIMP Interfaces“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’08. Florence, Italy: ACM, 2008, pp. 201–210 (cit. on pp. 3, 8, 15, 22, 25, 115, 145, 177).

[Jac90] R. J. K. Jacob. „What You Look at is What You Get: Eye Movement-based Interaction Techniques“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’90. Seattle, Washington, USA: ACM, 1990, pp. 11–18 (cit. on p. 69).

[Jak+13] M. R. Jakobsen, Y. Sahlemariam Haile, S. Knudsen, and K. Hornbaek. „Information Visualization and Proxemics: Design Opportunities and Empirical Findings“. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2386–2395 (cit. on pp. 23, 27, 28, 97, 103, 106, 108, 143, 145).

[Jak+15] M. R. Jakobsen, Y. Jansen, S. Boring, and K. Hornbæk. „Should I Stay or Should I Go? Selecting Between Touch and Mid-Air Gestures for Large-Display Interaction“. In: *International Conference on Human-computer Interaction*. INTERACT ’15. Bamberg, Germany: Springer International Publishing, 2015, pp. 455–473 (cit. on pp. 22, 27, 104, 125, 141).

[JD13] Y. Jansen and P. Dragicevic. „An Interaction Model for Visualizations Beyond The Desktop“. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (Dec. 2013), pp. 2396–2405 (cit. on pp. 23, 181).

[JDF12] Y. Jansen, P. Dragicevic, and J.-D. Fekete. „Tangible Remote Controllers for Wall-size Displays“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’12. Austin, Texas, USA: ACM, 2012, pp. 2865–2874 (cit. on pp. 19, 20, 23).

[JDK10] I. Jusufi, Y. Dingjie, and A. Kerren. „The Network Lens: Interactive Exploration of Multivariate Networks Using Visual Filtering“. In: *International Conference Information Visualisation*. IEEE Computer Society, 2010, pp. 35–42 (cit. on pp. 54, 56, 64).

[Jet+11] H.-C. Jetter, J. Gerken, M. Zöllner, H. Reiterer, and N. Milic-Frayling. „Materializing the Query with Facet-streams: A Hybrid Surface for Collaborative Search on Tabletops“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’11. Vancouver, BC, Canada: ACM, 2011, pp. 3013–3022 (cit. on pp. 99, 100).

[JH07] M. R. Jakobsen and K. Hornbæk. „Transient Visualizations“. In: *Proc. Australasian Conference on Computer-Human Interaction: Entertaining User Interfaces*. OZCHI ’07. Adelaide, Australia: ACM, 2007, pp. 69–76 (cit. on p. 44).

[JH14] M. R. Jakobsen and K. Hornbæk. „Up Close and Personal: Collaborative Work on a High-resolution Multitouch Wall Display“. In: *ACM Transactions on Computer-Human Interaction* 21.2 (Feb. 2014), 11:1–11:34 (cit. on pp. 4, 21, 30, 31, 136, 140, 146).

[JH15] M. R. Jakobsen and K. Hornbæk. „Is Moving Improving?: Some Effects of Locomotion in Wall-Display Interaction“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’15. Seoul, Republic of Korea: ACM, 2015, pp. 4169–4178 (cit. on p. 28).

[Jor+07] S. Jordà, G. Geiger, M. Alonso, and M. Kaltenbrunner. „The reacTable: Exploring the Synergy Between Live Music Performance and Tabletop Tangible Interfaces“. In: *Proc. International Conference on Tangible and Embedded Interaction*. TEI ’07. Baton Rouge, Louisiana: ACM, 2007, pp. 139–146 (cit. on pp. 17, 18).

[KAP11] D. P. Käser, M. Agrawala, and M. Pauly. „FingerGlass: efficient multiscale interaction on multitouch screens“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’11. Vancouver, BC, Canada: ACM, 2011, pp. 1601–1610 (cit. on pp. 46, 65, 73, 77).

[KB94] G. Kurtenbach and W. Buxton. „User Learning and Performance with Marking Menus“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’94. Boston, Massachusetts, USA: ACM, 1994, pp. 258–264 (cit. on p. 131).

[KE12] K. Kim and N. Elmqvist. „Embodied Lenses for Collaborative Visual Queries on Tabletop Displays“. In: *Information Visualization* 11.4 (Oct. 2012), pp. 319–338 (cit. on pp. 48, 67, 73, 100, 101).

[Ken90] A. Kendon. *Conducting interaction: Patterns of behavior in focused encounters*. Vol. 7. CUP Archive, 1990 (cit. on pp. 21, 162).

[Kha+04] A. Khan, G. Fitzmaurice, D. Almeida, N. Burtnyk, and G. Kurtenbach. „A Remote Control Interface for Large Displays“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’04. Santa Fe, NM, USA: ACM, 2004, pp. 127–136 (cit. on pp. 20, 65, 73).

[KHA11] K. Kin, B. Hartmann, and M. Agrawala. „Two-handed Marking Menus for Multitouch Devices“. In: *ACM Transactions on Computer-Human Interaction* 18.3 (Aug. 2011), 16:1–16:23 (cit. on p. 131).

[Kis+15] U. Kister, P. Reipschläger, F. Matulic, and R. Dachselt. „BodyLenses: Embodied Magic Lenses and Personal Territories for Wall Displays“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’15. Funchal, Portugal: ACM, 2015, pp. 117–126 (cit. on p. 70).

[Kis+17] U. Kister, K. Klamka, C. Tominski, and R. Dachselt. „GraSp: Combining Spatially-aware Mobile Devices and a Display Wall for Graph Visualization and Interaction“. In: *Computer Graphics Forum* 36.3 (2017), pp. 503–514.

[KKD16] U. Kister, K. Klamka, and R. Dachselt. „Supporting Graph Exploration Tasks on Display Walls using Spatially-aware Mobile Devices“. In: *IEEE Conference on Information Visualization*. Baltimore, MD, USA, Oct. 2016.

[Klu+12] S. Klum, P. Isenberg, R. Langner, J.-D. Fekete, and R. Dachselt. „Stackables: Combining Tangibles for Faceted Browsing“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’12. Capri Island, Italy: ACM, 2012, pp. 241–248 (cit. on pp. 18, 29).

[Koy+18] P. Koytek, C. Perin, J. Vermeulen, E. André, and S. Carpendale. „MyBrush: Brushing and Linking with Personal Agency“. In: *IEEE Transactions on Visualization and Computer Graphics* 24.1 (Jan. 2018), pp. 605–615 (cit. on p. 128).

[KPW07] M. Kumar, A. Paepcke, and T. Winograd. „EyePoint: Practical Pointing and Selection Using Gaze and Keyboard“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’07. San Jose, California, USA: ACM, 2007, pp. 421–430 (cit. on p. 69).

[KRD14] U. Kister, P. Reipschläger, and R. Dachselt. „Multi-Touch Manipulation of Magic Lenses for Information Visualization“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’14. Dresden, Germany: ACM, 2014, pp. 431–434.

[KRD16] U. Kister, P. Reipschläger, and R. Dachselt. „MultiLens: Fluent Interaction with Multi-Functional Multi-Touch Lenses for Information Visualization“. In: *Proc. ACM International Conference on Interactive Surfaces and Spaces*. ISS ’16. Niagara Falls, Ontario, Canada: ACM, 2016, pp. 139–148.

[Krü+13] 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: *Computer Graphics Forum* 32.3pt4 (2013), pp. 451–460 (cit. on p. 44).

[LAD14] R. Langner, A. Augsburg, and R. Dachselt. „CubeQuery: Tangible Interface for Creating and Manipulating Database Queries“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’14. Dresden, Germany: ACM, 2014, pp. 423–426 (cit. on pp. 29, 99, 100).

[LAM10] A. Lambert, D. Auber, and G. Melançon. „Living Flows: Enhanced Exploration of Edge-Bundled Graphs Based on GPU-Intensive Edge Rendering“. In: *International Conference Information Visualisation*. July 2010, pp. 523–530 (cit. on pp. 3, 54, 58, 173).

[Lan+11] T. von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. J. van Wijk, J. Fekete, and D. W. Fellner. „Visual Analysis of Large Graphs: State-of-the-Art and Future Research Challenges“. In: *Computer Graphics Forum* 30.6 (2011), pp. 1719–1749 (cit. on pp. 32, 34, 35, 56).

[Lan+15] R. Langner, U. Kister, C. Engel, T. Horak, and R. Dachselt. „Towards a Fluid Interaction Concept Bringing Star Plots to Interactive Displays“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’15. Madeira, Portugal: ACM, 2015, pp. 349–354 (cit. on pp. 24–26).

[Lan+16] R. Langner, U. von Zadow, T. Horak, A. Mitschick, and R. Dachselt. „Content Sharing Between Spatially-Aware Mobile Phones and Large Vertical Displays Supporting Collaborative Work“. In: *Collaboration Meets Interactive Spaces*. Springer International Publishing, Dec. 2016, pp. 75–96 (cit. on pp. 19, 20, 108, 125, 135).

[Lan00] C. Lankford. „Effective Eye-gaze Input into Windows“. In: *Proc. Symposium on Eye Tracking Research & Applications*. ETRA ’00. Palm Beach Gardens, Florida, USA: ACM, 2000, pp. 23–27 (cit. on pp. 69, 73).

[LBC04] J. Looser, M. Billinghurst, and A. Cockburn. „Through the Looking Glass: The Use of Lenses As an Interface Tool for Augmented Reality Interfaces“. In: *Proc. International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*. GRAPHITE ’04. Singapore: ACM, 2004, pp. 204–211 (cit. on pp. 68, 73).

[LBS85] S. Lee, W. Buxton, and K. C. Smith. „A Multi-touch Three Dimensional Touch-sensitive Tablet“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’85. San Francisco, California, USA: ACM, 1985, pp. 21–25 (cit. on p. 17).

[Led+15] D. Ledo, S. Greenberg, N. Marquardt, and S. Boring. „Proxemic-Aware Controls: Designing Remote Controls for Ubiquitous Computing Ecologies“. In: *Proc. International Conference on Human Computer Interaction with Mobile Devices & Services*. MobileHCI ’15. Copenhagen, Denmark: ACM, 2015, pp. 187–198 (cit. on p. 21).

[Lee+06] B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete, and N. Henry. „Task taxonomy for graph visualization“. In: *Proc. 2006 AVI Workshop on BEyond Time and Errors: Novel Evaluation Methods for Information Visualization*. BELIV ’06. Venice, Italy: ACM, 2006, pp. 1–5 (cit. on pp. 2, 6, 35–38, 104, 188).

[Lee+12] B. Lee, P. Isenberg, N. H. Riche, and S. Carpendale. „Beyond Mouse and Keyboard: Expanding Design Considerations for Information Visualization Interactions“. In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (2012), pp. 2689–2698 (cit. on pp. 22, 35, 181).

[Leh+11] A. Lehmann, H. Schumann, O. Staadt, and C. Tominski. „Physical Navigation to Support Graph Exploration on a Large High-Resolution Display“. In: *Advances in Visual Computing*. Vol. 6938. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2011, pp. 496–507 (cit. on pp. 39, 69, 70, 73, 154, 157).

[Lei+15] S.-w. Leigh, P. Schoessler, F. Heibeck, P. Maes, and H. Ishii. „THAW: Tangible Interaction with See-Through Augmentation for Smartphones on Computer Screens“. In: *Proc. International Conference on Tangible and Embedded Interaction*. TEI ’15. Stanford, California, USA: ACM, 2015, pp. 89–96 (cit. on p. 19).

[Lew95] J. R. Lewis. „IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use“. In: *International Journal of Human-Computer Interaction* 7.1 (1995), pp. 57–78 (cit. on p. 135).

[LHD18] R. Langner, T. Horak, and R. Dachselt. „VisTiles: Coordinating and Combining Colocated Mobile Devices for Visual Data Exploration“. In: *IEEE Transactions on Visualization and Computer Graphics* 24.1 (Jan. 2018), pp. 626–636 (cit. on pp. 20, 29, 100, 103).

[Liu+14] C. Liu, O. Chapuis, M. Beaudouin-Lafon, E. Lecolinet, and W. E. Mackay. „Effects of Display Size and Navigation Type on a Classification Task“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: ACM, 2014, pp. 4147–4156 (cit. on pp. 3, 26).

[Liu+16] C. Liu, O. Chapuis, M. Beaudouin-Lafon, and E. Lecolinet. „Shared Interaction on a Wall-Sized Display in a Data Manipulation Task“. In: *Proc. 2016 CHI Conference on Human Factors in Computing Systems*. CHI ’16. San Jose, California, USA: ACM, 2016, pp. 2075–2086 (cit. on pp. 4, 31).

[LK12] M. Li and L. Kobbelt. „Dynamic Tiling Display: Building an Interactive Display Surface Using Multiple Mobile Devices“. In: *Proc. International Conference on Mobile and Ubiquitous Multimedia*. MUM ’12. Ulm, Germany: ACM, 2012, 24:1–24:4 (cit. on p. 20).

[Mar+12] N. Marquardt, T. Ballendat, S. Boring, S. Greenberg, and K. Hinckley. „Gradual Engagement: Facilitating Information Exchange Between Digital Devices As a Function of Proximity“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’12. Cambridge, Massachusetts, USA: ACM, 2012, pp. 31–40 (cit. on p. 155).

[MBS97] M. R. Mine, F. P. Brooks Jr., and C. H. Sequin. „Moving Objects in Space: Exploiting Proprioception in Virtual-environment Interaction“. In: *Proc. Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH ’97. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1997, pp. 19–26 (cit. on p. 115).

[MCW15] D. S. Ma, J. Correll, and B. Wittenbrink. „The Chicago face database: A free stimulus set of faces and norming data“. In: *Behavior Research Methods* 47.4 (2015), pp. 1122–1135 (cit. on pp. 115, 197).

[MHG12] N. Marquardt, K. Hinckley, and S. Greenberg. „Cross-device interaction via micro-mobility and f-formations“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’12. Cambridge, Massachusetts, USA: ACM, 2012, pp. 13–22 (cit. on pp. 21, 30, 103, 162).

[MI94] C. L. MacKenzie and T. Iberall. *The Grasping Hand*. Advances in Psychology. Elsevier Science, 1994 (cit. on pp. 17, 18).

[MJ09] M. J. McGuffin and I. Jurisica. „Interaction Techniques for Selecting and Manipulating Subgraphs in Network Visualizations“. In: *IEEE Transactions on Visualization and Computer Graphics* 15.6 (Nov. 2009), pp. 937–944 (cit. on p. 38).

[MLG10] S. Malacria, E. Lecolinet, and Y. Guiard. „Clutch-free panning and integrated pan-zoom control on touch-sensitive surfaces: the cyclostar approach“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’10. Atlanta, Georgia, USA: ACM, 2010, pp. 2615–2624 (cit. on p. 83).

[Mor+14] R. Morawa, T. Horak, U. Kister, A. Mitschick, and R. Dachselt. „Combining Timeline and Graph Visualization“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’14. Dresden, Germany: ACM, 2014, pp. 345–350 (cit. on p. 26).

[Mos+09] T. Moscovich, F. Chevalier, N. Henry, E. Pietriga, and J.-D. Fekete. „Topology-aware navigation in large networks“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’09. Boston, MA, USA: ACM, 2009, pp. 2319–2328 (cit. on pp. 54, 58, 78).

[MRC91] J. D. Mackinlay, G. G. Robertson, and S. K. Card. „The Perspective Wall: Detail and Context Smoothly Integrated“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’91. New Orleans, Louisiana, USA: ACM, 1991, pp. 173–176 (cit. on p. 43).

[MS91] J. Marks and S. M. Shieber. „The Computational Complexity of Cartographic Label Placement“. In: *Harvard Computer Science Group Technical Report TR-05-91* (1991) (cit. on p. 56).

[MT14] N. Mahyar and M. Tory. „Supporting communication and coordination in collaborative sensemaking“. In: *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), pp. 1633–1642 (cit. on p. 31).

[Mül+10] J. Müller, F. Alt, D. Michelis, and A. Schmidt. „Requirements and Design Space for Interactive Public Displays“. In: *Proc. ACM International Conference on Multimedia*. MM ’10. Firenze, Italy: ACM, 2010, pp. 1285–1294 (cit. on p. 20).

[Mül+15] J. Müller, R. Rädle, H.-C. Jetter, and H. Reiterer. „An Experimental Comparison of Vertical and Horizontal Dynamic Peephole Navigation“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’15. Seoul, Republic of Korea: ACM, 2015, pp. 1523–1526 (cit. on p. 19).

[MW14] M. E. Mott and J. O. Wobbrock. „Beating the Bubble: Using Kinematic Triggering in the Bubble Lens for Acquiring Small, Dense Targets“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: ACM, 2014, pp. 733–742 (cit. on p. 52).

[Mye+02] B. A. Myers, R. Bhatnagar, J. Nichols, C. H. Peck, D. Kong, R. Miller, and A. C. Long. „Interacting at a Distance: Measuring the Performance of Laser Pointers and Other Devices“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’02. Minneapolis, Minnesota, USA: ACM, 2002, pp. 33–40 (cit. on p. 127).

[NBC06] T. Ni, D. A. Bowman, and J. Chen. „Increased Display Size and Resolution Improve Task Performance in Information-Rich Virtual Environments“. In: *Proc. of Graphics Interface*. GI ’06. Quebec, Canada: Canadian Information Processing Society, 2006, pp. 139–146 (cit. on p. 3).

[ND86] D. A. Norman and S. W. Draper. *User Centered System Design; New Perspectives on Human-Computer Interaction*. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1986 (cit. on p. 15).

[Nie94] J. Nielsen. „Heuristic evaluation“. In: *Usability inspection methods* 17.1 (1994), pp. 25–62 (cit. on pp. 16, 22).

[Nor10] D. A. Norman. „Natural User Interfaces Are Not Natural“. In: *interactions* 17.3 (May 2010), pp. 6–10 (cit. on pp. 15, 16).

[Olw+11] A. Olwal, O. Frykholm, K. Groth, and J. Moll. „Design and Evaluation of Interaction Technology for Medical Team Meetings“. In: *International Conference on Human-computer Interaction*. INTERACT ’11. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 505–522 (cit. on p. 19).

[OT12] T. Ohta and J. Tanaka. „Pinch: An Interface That Relates Applications on Multiple Touch-Screen by ’Pinching’ Gesture“. In: *Advances in Computer Entertainment*. Ed. by A. Nijholt, T. Romão, and D. Reidsma. Vol. 7624. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 320–335 (cit. on p. 20).

[PA08] E. Pietriga and C. Appert. „Sigma lenses: focus-context transitions combining space, time and translucence“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’08. Florence, Italy: ACM, 2008, pp. 1343–1352 (cit. on pp. 4, 44, 47).

[Pan+11] A. Panagiotidis, H. Bosch, S. Koch, and T. Ertl. „EdgeAnalyzer: Exploratory Analysis through Advanced Edge Interaction“. In: *2011 44th Hawaii International Conference on System Sciences*. Jan. 2011, pp. 1–10 (cit. on pp. 54, 57, 60).

[PBC17a] A. Prouzeau, A. Bezerianos, and O. Chapuis. „Evaluating Multi-User Selection for Exploring Graph Topology on Wall-Displays“. In: *IEEE Transactions on Visualization and Computer Graphics* 23.8 (Aug. 2017), pp. 1936–1951 (cit. on pp. 31, 39, 40, 139).

[PBC17b] A. Prouzeau, A. Bezerianos, and O. Chapuis. „Trade-offs Between a Vertical Shared Display and Two Desktops in a Collaborative Path-Finding Task“. In: *Proc. of Graphics Interface*. GI ’17. Edmonton, Alberta, Canada: CHCCS, May 2017 (cit. on pp. 31, 40).

[PD10] B. Preim and R. Dachselt. *Interaktive Systeme – Band 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung*. German. Springer Verlag, Nov. 2010 (cit. on p. 43).

[PD15] B. Preim and R. Dachselt. *Interaktive Systeme – Band 2: User Interface Engineering, 3D-Interaktion, Natural User Interfaces*. German. Vol. 2. Springer Verlag, Apr. 2015 (cit. on pp. 5, 15, 17).

[Pin+12] C. Pindat, E. Pietriga, O. Chapuis, and C. Puech. „JellyLens: content-aware adaptive lenses“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’12. Cambridge, Massachusetts, USA: ACM, 2012, pp. 261–270 (cit. on pp. 48, 63, 150).

[PNB09] S. M. Peck, C. North, and D. Bowman. „A multiscale interaction technique for large, high-resolution displays“. In: *IEEE Symposium on 3D User Interfaces*. 3DUI ’09. Lafayette, LA, USA, Mar. 2009, pp. 31–38 (cit. on p. 19).

[PPS14] J. Pretorius, H. C. Purchase, and J. T. Stasko. „Tasks for Multivariate Network Analysis“. In: *Multivariate Network Visualization: Dagstuhl Seminar #13201, Dagstuhl Castle, Germany, May 12-17, 2013, Revised Discussions*. Ed. by A. Kerren, H. C. Purchase, and M. O. Ward. Cham: Springer International Publishing, 2014, pp. 77–95 (cit. on pp. 2, 5, 6, 35, 36, 105).

[Pro+12] K. Probst, F. Perteneder, J. Leitner, M. Haller, A. Schrempf, and J. Glöckl. „Active Office: Towards an Activity-promoting Office Workplace Design“. In: *Extended Abstracts on Human Factors in Computing Systems*. CHI EA ’12. Austin, Texas, USA: ACM, 2012, pp. 2165–2170 (cit. on p. 177).

[Räd+13] R. Rädle, H.-C. Jetter, S. Butscher, and H. Reiterer. „The Effect of Egocentric Body Movements on Users’ Navigation Performance and Spatial Memory in Zoomable User Interfaces“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’13. St. Andrews, Scotland, United Kingdom: ACM, 2013, pp. 23–32 (cit. on p. 28).

[Räd+14] R. Rädle, H.-C. Jetter, N. Marquardt, H. Reiterer, and Y. Rogers. „HuddleLamp: Spatially-Aware Mobile Displays for Ad-hoc Around-the-Table Collaboration“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’14. Dresden, Germany: ACM, 2014, pp. 45–54 (cit. on p. 20).

[Raj+15] F. Rajabiyazdi, J. Walny, C. Mah, J. Brosz, and S. Carpendale. „Understanding Researchers’ Use of a Large, High-Resolution Display Across Disciplines“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’15. Madeira, Portugal: ACM, 2015, pp. 107–116 (cit. on p. 3).

[Ram+07] G. Ramos, A. Cockburn, R. Balakrishnan, and M. Beaudouin-Lafon. „Pointing Lenses: Facilitating Stylus Input Through Visual-and Motor-space Magnification“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’07. San Jose, California, USA: ACM, 2007, pp. 757–766 (cit. on pp. 66, 73).

[Red+15] K. Reda, A. E. Johnson, M. E. Papka, and J. Leigh. „Effects of Display Size and Resolution on User Behavior and Insight Acquisition in Visual Exploration“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’15. Seoul, Republic of Korea: ACM, 2015, pp. 2759–2768 (cit. on p. 3).

[Ric+12] N. H. Riche, T. Dwyer, B. Lee, and S. Carpendale. „Exploring the Design Space of Interactive Link Curvature in Network Diagrams“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’12. Capri Island, Italy: ACM, 2012, pp. 506–513 (cit. on pp. 25, 26, 38, 39).

[RK14] J. M. Rzeszotarski and A. Kittur. „Kinetica: Naturalistic Multi-touch Data Visualization“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: ACM, 2014, pp. 897–906 (cit. on pp. 24, 26, 66, 73, 115).

[RM93] G. G. Robertson and J. D. Mackinlay. „The Document Lens“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’93. Atlanta, Georgia, USA: ACM, 1993, pp. 101–108 (cit. on pp. 51, 52, 64).

[RNQ12a] U. Rashid, M. A. Nacenta, and A. Quigley. „Factors Influencing Visual Attention Switch in Multi-display User Interfaces: A Survey“. In: *Proc. International Symposium on Pervasive Displays*. PerDis '12. Porto, Portugal: ACM, 2012, 1:1–1:6 (cit. on pp. 20, 111).

[RNQ12b] U. Rashid, M. A. Nacenta, and A. Quigley. „The Cost of Display Switching: A Comparison of Mobile, Large Display and Hybrid UI Configurations“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI '12. Capri Island, Italy: ACM, 2012, pp. 99–106 (cit. on pp. 20, 122, 123).

[Rob+12] C. Roberts, B. Alper, J. K. Morin, and T. Höllerer. „Augmented textual data viewing in 3D visualizations using tablets“. In: *IEEE Symposium on 3D User Interfaces*. 3DUI '12. Mar. 2012, pp. 101–104 (cit. on p. 40).

[Rob+14] J. C. Roberts, P. D. Ritsos, S. K. Badam, D. Brodbeck, J. Kennedy, and N. Elmqvist. „Visualization beyond the Desktop–the Next Big Thing“. In: *IEEE Computer Graphics and Applications* 34.6 (Nov. 2014), pp. 26–34 (cit. on pp. 1, 22, 181).

[Rob07] J. C. Roberts. „State of the Art: Coordinated & Multiple Views in Exploratory Visualization“. In: *Proc. Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization*. IEEE Computer Society, 2007, pp. 61–71 (cit. on pp. 29, 125).

[RUO01] J. Rekimoto, B. Ullmer, and H. Oba. „DataTiles: A Modular Platform for Mixed Physical and Graphical Interactions“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI '01. Seattle, Washington, USA: ACM, 2001, pp. 269–276 (cit. on p. 18).

[SB92] M. Sarkar and M. H. Brown. „Graphical fisheye views of graphs“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI '92. Monterey, California, USA: ACM, 1992, pp. 83–91 (cit. on pp. 3, 54, 55, 173).

[SB94] M. Sarkar and M. H. Brown. „Graphical fisheye views“. In: *Communications of the ACM* 37.12 (Dec. 1994), pp. 73–83 (cit. on pp. 43, 54, 55).

[SBD12] M. Spindler, W. Büschel, and R. Dachselt. „Use Your Head: Tangible Windows for 3D Information Spaces in a Tabletop Environment“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS '12. Cambridge, Massachusetts, USA: ACM, 2012, pp. 245–254 (cit. on pp. 19, 70).

[SBR13] J. Seifert, A. Bayer, and E. Rukzio. „PointerPhone: Using Mobile Phones for Direct Pointing Interactions with Remote Displays“. In: *International Conference on Human-computer Interaction*. Ed. by P. Kotzé, G. Marsden, G. Lindgaard, J. Wesson, and M. Winckler. INTERACT '13. Cape Town, South Africa: Springer, 2013, pp. 18–35 (cit. on pp. 19, 125).

[SCG10] D. Schmidt, M. K. Chong, and H. Gellersen. „IdLenses: Dynamic Personal Areas on Shared Surfaces“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS '10. Saarbrücken, Germany: ACM, 2010, pp. 131–134 (cit. on pp. 48, 160, 162).

[Sch+10a] D. Schmidt, F. Chehimi, E. Rukzio, and H. Gellersen. „PhoneTouch: A Technique for Direct Phone Interaction on Surfaces“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST '10. New York, New York, USA: ACM, 2010, pp. 13–16 (cit. on pp. 19, 103).

[Sch+10b] S. Schmidt, M. A. Nacenta, R. Dachselt, and S. Carpendale. „A Set of Multi-touch Graph Interaction Techniques“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS '10. Saarbrücken, Germany: ACM, 2010, pp. 113–116 (cit. on pp. 3, 25, 26, 39, 54, 57, 66, 73, 77).

[Sch+13] H. J. Schulz, T. Nocke, M. Heitzler, and H. Schumann. „A Design Space of Visualization Tasks“. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (Dec. 2013), pp. 2366–2375 (cit. on pp. 6, 35, 36).

[SCI04] S. D. Scott, M. S. T. Carpendale, and K. M. Inkpen. „Territoriality in Collaborative Tabletop Workspaces“. In: *Proc. ACM Conference on Computer-Supported Cooperative Work*. CSCW ’04. Chicago, Illinois, USA: ACM, 2004, pp. 294–303 (cit. on pp. 4, 21, 30, 146).

[SD13] S. Stellmach and R. Dachselt. „Still Looking: Investigating Seamless Gaze-supported Selection, Positioning, and Manipulation of Distant Targets“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’13. Paris, France: ACM, 2013, pp. 285–294 (cit. on pp. 69, 73).

[SFB94] M. C. Stone, K. Fishkin, and E. A. Bier. „The movable filter as a user interface tool“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’94. Boston, Massachusetts, USA: ACM, 1994, pp. 306–312 (cit. on pp. 48, 49, 56, 60).

[SH10] O. Shaer and E. Hornecker. „Tangible User Interfaces: Past, Present, and Future Directions“. In: *Foundations and Trends in Human-Computer Interaction* 3.1–2 (Jan. 2010), pp. 1–137 (cit. on pp. 18, 100).

[Sha+17] L. Shao, A. Mahajan, T. Schreck, and D. J. Lehmann. „Interactive Regression Lens for Exploring Scatter Plots“. In: *Computer Graphics Forum* 36.3 (June 2017), pp. 157–166 (cit. on pp. 52, 53, 152, 182).

[Shn83] B. Shneiderman. „Direct Manipulation: A Step Beyond Programming Languages“. In: *Computer* 16.8 (Aug. 1983), pp. 57–69 (cit. on pp. 16, 17).

[Sho+10] G. Shoemaker, T. Tsukitani, Y. Kitamura, and K. S. Booth. „Body-centric Interaction Techniques for Very Large Wall Displays“. In: *Proc. Nordic Conference on Human-Computer Interaction*. NordiCHI ’10. Reykjavik, Iceland: ACM, 2010, pp. 463–472 (cit. on pp. 21, 22, 143, 145, 151, 156, 162).

[SMD12] M. Spindler, M. Martsch, and R. Dachselt. „Going Beyond the Surface: Studying Multi-Layer Interaction Above the Tabletop“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’12. Austin, TX, USA: ACM, May 2012, pp. 1277–1286 (cit. on p. 102).

[Sör+11] G. Sörös, H. Seichter, P. Rautek, and E. Gröller. „Augmented Visualization with Natural Feature Tracking“. In: *Proc. International Conference on Mobile and Ubiquitous Multimedia*. MUM ’11. ACM, 2011, pp. 4–12 (cit. on pp. 67, 73).

[Spe14] R. Spence. *Information Visualization: An Introduction*. Cham: Springer International Publishing, 2014 (cit. on pp. 1, 22, 181).

[Spi+10] M. Spindler, C. Tominski, H. Schumann, and R. Dachselt. „Tangible Views for Information Visualization“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’10. Saarbrücken, Germany: ACM, 2010, pp. 157–166 (cit. on pp. 18, 28, 54, 56, 67, 71, 73, 100, 102, 104, 108, 111).

[Spi+14a] M. Spindler, W. Büschel, C. Winkler, and R. Dachselt. „Tangible Displays for the Masses: Spatial Interaction with Handheld Displays by Using Consumer Depth Cameras“. In: *Personal Ubiquitous Comput.* 18.5 (June 2014), pp. 1213–1225 (cit. on pp. 67, 102).

[Spi+14b] M. Spindler, M. Schuessler, M. Martsch, and R. Dachselt. „Pinch-Drag-Flick vs. Spatial Input: Rethinking Zoom & Pan on Mobile Displays“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Canada: ACM, Apr. 2014, pp. 1113–1122 (cit. on p. 19).

[SS14] R. Sadana and J. Stasko. „Designing and Implementing an Interactive Scatterplot Visualization for a Tablet Computer“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’14. Como, Italy: ACM, 2014, pp. 265–272 (cit. on pp. 23–26, 65, 73, 77).

[SS16a] R. Sadana and J. Stasko. „Designing multiple coordinated visualizations for tablets“. In: *Computer Graphics Forum* 35.3 (2016), pp. 261–270 (cit. on pp. 24, 26).

[SS16b] R. Sadana and J. Stasko. „Expanding Selection for Information Visualization Systems on Tablet Devices“. In: *Proc. ACM International Conference on Interactive Surfaces and Spaces*. ISS ’16. Niagara Falls, Ontario, Canada: ACM, 2016, pp. 149–158 (cit. on pp. 23, 25, 26).

[SS17] A. Srinivasan and J. T. Stasko. „Natural Language Interfaces for Data Analysis with Visualization: Considering What Has and Could Be Asked“. In: *Eurographics Conference on Visualization - Short Papers*. EuroVis ’17. The Eurographics Association, 2017 (cit. on p. 7).

[SSD09] M. Spindler, S. Stellmach, and R. Dachselt. „PaperLens: Advanced Magic Lens Interaction Above the Tabletop“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’09. Banff, Alberta, Canada: ACM, 2009, pp. 69–76 (cit. on pp. 19, 23, 48, 66, 67, 73, 100, 102, 104).

[STB07] G. Shoemaker, A. Tang, and K. S. Booth. „Shadow Reaching: A New Perspective on Interaction for Large Displays“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’07. Newport, Rhode Island, USA: ACM, 2007, pp. 53–56 (cit. on pp. 21, 149, 160).

[Ste+11] S. Stellmach, S. Stober, A. Nürnberger, and R. Dachselt. „Designing Gaze-supported Multimodal Interactions for the Exploration of Large Image Collections“. In: *Proc. 1st Conference on Novel Gaze-Controlled Applications*. ACM, 2011, 1:1–1:8 (cit. on pp. 7, 69).

[Tan+06] A. Tang, M. Tory, B. Po, P. Neumann, and S. Carpendale. „Collaborative Coupling over Tabletop Displays“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’06. Montréal, Québec, Canada: ACM, 2006, pp. 1181–1190 (cit. on pp. 21, 30, 137, 138).

[TAS09] C. Tominski, J. Abello, and H. Schumann. „CGV – An Interactive Graph Visualization System“. In: *Computers & Graphics* 33.6 (2009), pp. 660–678 (cit. on pp. 38, 51, 54, 58, 60, 71).

[TBJ15] T. Tsandilas, A. Bezerianos, and T. Jacob. „SketchSliders: Sketching Widgets for Visual Exploration on Wall Displays“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’15. Seoul, Republic of Korea: ACM, 2015, pp. 3255–3264 (cit. on pp. 20, 122).

[TFJ12] C. Tominski, C. Forsell, and J. Johansson. „Interaction Support for Visual Comparison Inspired by Natural Behavior“. In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (Dec. 2012), pp. 2719–2728 (cit. on p. 162).

[TFS08] C. Thiede, G. Fuchs, and H. Schumann. „Smart Lenses“. In: *Smart Graphics*. Vol. 5166. SG ’08. Rennes, France: Springer-Verlag, 2008, pp. 178–189 (cit. on pp. 46, 61).

[TIC09] M. Tobiasz, P. Isenberg, and S. Carpendale. „Lark: Coordinating Co-located Collaboration with Information Visualization“. In: *IEEE Transactions on Visualization and Computer Graphics* 15.6 (Nov. 2009), pp. 1065–1072 (cit. on p. 30).

[Tom+06] C. Tominski, J. Abello, F. van Ham, and H. Schumann. „Fisheye Tree Views and Lenses for Graph Visualization“. In: *International Conference Information Visualisation*. IV ’06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 17–24 (cit. on pp. 43, 44, 46, 51, 54–58, 60, 61, 63, 111, 165, 173).

[Tom+12] C. Tominski, H. Schumann, G. Andrienko, and N. Andrienko. „Stacking-Based Visualization of Trajectory Attribute Data". In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (Dec. 2012), pp. 2565–2574 (cit. on pp. 46, 52, 53).

[Tom+14] C. Tominski, S. Gladisch, U. Kister, R. Dachselt, and H. Schumann. „A Survey on Interactive Lenses in Visualization". In: *Eurographics Conference on Visualization (EuroVis) - STARs*. The Eurographics Association, 2014.

[Tom+17] C. Tominski, S. Gladisch, U. Kister, R. Dachselt, and H. Schumann. „Interactive Lenses for Visualization: An Extended Survey". In: *Computer Graphics Forum* 36.6 (2017), pp. 173–200 (cit. on pp. 2, 41, 45, 49, 50, 89, 143, 173).

[Tom15] C. Tominski. *Interaction for Visualization*. Vol. 3. Synthesis Lectures on Visualization. San Rafael, CA: Morgan & Claypool, 2015 (cit. on pp. 22, 23, 35, 181).

[UČK12] J. Ukrop, Z. Čıková, and P. Kapec. „Visual Access to Graph Content Using Magic Lenses and Filtering". In: *Proc. 28th Spring Conference on Computer Graphics*. SCCG '12. Budmerice, Slovakia: ACM, 2012, pp. 23–30 (cit. on pp. 54, 56).

[UI97] B. Ullmer and H. Ishii. „The metaDESK: Models and Prototypes for Tangible User Interfaces". In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST '97. Banff, Alberta, Canada: ACM, 1997, pp. 223–232 (cit. on pp. 18, 101).

[UIJ03] B. Ullmer, H. Ishii, and R. J. K. Jacob. „Tangible query interfaces: Physically constrained tokens for manipulating database queries". In: *International Conference on Human-computer Interaction*. INTERACT '03. 2003, pp. 279–286 (cit. on p. 100).

[UIJ05] B. Ullmer, H. Ishii, and R. J. K. Jacob. „Token+Constraint Systems for Tangible Interaction with Digital Information". In: *ACM Transactions on Computer-Human Interaction* 12.1 (Mar. 2005), pp. 81–118 (cit. on p. 18).

[VB04] D. Vogel and R. Balakrishnan. „Interactive Public Ambient Displays: Transitioning from Implicit to Explicit, Public to Personal, Interaction with Multiple Users". In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST '04. Santa Fe, NM, USA: ACM, 2004, pp. 137–146 (cit. on pp. 20, 21, 39, 155).

[VBW17] C. Vehlow, F. Beck, and D. Weiskopf. „Visualizing Group Structures in Graphs: A Survey". In: *Computer Graphics Forum* 36.6 (2017), pp. 201–225 (cit. on p. 32).

[Voi+09] 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: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS '09. Banff, Alberta, Canada: ACM, 2009, pp. 109–116 (cit. on pp. 17, 19, 23, 64, 67, 73, 77, 104).

[VPF06] E. R. A. Valiati, M. S. Pimenta, and C. M. D. S. Freitas. „A Taxonomy of Tasks for Guiding the Evaluation of Multidimensional Visualizations". In: *Proc. 2006 AVI Workshop on BEyond Time and Errors: Novel Evaluation Methods for Information Visualization*. BELIV '06. Venice, Italy: ACM, 2006, pp. 1–6 (cit. on pp. 1, 6, 35–37).

[Wal+06] M. Waldner, J. Hauber, J. Zauner, M. Haller, and M. Billinghurst. „Tangible Tiles: Design and Evaluation of a Tangible User Interface in a Collaborative Tabletop Setup". In: *Proc. Australian Computer-Human Interaction Conference*. OZCHI '06. Sydney, Australia: ACM, 2006, pp. 151–158 (cit. on pp. 18, 23).

[Wal+12] J. Walny, B. Lee, P. Johns, N. Henry Riche, and S. Carpendale. „Understanding Pen and Touch Interaction for Data Exploration on Interactive Whiteboards". In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (Dec. 2012), pp. 2779–2788 (cit. on p. 23).

[Wat06] M. Wattenberg. „Visual Exploration of Multivariate Graphs". In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI '06. Montréal, Québec, Canada: ACM, 2006, pp. 811–819 (cit. on p. 38).

[WCG03] N. Wong, S. Carpendale, and S. Greenberg. „EdgeLens: an interactive method for managing edge congestion in graphs“. In: *IEEE Conference on Information Visualization*. INFOVIS’03. Seattle, Washington: IEEE Computer Society, 2003, pp. 51–58 (cit. on pp. 2, 4, 54, 56, 57, 63).

[Wei91] M. Weiser. „The Computer for the 21st Century“. In: *Scientific american* 265.3 (1991), pp. 94–105 (cit. on pp. 16, 145).

[WLI14] W. Willett, Q. Lan, and P. Isenberg. „Eliciting Multi-touch Selection Gestures for Interactive Data Graphics“. In: *Eurographics Conference on Visualization - Short Papers*. The Eurographics Association, 2014 (cit. on pp. 23, 26).

[Woź+14] P. Woźniak, L. Lischke, B. Schmidt, S. Zhao, and M. Fjeld. „Thaddeus: A Dual Device Interaction Space for Exploring Information Visualisation“. In: *Proc. Nordic Conference on Human-Computer Interaction*. NordiCHI ’14. Helsinki, Finland: ACM, 2014, pp. 41–50 (cit. on pp. 29, 125).

[WW11] D. Wigdor and D. Wixon. *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011 (cit. on pp. 5, 15–17, 22).

[Wyb+14] M. Wybrow, N. Elmqvist, J.-D. Fekete, T. von Landesberger, J. J. van Wijk, and B. Zimmer. „Interaction in the Visualization of Multivariate Networks“. In: *Multivariate Network Visualization: Dagstuhl Seminar #13201, Dagstuhl Castle, Germany, May 12-17, 2013, Revised Discussions*. Ed. by A. Kerren, H. C. Purchase, and M. O. Ward. Cham: Springer International Publishing, 2014, pp. 97–125 (cit. on pp. 2, 105).

[Yee03] K.-P. Yee. „Peephole Displays: Pen Interaction on Spatially Aware Handheld Computers“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’03. Ft. Lauderdale, Florida, USA: ACM, 2003, pp. 1–8 (cit. on p. 19).

[Yi+07] J. S. Yi, Y. a. Kang, J. Stasko, and J. Jacko. „Toward a Deeper Understanding of the Role of Interaction in Information Visualization“. In: *IEEE Transactions on Visualization and Computer Graphics* 13.6 (Nov. 2007), pp. 1224–1231 (cit. on pp. 2, 5, 6, 35–38, 50, 104, 188).

[Zad+14] U. von Zadow, W. Büschel, R. Langner, and R. Dachselt. „SleeD: Using a Sleeve Display to Interact with Touch-sensitive Display Walls“. In: *Proc. ACM International Conference on Interactive Tabletops and Surfaces*. ITS ’14. Dresden, Germany: ACM, 2014, pp. 129–138 (cit. on pp. 19, 20, 23, 28, 103, 116, 125, 131).

[Zad+16] U. von Zadow, P. Reipschläger, D. Bösel, A. Sellent, and R. Dachselt. „YouTouch! Low-Cost User Identification at an Interactive Display Wall“. In: *Proc. International Working Conference on Advanced Visual Interfaces*. AVI ’16. Bari, Italy: ACM, 2016, pp. 144–151 (cit. on p. 160).

[ZD17] U. von Zadow and R. Dachselt. „GIAnT: Visualizing Group Interaction at Large Wall Displays“. In: *Proc. SIGCHI Conference on Human Factors in Computing Systems*. CHI ’17. Denver, Colorado, USA: ACM, 2017, pp. 2639–2647 (cit. on pp. 30, 135, 202).

[ZDS12] U. von Zadow, F. Daiber, and J. Schöning. „GeoLens: Multi-User Interaction with Rich Geographic Information“. In: *Dexis - Workshop on Data Exploration for Interactive Surfaces at ITS* (2012), p. 16 (cit. on pp. 48, 80).

[ZG13] O. Zuckerman and A. Gal-Oz. „To TUI or Not to TUI: Evaluating Performance and Preference in Tangible vs. Graphical User Interfaces“. In: *International Journal of Human-Computer Studies* 71.7-8 (July 2013), pp. 803–820 (cit. on p. 18).

[Zha+13] Y. Zhang, T. Han, Z. Ren, N. Umetani, X. Tong, Y. Liu, T. Shiratori, and X. Cao. „BodyAvatar: Creating Freeform 3D Avatars Using First-person Body Gestures“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’13. St. Andrews, Scotland, United Kingdom: ACM, 2013, pp. 387–396 (cit. on p. 151).

[Zil+14] J. Zillner, C. Rhemann, S. Izadi, and M. Haller. „3D-board: A Whole-body Remote Collaborative Whiteboard“. In: *Proc. ACM Symposium on User Interface Software and Technology*. UIST ’14. Honolulu, Hawaii, USA: ACM, 2014, pp. 471–479 (cit. on p. 160).

Web References

[@BF11] BUT Dubai and Firmavision. *Interactive Video Wall*. 2011. URL: `https://youtu.be/b2kqRv1WdEM` (visited on Oct. 6, 2017) (cit. on pp. 145, 154).

[@Blo07] V. Blomqvist. *pymunk – 2d physics library for python*. 2007. URL: `www.pymunk.org` (visited on Nov. 17, 2017) (cit. on p. 115).

[@Bro13] Brother System. *Vitrine Interactive "Les Rives de l’Orne"*. 2013. URL: `https://youtu.be/KLOB-T1mgdY` (visited on Oct. 6, 2017) (cit. on p. 150).

[@Bux07] W. A. S. Buxton. *Multi-Touch Systems that I Have Known and Loved*. 2007. URL: `http://www.billbuxton.com/multitouchOverview.html` (visited on Jan. 27, 2018) (cit. on p. 17).

[@CL13] B. Chung and O. Li. *Domain Mall Interactive Wall Display in Hong Kong*. 2013. URL: `https://youtu.be/NRhDpDxTsLA` (visited on Oct. 26, 2017) (cit. on pp. 150, 154).

[@Dav+15] C. Davis, A. Chmieliauskas, G. Dijkema, and I. Nikolic. *Enipedia*. 2015. URL: `http://enipedia.tudelft.nl` (visited on Jan. 5, 2018) (cit. on p. 166).

[@Fac15] Facebook, Inc. *Facebook Graph API Explorer*. 2015. URL: `https://developers.facebook.com/tools/explorer` (visited on Aug. 19, 2017) (cit. on pp. 85, 165).

[@Gra09] Graph# on CodePlex (coordinating user: palesz). *Graph#, graph layout framework for WPF applications*. 2009. URL: `http://graphsharp.codeplex.com/` (visited on Aug. 19, 2017) (cit. on p. 85).

[@lib03] libavg on GitHub (main contributor: Ulrich von Zadow). *libavg*. 2003. URL: `https://github.com/libavg/libavg` (visited on Oct. 31, 2017) (cit. on pp. 115, 163).

[@Mic13] Microsoft. *Microsoft Kinect for Xbox One*. 2013. URL: `https://developer.microsoft.com/en-us/windows/kinect` (visited on Oct. 31, 2017) (cit. on pp. 135, 163).

[@Mic15] Microsoft. *Microsoft Surface Hub*. 2015. URL: `https://www.microsoft.com/en-us/surface/devices/surface-hub/overview` (visited on Jan. 27, 2018) (cit. on p. 173).

[@Nat] NaturalPoint, Inc. DBA OptiTrack. *OptiTrack – Motion Capture Systems*. URL: `http://optitrack.com/` (visited on Nov. 17, 2017) (cit. on pp. 115, 131).

[@Per06] Perceptive Pixel, Inc. (since 2012: division of Microsoft). *Perceptive Pixel Display*. 2006. URL: `http://www.perceptivepixel.com/` (visited on Aug. 19, 2017) (cit. on pp. 85, 90).

[@Qui07] QuickGraph on CodePlex (coordinating user: Jonathan de Halleux). *QuickGraph, Graph Data Structures and Algorithms for .NET*. 2007. URL: `http://quickgraph.codeplex.com/` (visited on Aug. 19, 2017) (cit. on p. 85).

List of Figures

List of Tables

Colophon

This thesis was typeset with $\LaTeX\ 2_\varepsilon$. It uses the *Clean Thesis* style developed by Ricardo Langner. The design of the *Clean Thesis* style is inspired by user guide documents from Apple Inc.

Download the *Clean Thesis* style at `http://cleanthesis.der-ric.de/`.