

Visual Analytics of Eye-Tracking and Video Data

Von der Fakultät Informatik, Elektrotechnik und
Informationstechnik der Universität Stuttgart
zur Erlangung der Würde eines
Doktors der Naturwissenschaften (Dr. rer. nat.)
genehmigte Abhandlung

Vorgelegt von

Kuno Kurzhals

aus Temeschburg (RO)

Hauptberichter: Prof. Dr. Daniel Weiskopf
Mitberichter: Prof. Min Chen,
BSc, PhD, FBCS, FEG, FLSW
Tag der mündlichen Prüfung: 18. Dezember 2018

Visualisierungsinstitut
der Universität Stuttgart

2018

Acknowledgments

First and foremost, I would like to thank Daniel Weiskopf for giving me the opportunity to be his PhD student. His support and supervision made this thesis possible. And thank you, Daniel, for giving me the freedom to follow some of my stranger ideas.

Moreover, I thank Min Chen for reviewing this thesis and for taking part in my PhD defense. I would also like to thank Markus Höferlin for supervising my work as a student. He got me interested in video visualization and influenced my research direction initially. Further thanks to Michael Burch and Filip Sadlo who supported me at the beginning and during my thesis.

Thanks to all my co-authors who made this thesis possible: Michael Burch, Tanja Blascheck, Florian Heimerl, Marcel Hlawatsch, Markus John, Rudolf Netzel, Robert Krüger, Thomas Ertl, Natalia and Gennady Andrienko, Brian Fisher, Thies Pfeiffer, Yongtao Hu, Wenping Wang, and Christof Seeger. I also want to thank my students Stefan Strohmaier, Emine Çetinkaya, Paul Kuznecov, and Maurice Koch for their excellent work that made parts of this thesis possible.

Special thanks goes to my roommates Florian Heimerl and Qi Han. I thank both of you for the pleasant time we spent sharing an office. Thank you, Marcel, not for the FaPra, but for the fun we had discussing new ideas and realizing them. Thanks to Michael Krone for the FaPra and to Michael Wörner, who dubbed my videos more than once. I enjoyed the company of the Pokkez gang: Grzegorz, Michael S., Michael B., Fabian, Dominik, Valentin, Gleb, and Moataz. And I want to thank the core of our board game group, consisting of Oli, Alex, Marcel, and Dominik for joining the relaxing game nights while I was writing this thesis.

I want to express my gratitude to my family, especially to my parents, who supported me during my studies and my PhD thesis. And finally, thank you, Anne, for your support during those years. You all motivated me to follow my interests and work on this thesis in the first place.

My work at the Visualization Research Center of the University of Stuttgart was funded by the German Research Foundation DFG within the SFB 716, the Priority Program Scalable Visual Analytics (SPP 1335), and the SFB/Transregio 161.

Table of Contents

Acknowledgments	iii
List of Figures	vii
List of Tables	ix
Summary	x
Zusammenfassung	xi
1 Introduction	1
1.1 Research Questions	2
1.2 Outline and Contributions	4
2 Visual Support for Video Analysis	9
2.1 Low-Level Computer Vision	11
2.1.1 Image Comparison	11
2.1.2 Optical Flow	13
2.2 Visualization of Video Data	18
2.2.1 Visualization Reference Model	18
2.2.2 Sensemaking Process	19
2.2.3 Video Visualization	21
2.2.4 Video Visual Analytics	25
2.3 Example: Visual Movie Analytics	26
2.3.1 Related Work	27
2.3.2 Visual Analytics Approach	30
2.3.3 Data Pre-Processing	31
2.3.4 Analytics Environment	33
2.3.5 Analytical Reasoning	36
3 User-Based Evaluation of Visualization	39
3.1 Methodology	40
3.1.1 Quantitative Evaluation	40
3.1.2 Qualitative Evaluation	42
3.2 User Performance Studies	43
3.3 Repertory Grids for Visualization	44
3.3.1 The Repertory Grid Technique	45
3.3.2 Related Work	47

Contents

3.3.3	Visualization-Specific Requirements	48
3.3.4	How to Conduct the Interview	49
3.3.5	Comparison with other Qualitative Methods	53
3.3.6	Application Scenarios	54
3.4	Eye Tracking for Visualization	56
3.4.1	Foundations of Eye Tracking	56
3.4.2	Including Eye Tracking in Evaluation Methodology	59
3.4.3	Example: Evaluation of Subtitle Layouts	64
3.4.4	Eye-Tracking Evaluation in the Visualization Community	67
4	Visualization of Eye-Tracking Data	75
4.1	Eye-Tracking Visualization Pipeline	76
4.2	Taxonomy	78
4.2.1	Task-Related Categories	79
4.2.2	Technical Categories	82
4.3	Categorization of Visualization Techniques	85
4.3.1	State of the Art	85
4.3.2	Contributed Techniques	87
4.4	Benchmark Data for Visualization Techniques	89
5	Analyzing a Single Video and Multiple Participants	93
5.1	Point-Based Visualization of Gaze Distributions	94
5.1.1	Motion-Compensated Heat Map	94
5.1.2	Space-Time Cube	97
5.1.3	Example: Subtitle Layouts	102
5.1.4	Discussion	103
5.2	AOI-Based Scanpath Analysis	104
5.2.1	AOI Editor	105
5.2.2	AOI Timelines and Scarf Plots	106
5.2.3	AOI Transition Trees	111
5.2.4	Example: UNO Card Game	118
5.2.5	Discussion	120
5.3	Image-Based Eye-Tracking Visualization	121
5.3.1	Gaze Stripes	122
5.3.2	Fixation-Image Charts	128
5.3.3	Gaze-Guided Slit-Scans	133
5.3.4	Discussion	143
6	Visual Analytics for Mobile Eye Tracking	145
6.1	Personal Visual Analytics	146
6.1.1	Eye Tracking in the Context of Personal Visual Analytics	148

Contents

6.1.2	Special Requirements	149
6.1.3	Example: Personal Encounters Analysis	151
6.1.4	Discussion	155
6.2	Image-Based Visual Analytics for Mobile Eye Tracking	157
6.2.1	Related Work	157
6.2.2	Domain-Specific Analysis Process	159
6.2.3	Pre-Processing	161
6.2.4	Analytics Environment	164
6.2.5	Example: Print Media Study	169
6.2.6	Expert User Study	171
6.2.7	Discussion	176
7	Conclusion	179
7.1	Summary of Chapters	180
7.2	Overarching Discussion	181
7.2.1	Research Question 1	181
7.2.2	Research Question 2	182
7.2.3	Research Question 3	182
7.3	Future Directions	184
	Author's Work	187
	Bibliography	191

List of Figures

1.1	Thesis structure	3
2.1	Example scenarios for video analysis	10
2.2	Histogram comparison	12
2.3	Shot detection example	14
2.4	<i>FlowBrush</i> processing steps	15
2.5	Particle tracing	17
2.6	<i>FlowBrush</i> examples	18
2.7	Reference model for visualization	19
2.8	Sensemaking process	20
2.9	Fast-forward visualization	23
2.10	Attention-guiding video visualization	24
2.11	Processing pipeline for visual movie analytics	30
2.12	Movie script example	32
2.13	Alignment for script and movie	33
2.14	Multi-layered timelines	34
2.15	Tagline visualization	35
2.16	Motion tag example	36
2.17	Analytics environment for movie analysis	37
3.1	Experimental procedure for quantitative evaluation	41
3.2	Cartoon stimulus for detection tasks	43
3.3	Performance for fast-forward visualizations	43
3.4	Performance for attention-guiding visualizations	44
3.5	Repertory grid technique applied to information visualization	45
3.6	Procedure of the repertory grid technique	46
3.7	Important steps for the repertory grid	50
3.8	Visual interface for conducting a repertory grid interview	52
3.9	Remote eye tracker and glasses	57
3.10	Evaluation pipeline including eye tracking	59
3.11	Standard methods for visual analysis of eye-tracking data	62
3.12	Regular and speaker-following subtitles	64
3.13	Boxplots for saccade length and fixation count	66
3.14	Heat maps of approximately 10 seconds of a video with subtitles	66
3.15	Histogram of publications about eye tracking for visualization	67
4.1	Extended visualization pipeline for eye-tracking data	76
4.2	Main categories of the taxonomy of visualization for eye tracking	79

4.3	Taxonomy of visualizations for eye-tracking	83
4.4	Summarization of publications about visualization for eye tracking	86
4.5	Overview of datasets that include video stimuli and eye-tracking data	90
5.1	Schematic example for motion-compensated heat maps	95
5.3	Comparison between regular and motion-compensated heat map	96
5.4	Depiction of gaze data in the space-time cube	98
5.5	<i>ISeeCube</i> for spatio-temporal analysis	101
5.6	Subtitle comparison in the space-time cube	102
5.7	Cluster results for subtitle layouts	103
5.8	Volume visualization for video and eye tracking	104
5.9	Editor for the definition of AOIs	105
5.10	AOIs depicted in the space-time cube	106
5.11	AOI timeline visualization	107
5.12	AOI histograms of gaze, size, and position	108
5.13	Scarf plots of 25 participants	110
5.14	String reduction for the analysis of transition patterns	111
5.15	Creation of a transition tree	114
5.16	Shot sequence with three AOI transition trees	116
5.17	AOI thumbnail creation	117
5.18	AOI timelines for the <i>UNO</i> dataset	118
5.19	AOI transition tree showing frequent sequences in the data	118
5.20	Scanpath comparison based on Levenshtein distance and clustering	120
5.21	Gaze stripes in comparison with scarf plots	123
5.22	Gaze stripes of the <i>Kite</i> video	124
5.23	Gaze stripes enriched by several complementary views	125
5.24	Example clustering of a short sequence of gaze stripes	127
5.25	Fixation-image glyph and sequence of glyphs	129
5.26	Time streams for fixation images	130
5.27	Fixation-image charts overview	131
5.28	Labeling example for the <i>Memory</i> dataset	133
5.29	Slit-scan technique demonstrated on the <i>Car Pursuit</i> video	135
5.30	Slit-scan visualization scheme	136
5.31	Framework for image-based scanpath comparison	138
5.32	Matrix overview of multiple metrics	139
5.33	Recorded gaze point examples from smooth pursuit patterns	140
5.34	Slit-scans of participants watching the <i>Memory</i> video	142
6.1	Radial visualization for personal visual analytics	152
6.2	AOI cloud on mobile device	153
6.3	AOI cloud for eight persons over four videos	154

6.4	Analysis process for mobile eye-tracking data	160
6.5	Video segmentation and image clustering process	162
6.6	Segmentation example of a thumbnail sequence	163
6.7	System overview for mobile eye-tracking analysis	164
6.8	Cluster editor for thumbnail labeling	167
6.9	Video player for gaze replay and annotation	168
6.10	Visual stimulus for the investigated user study	169
6.11	Average annotation times for the dataset	170
6.12	Comparison of gaze distribution on AOIs	170
6.13	Scarf plots of all participants in the dataset	172
6.14	Scarf plots for the expert user study	172
6.15	Timeline overview with the longest time interval	172

List of Tables

3.1	Comparison of qualitative evaluation methods	53
4.1	Recorded stimuli with a description of the stimulus settings	91
5.1	Comparison of implemented scanpath similarity metrics	137
5.2	Averaged F_1 -scores for smooth pursuit stimuli	141
5.3	Averaged Spearman correlations of similarity values	141
6.1	Results for the order of AOI visits	173
6.2	Results for the gaze duration on AOIs	174
6.3	Results for the longest time interval	174
6.4	Questionnaire results for the expert user study	175

Summary

Eye tracking, i.e., the detection of gaze points, becomes increasingly popular in numerous research areas as a means to investigate perceptual and cognitive processes. In comparison to other evaluation methods, eye tracking provides insights into the distribution of attention and sequential viewing behavior, which are essential for many research questions. For visualization research, such insights help assess a visualization design and identify potential flaws. Gaze data coupled with a visual stimulus poses a complex analysis problem that is approached by statistical and visual methods. Statistical methods are often limited to hypothesis-driven evaluation and modeling of processes. Visualization is applied to confirm statistical results and for exploratory data analysis to form new hypotheses. Surveying the state of the art of visualizations for eye tracking shows a deficiency of appropriate methods, particularly for dynamic stimuli (e.g., videos).

Video visualization and visual analytics provide methods that can be adapted to perform the required analysis processes. The automatic processing of video and gaze data is combined with interactive visualizations to provide an overview of the data, support efficient browsing, detect interesting events, and annotate important parts of the data. The techniques developed for this thesis focus on the analysis of videos from *remote* and from *mobile* eye tracking. The discussed remote eye-tracking scenarios consist of one video that is investigated by multiple participants. Mobile eye tracking comprises scenarios in which participants wear glasses with a built-in device to record their gaze. Both types of scenarios pose individual challenges that have to be addressed for an effective analysis. In general, the comparison of gaze behavior between participants plays an important role to detect common behavior and outliers.

This thesis addresses the topic of *eye tracking and visualization* bidirectionally: Eye tracking is applied in user studies to evaluate visualization techniques beyond established performance measures and questionnaires. The current application of eye tracking in visualization research is surveyed. Further, it is discussed how existing methodology can be extended to incorporate eye tracking for future analysis scenarios. Vice versa, a set of new visualization techniques for data from remote and mobile eye-tracking devices are introduced that support the analysis of gaze behavior in general. Here, techniques for raw data and for data with annotations are introduced, as well as approaches to perform the tedious annotation process more efficiently.

Zusammenfassung

Eye-Tracking, d.h., die Erkennung und Verfolgung von Blickpunkten, wird in zahlreichen Forschungsbereichen immer beliebter, um Wahrnehmungs- und kognitive Prozesse zu untersuchen. Im Vergleich zu anderen Evaluationsmethoden gewährt Eye-Tracking Einblicke in die Aufmerksamkeitsverteilung und in sequenzielles Blickverhalten, welche für viele Forschungsfragen unerlässlich sind. In der Visualisierungsforschung helfen solche Einblicke, ein Visualisierungsdesign zu bewerten und mögliche Schwächen zu identifizieren. Blickdaten kombiniert mit einem visuellen Stimulus stellen ein komplexes Analyseproblem dar, welches mit statistischen und visuellen Methoden angegangen wird. Statistische Methoden beschränken sich oft auf die hypothesengetriebene Auswertung und Modellierung von Prozessen. Visualisierung wird zur Bestätigung statistischer Ergebnisse und zur explorativen Analyse für die Formulierung neuer Hypothesen eingesetzt. Der aktuelle Stand der Technik von Eye-Tracking-Visualisierungen weist einen Mangel an geeigneten Methoden auf, insbesondere für dynamische Stimuli (z.B. Videos).

Videovisualisierung und visuelle Analytik bieten Methoden, die an die benötigten Analyseprozesse angepasst werden können. Die automatische Verarbeitung von Video- und Blickdaten wird kombiniert mit interaktiven Visualisierungen, um einen Überblick über die Daten zu erhalten, effizientes Durchsuchen zu unterstützen, interessante Ereignisse zu erkennen und wichtige Teile der Daten zu annotieren. Die Techniken, welche in dieser Dissertation entwickelt wurden, fokussieren sich auf die Analyse von Videos von *Remote-* und *mobilem* Eye-Tracking. Die besprochenen Remote-Szenarien beinhalten ein Video, das von mehreren Teilnehmern betrachtet wird. Mobiles Eye-Tracking umfasst Szenarien, in denen die Teilnehmer eine Brille mit einem eingebauten Gerät tragen, um ihren Blick aufzunehmen. Beide Arten von Szenarien stellen individuelle Herausforderungen dar, die für eine effektive Analyse angegangen werden müssen. Im Allgemeinen spielt der Vergleich des Blickverhaltens zwischen den Teilnehmern eine wichtige Rolle um Gemeinsamkeiten und Ausreißer zu erkennen.

Diese Arbeit beschäftigt sich mit dem Thema *Eye-Tracking und Visualisierung* in beide Richtungen: Eye-Tracking wird in Nutzerstudien eingesetzt, um Visualisierungstechniken über etablierte Leistungsmaßstäbe und Fragebögen hinaus zu bewerten. Die aktuelle Anwendung von Eye-Tracking in der Visualisierungsforschung wird untersucht. Darüber hinaus wird diskutiert, wie bestehende Methoden erweitert werden können, um Eye-Tracking in zukünftige Analyseszenarien zu integrieren. Umgekehrt werden eine Reihe neuer Visualisierungstechniken für Daten von Remote- und mobilen Eye-Trackern vorgestellt, welche die Analyse des Blickverhaltens im Allgemeinen unterstützen. Hierbei werden Techniken für Rohdaten und für Daten mit Annotationen vorgestellt, sowie Ansätze, die den mühsamen Annotationsprozess effizienter gestalten.

Introduction

Eye tracking is the process of capturing viewing behavior of people watching and eventually interacting with a visual stimulus, for example, a video or a computer application. Revealing where persons looked at provides valuable insights into their perceptual and cognitive processes. Apart from the common application for marketing purposes (*How interesting is a specific product to a test group?*), eye tracking is applied in psychology and in numerous other research fields for the evaluation of gaze behavior under the conditions of a stimulus. The difficulty in analyzing such data increases when comparing multiple participants and in cases where the stimulus changes dynamically.

Visualization research aims to depict data in perceivable ways to help with the analysis for a human interpreter. Showing data with visual representations instead of numbers helps make sense of measurements and facilitates the detection of patterns in the data. Video visual analytics strives to enhance and abstract video data with visualization in combination with automatic processing and interaction, so an analyst can more effectively examine the data for important events instead of having to investigate the whole material.

The combination of eye tracking and visualization offers two research directions:

- **Evaluation of visualization with eye tracking:** Visualization design is guided by heuristics based on human perception and cognition. However, for a new visualization, evaluation is often necessary to confirm if initial assumptions are valid and the technique serves its purpose. Eye tracking assists as one means to evaluate how participants investigate a visualization. For example, one can find out if important visual components were ignored by participants and might require more emphasis in the visualization. Furthermore, visual strategies can be observed that provide a glimpse into the cognitive processes of a participant solving a task. Eye tracking can be integrated easily into evaluation methodology for visualization and provides a valuable addition to existing approaches.

- **Visualization of eye-tracking data:** Visualization is applied to support the analysis of eye-tracking data. Established statistical methods for the evaluation of gaze data are not sufficient to cover a thorough analysis. Visualizations provide support to confirm statistical results and help explore the data for the formalization of new hypotheses. Especially in the case where temporal changes in viewing behavior are important, visualization becomes a necessity for the analysis process. Knowledge from visualizations for video content can be adapted to design new approaches tailored to the requirements of eye-tracking analysis.

The work presented in this thesis covers research in both directions and is motivated by the observation that there is a deficiency of techniques for eye-tracking data analysis. Chronologically, the first publications comprise work on video visualization where surveillance videos were visually enhanced to support human inspection tasks. Among other methods, eye tracking was applied to evaluate the influence of the visualizations on gaze distributions. For the analysis of recorded data, existing techniques were not sufficient. Surveying the state of the art of visualizations for eye tracking revealed that further research is required, especially for techniques with a focus on video stimuli and eye tracking with mobile glasses. Hence, the technical contributions in this thesis have the main focus on the combination of video visual analytics with eye-tracking data. Both data sources introduce individual challenges and are addressed by techniques that require human interpretation. To ease this task, complementary visualization techniques for data summarization, comparison, and exploration are necessary and will be discussed in the course of this thesis.

1.1 Research Questions

The structure of this thesis is based on three main components: (1) video visualization and visual analytics, (2) evaluation for visualization, and (3) visual analytics for eye tracking with a focus on video stimuli (Figure 1.1). First, it will be discussed how video analysis can be supported by visualization. Second, evaluation methodology for visualization and its application in this thesis are discussed. Knowledge from the first two components is combined to develop new techniques to analyze eye-tracking data. Accordingly, the research questions are framed.

Video Visualization and Visual Analytics The interpretation of video data is a complex task that can be supported by computer vision. However, high-level semantic interpretations and decision making are typically left to the human analyst. The combination of computational methods and appropriate, interactive visual representation of the results eases the analytical reasoning, even for large datasets.

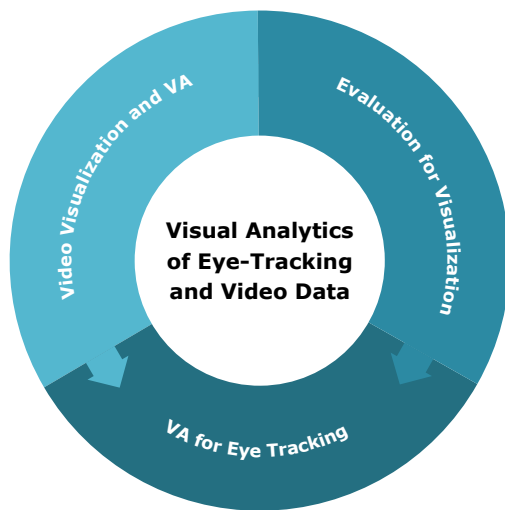


Figure 1.1: Thesis structure: The visualization techniques developed in this work are based on the principles of video visual analytics. Evaluation methodology for visualization includes the application of eye tracking, which requires new techniques for the analysis of complex gaze data in combination with dynamic stimuli. Visual analytics for eye-tracking and video data incorporates knowledge from both fields to provide the required techniques.

Research Question 1

How can we enhance/abstract video material to support specific tasks?

This question focuses on alternative representations for video content, emphasizing specific aspects and incorporating additional data sources (e.g., eye tracking). The techniques developed in this thesis present multiple levels of abstraction from the video content, covering a wide range of possible application scenarios.

Evaluation for Visualization The evaluation of visualization techniques plays an important role in identifying perceptual and cognitive issues with a specific technique, but also in deriving general guidelines for visualization design. In this thesis, multiple user studies for the comparison of visualization techniques were conducted, and different methodological approaches were applied to gain insights into visualization design. Eye tracking is a powerful evaluation technique, as it provides insights that are hard to achieve with other established methods.

Research Question 2

How can we leverage eye tracking to evaluate visualization techniques?

This thesis comprises an overview of user studies including eye tracking in visualization research. Visualization approaches are surveyed and a taxonomy for existing techniques is discussed. Furthermore, multiple eye-tracking studies were conducted to evaluate how participants perceive visualization.

Visual Analytics for Eye Tracking There is a demand for techniques combining automatic data processing and interactive visualization for interpretation and exploration. Hence, the concepts of video visual analytics are applied to improve eye-tracking analysis. The new techniques developed in this thesis can be separated into two categories, i.e., approaches for single videos that were presented to multiple participants and approaches for mobile eye tracking with individual videos from each participant.

Research Question 3

How can we improve the state of the art of visualizations for eye tracking?

With the systematic review of existing techniques to analyze eye-tracking data, a general need for new techniques for dynamic stimulus analysis was identified. With respect to the first question, this thesis introduces new approaches or extends existing techniques for the analysis of video and eye-tracking data.

1.2 Outline and Contributions

This section outlines the structure of the thesis and conveys the topics of each chapter. For the majority of the publications, I am the first author and developed the respective software prototypes. Collaborations with other authors and projects under my supervision are also discussed in the following. My supervisor Daniel Weiskopf was involved in all publications as a co-author and contributed his experience to each paper.

Chapter 2 – Visual Support for Video Analysis This chapter summarizes the foundations of computer vision and visualization for this thesis. The applied concepts are exemplified by respective publications. The provided example of video visualization [12] builds on the work of my Studienarbeit, supervised by Markus Höferlin. For the respective publication, I implemented new visualizations and conducted an eye-tracking study. *FlowBrush* [28] is included as an example of video-based graphics. Michael Stoll and Andrés Bruhn provided their expert knowledge on visual computing and co-authored this publication. The main example, *visual movie analytics* [25], is based on Paul Kuznecov’s B.Sc. thesis, supervised by Markus John, Florian Heimerl, and me. I supervised the visualization concept and techniques for low-level computer vision. This approach was later extended for multiple movies [9].

Chapter 3 – User-Based Evaluation of Visualization The next chapter provides an overview of evaluation methodology and techniques that are applied in this thesis to evaluate visualization. I discuss the *repertory grid* as a method to extend existing

methodology [19] for which I also implemented an interface to conduct interviews. Furthermore, the application of eye tracking to evaluate visualization is discussed. I surveyed existing user studies to provide an overview of the current state of the art. Michael Burch contributed his expertise on eye-tracking studies to these publications. Thies Peiffer co-authored one publication, providing his expertise for eye tracking in virtual reality [14]. Brian Fisher contributed his expertise in cognitive science [21, 22], allowing us to provide a glimpse into the possible future of eye tracking for visualization and visual analytics. As an example, the evaluation of speaker-following subtitles with eye tracking is presented [27]. Emine Çetinkaya conducted and evaluated the study under my supervision as part of her B.Sc. thesis. Yongtao Hu and Wenping Wang provided their expertise and the visual stimuli for the study.

Chapter 4 – Visualization of Eye-Tracking Data This chapter describes how visualization is applied to analyze gaze data. First, visual analysis is discussed from a task-based perspective [26]. This publication is based on the common expertise from my co-authors Michael Burch, Tanja Blascheck, Gennady Andrienko, Natalia Andrienko, and me. The current state of the art of visualization techniques for eye tracking is summarized from the respective publications [1, 4]. Tanja Blascheck and I conducted the main research for these literature surveys. I surveyed the techniques related to dynamic stimuli. The resulting taxonomy was derived in collaboration with Michael Raschke and Michael Burch. I further supervised the creation of a benchmark dataset containing videos and eye tracking data [20]. Fabian Bopp, Jochen Bässler, and Felix Ebinger recorded the data as part of their Projekt INF.

Chapter 5 – Analyzing a Single Video and Multiple Participants The fifth chapter presents techniques for the analysis of gaze data from multiple participants, all of them investigating the same video stimulus. I developed a visual analytics approach with a space-time cube visualization and motion-compensated heat maps [16]. This framework was later named *ISecCube* and extended with methods based on areas of interest for analysis [13, 15]. Florian Heimerl implemented the string-based comparison methods for this framework. Image-based visualization for gaze data is introduced as a promising direction for future research. The *gaze stripes* [24] and their extension *fixation-image charts* [23] were developed together with Marcel Hlawatsch, Florian Heimerl, and Michael Burch. I implemented both approaches, and Florian Heimerl contributed the comparison methods for gaze stripes. Marcel Hlawatsch and Michael Burch helped develop the concept and design. I further introduced the concept of *gaze-guided slit-scans* [18], an image-based approach that was extended [10] by Maurice Koch for his B.Sc. thesis under my supervision.

Chapter 6 – Visual Analytics for Mobile Eye-Tracking This chapter concerns how visual analytics can be applied to data from eye-tracking glasses. First, it is discussed how pervasive eye-tracking could be applied for personal visual analytics [17]. In collaboration with Christof Seeger from the Stuttgart Media University, who provided us with data and feedback on the visualization design, we developed a visual analytics approach, applying the image-based visualization technique from the previous chapter for labeling and analyzing mobile eye-tracking data [29]. Marcel Hlawatsch and I conceptualized the design, and he implemented a timeline overview.

The final chapter concludes my work, providing a summarization and overarching discussion of this thesis. The research questions are discussed with respect to the developed techniques and future research directions are outlined.

Materials from the publications [8, 14, 16, 17, 18, 24, 25, 29] are under copyright of IEEE and reused with kind permission of IEEE under the agreement for reuse in this dissertation. Materials from the publications [10, 15, 20, 21, 23, 27, 28] are under copyright of ACM and reused with kind permission of ACM under the agreement for reuse in this dissertation. Materials from the publications [4, 12, 19] are under copyright of John Wiley and Sons and reused with kind permission of John Wiley and Sons under the agreement for reuse in this dissertation. Materials from the publication [26] are under copyright of Springer Nature and reused with kind permission of Springer Nature under the agreement for reuse in this dissertation. Materials from the publication [22] are under copyright of Sage Publishing and reused with kind permission of Sage Publishing under the agreement for reuse in this dissertation. Materials from the publication [13] are reused with kind permission of the Canadian Human Computer Communications Society (CHCCS) under the agreement for reuse in this dissertation.

I was further involved in other publications which are not part of this thesis, mainly considering my expertise in eye tracking. I was involved in the conceptualization of an eye-tracking study for metro maps [6, 5, 30], a visual analytics approach that combines eye tracking, interaction logs, and think aloud [3]. Furthermore, I provided my expertise for the visual comparison of gaze data with multiple sequence alignment [7] and visual analytics for video applications [31]. Together with Tanja Blascheck, I supervised Stefan Strohmaier's diploma thesis which was later published [2]. The developed space-time cube was also applied to visualize indoor event data [11].

Overall Contributions

This thesis has novel contributions in the field of visualization under numerous aspects:

- **Evaluation methodology for visualization:** This thesis discusses the repository grid as a promising qualitative evaluation method for visualization research [19]. Related work considered only partial aspects while the presented work provides a holistic view of the method and possible application scenarios. For quantitative evaluation, the current application and the potential for future extensions of eye tracking in the context of visualization and visual analytics are discussed [14, 21, 22]. To this point, a systematic review of eye-tracking applications in visualization did not exist. Furthermore, eye tracking is applied to evaluate visualizations for attention guidance [12] and label placement [27]. The results provide new knowledge about the influence of visualization techniques on the distribution of attention and support related work with empirical evidence.
- **Visualization of eye-tracking and video data:** The presented approach for video analysis [9, 25] introduces the concepts of visual analytics in the context of feature films and text. In contrast to related work, the human user is an essential part of the concept to bridge the gap between automatic processing of the data and high-level semantic interpretations. Elements of the applied algorithms are also deployed to create artistic content [28]. Furthermore, this thesis contributes solutions to the coupled data analysis of eye tracking and video. The respective literature surveys [1, 4, 26] comprise the state of the art in this research field and indicate a lack of appropriate techniques for this type of data. A benchmark dataset was created [20] to foster the development of new techniques, that, in contrast to existing datasets, is tailored to contain eye movements specific to dynamic stimuli. The presented research prototype *ISeeCube* [13, 15, 16] is a visual analytics framework that combines existing and new visualization approaches for an efficient interpretation of gaze data from multiple participants recorded with remote eye tracking. It includes established methods for automatic scanpath comparison and supports their interpretation by an appropriate representation of the results. The image-based techniques [10, 18, 23, 24] contribute a new approach to the comparison and interpretation of gaze data from multiple participants. The image-based approach is also integrated into a visual analytics concept for the annotation and interpretation of data from mobile eye tracking [29]. In contrast to the established annotation procedure, the presented method proves to be more efficient and easy to apply. Also in the context of mobile eye tracking, this thesis contributes a discussion of mobile eye tracking for personal visual analytics and a visualization concept for convenient data exploration [17].

Awards

The following publications received an award at the respective venue:

- T. Blascheck, M. John, K. Kurzhals, S. Koch, and T. Ertl. “VA²: A Visual Analytics Approach for Evaluating Visual Analytics Applications”. In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 61–70 – received an *honorable mention* at IEEE VIS 2015.
- K. Kurzhals, M. Hlawatsch, M. Burch, and D. Weiskopf. “Fixation-Image Charts”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2016, pp. 11–18 – received the *visual saliency award* at ETRA 2016.
- K. Kurzhals, M. Stoll, A. Bruhn, and D. Weiskopf. “FlowBrush: Optical Flow Art”. In: *Proceedings of the Symposium on Computational Aesthetics*. 2017, 1:1–1:9 – was rewarded as one of four *best papers* at Expressive 2017.
- K. Kurzhals and D. Weiskopf. “Exploring the Visualization Design Space with Repertory Grids”. In: *Computer Graphics Forum* 37.3 (2018), pp. 133–144 – received an *honorable mention* at EuroVis 2018.

Visual Support for Video Analysis

The analysis of video material plays an important role in numerous application and research domains. Popular examples include Closed Circuit Television (CCTV), sports events, eye tracking, and the analysis of movie content. Based on the analysis task, a problem can be solved automatically or requires human interpretation. This results mainly from the fact that some problems are well-defined, while others are not. Many low-level computer vision tasks are well-defined and can be solved automatically with high accuracy (e.g., the detection and recognition of faces [278]) with state-of-the-art techniques. In contrast, the analysis of semantics in video content (e.g., construing metaphorical imagery in movies) is sometimes ambiguous and worthy of discussion between human domain experts. These issues are part of what is commonly known as the semantic gap:

“ The semantic gap is the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation. ”

Smeulders et al. [266]

Figure 2.1 depicts the mentioned examples, ordered by the degree of automation and human interpretation. In all of the presented example scenarios, problems exist that are automatically solvable, as well as problems that require interpretation. The order is based on current practice with existing methods:

CCTV Recorded videos from surveillance cameras comprise hours of material for an individual camera. Complex systems of numerous cameras (e.g., [317]) create a vast amount of data that requires support by computer systems to provide an efficient sighting of events of interest. Object detection and recognition algorithms with high efficiency were developed to support the search for specific events. Deep Learning is



Figure 2.1: Example scenarios for video analysis: With an increasing degree of semantic abstraction, a human analyst is required to interpret the data.

currently applied in many computer vision scenarios to train computers how to react live to visual input [239]. However, detecting untrained and unexpected events still requires a human to provide labels for classification.

Sport Events Similar to the CCTV scenario, the analysis of sport events, for example, soccer games [91], poses challenges of individual player detection and tracking. In addition, the rules of the specific game have to be considered for a thorough analysis. To this point, most existing analysis approaches require human annotation to process complex analysis tasks.

Eye Tracking The application of eye tracking to visual stimuli [99] for the analysis of a participant’s perceptual and cognitive processes introduces an additional data channel. In such cases, the combined analysis of video and gaze data is necessary to understand events that cause a specific effect in the data. With the complexity of an additional data source, the automated detection of relevant events in the data becomes difficult, and the final interpretation of the results requires a human analyst.

Movie Analysis With increasing abstraction of the insights derived from the data, human interpretation is required. Questions like, “*How did the depiction of women smoking in Hollywood movies change over the last decades?*” [106], require a high degree of abstraction from automatically detectable visual features. For many research questions, the social and historical context when a video was created is also important for interpretation, which is highly reliant on expert knowledge.

This thesis covers work that aims to support a human analyst in such analysis tasks that cannot be solved automatically to this point. The presented work comprises research on visual support for CCTV and movie analysis. However, the main focus of this thesis lies on video data in combination with eye tracking. To achieve this goal, the concept of visual analytics [171] is applied, combining automatic data processing with interactive visualization for analytical reasoning.

This chapter is partly based on the following publications:

- K. Kurzhals, M. Höferlin, and D. Weiskopf. “Evaluation of Attention-Guiding Video Visualization”. In: *Computer Graphics Forum* 32.3 (2013), pp. 51–60 [12]
- K. Kurzhals, M. John, F. Heimerl, P. Kuznecov, and D. Weiskopf. “Visual Movie Analytics”. In: *IEEE Transactions on Multimedia* 18.11 (2016), pp. 2149–2160 [25]
- M. John, K. Kurzhals, S. Koch, and D. Weiskopf. “A Visual Analytics Approach for Semantic Multi-Video Annotation”. In: *Proceedings of the 2nd Workshop on Visualization for the Digital Humanities*. 2017, pp. 1–5 [9]
- K. Kurzhals, M. Stoll, A. Bruhn, and D. Weiskopf. “FlowBrush: Optical Flow Art”. In: *Proceedings of the Symposium on Computational Aesthetics*. 2017, 1:1–1:9 [28] 🏆

The remainder of this chapter summarizes the foundations of applied video processing steps (Chapter 2.1), visualizations for video data (Chapter 2.2), and video visual analytics (Chapter 2.2.4). As an example of the application of the concepts of video visual analytics, a developed approach for the analysis of movies is presented (Chapter 2.3).

2.1 Low-Level Computer Vision

Established techniques for low-level computer vision tasks are utilized to derive information from the data to visualize. The work in this thesis mainly relies on techniques for image comparison and optical flow. It should be mentioned that the discussed approaches are often interchangeable with alternative techniques, as they are mainly part of pre-processing stages and therefore open for future work that improves the results. Since the development of new computer vision techniques is not the focus of this thesis, it is referred to the literature on the foundations of computer vision [59, 136] and the main techniques applied in multiple projects are briefly discussed.

2.1.1 Image Comparison

Numerous tasks in image processing require quantification of the similarity between two pictures. In this thesis, similarity measures based on color and feature histograms are mainly applied for clustering and search queries.

Histogram-based

An image can be expressed by a distribution of its components. One basic approach is the representation by color histograms, showing the distribution of all pixel contributions in a specific color space. Figure 2.2 shows an example of two images, taken from a magazine with a glossy surface. The left image shows some light reflections. Comparing the histograms of different color channels, the similarity of images can be determined.

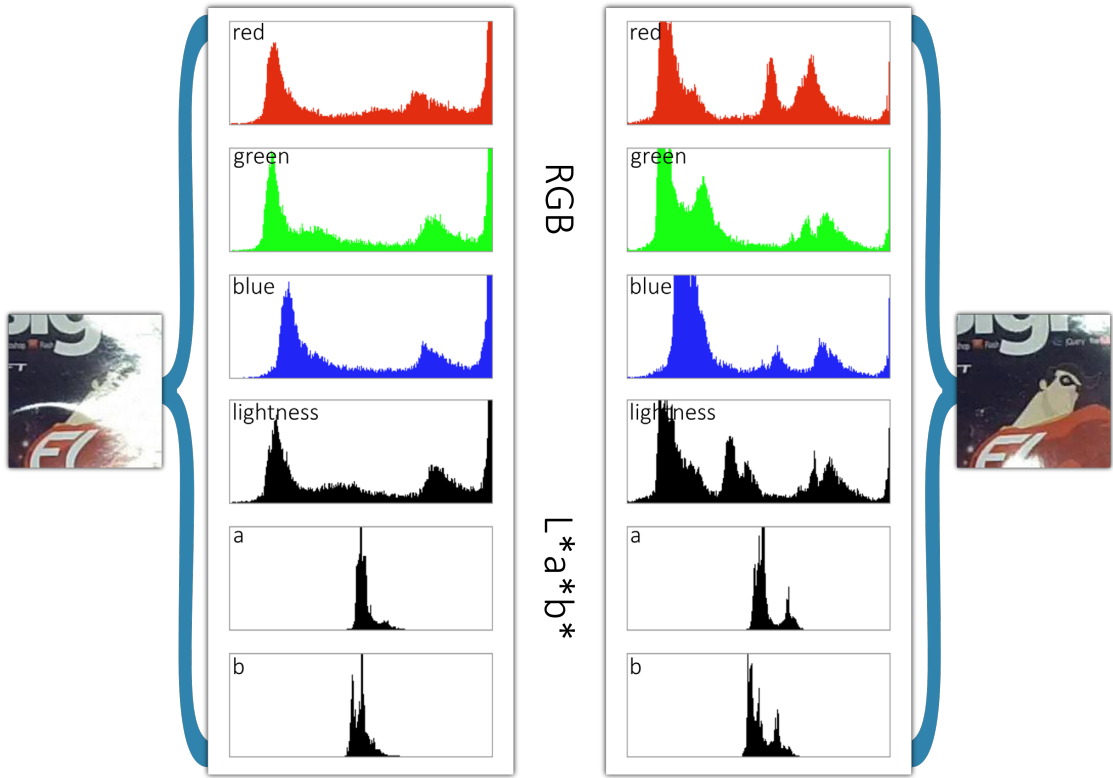


Figure 2.2: Histogram comparison of two images with (left) and without (right) a light reflection on the glossy surface. The color histograms for RGB and the $CIE L^*a^*b^*$ color space are shown.

If the standard RGB histogram is used, one can see that the light reflection has a strong influence on all three channels red , $green$, and $blue$. In the $CIE L^*a^*b^*$ histograms, the reflection has only a strong influence on the channel for lightness (L^*), the a^* and b^* channels remain more robust. Hence, image comparisons in this thesis are performed in color spaces where the lightness channel is excluded to provide more stable results.

To quantify the similarity, the histograms (H_1, H_2) consisting of B bins are compared using the Pearson correlation coefficient ρ :

$$\rho(H_1, H_2) = \frac{\sum_{n=1}^B (H_1(n) - \bar{H}_1)(H_2(n) - \bar{H}_2)}{\sqrt{\sum_{n=1}^B (H_1(n) - \bar{H}_1)^2 \sum_{n=1}^B (H_2(n) - \bar{H}_2)^2}}$$

with

$$\bar{H}_k = \frac{1}{B} \sum_{m=1}^B (H_k(m))$$

In some implementations, the Bhattacharya distance [51] is also applied to compare histograms. From the vast number of available similarity measures [75], these two are chosen as representative measures with good results for the tested datasets. The scenarios in this thesis include images from the same (Chapter 5.3) or a similar video stimulus (Chapter 6.2). Accordingly, the histogram-based similarity is applied for unsupervised clustering and segmentation of time spans with constant image content.

Bag of Features

Color histograms are suitable for image content that does not change significantly in color distribution. However, this approach is sensitive to changes of the distributions, for example, due to different crop margins around an object. Hence, alternative image features are often more reliable for comparison tasks.

For histogram-based representation, approaches that create codebooks of visual features [168] are often applied for texture analysis and scene classification tasks. *Scale-Invariant Feature Transform* (SIFT) [197] is one popular approach to detect image features that are extracted and clustered to create a codebook, or bag of features, for a set of images. From this codebook, a feature histogram can be derived for each input image. In this thesis, the feature histograms are applied to derive similarity measures for unsupervised clustering. Further details are discussed in Chapter 6.2.

2.1.2 Optical Flow

In video analysis, the motion between consecutive video frames is important for numerous reasons. For example, motion information can improve the tracking of objects [276], image stabilization [76], and enables the calculation of in-between frames, e.g., for frame rate up-conversion [311].

Optical flow describes the spatial correspondence between pixels in consecutive video frames. To calculate the optical flow between adjacent frames I^t and I^{t+1} with N pixels, a dense variational method [63] is applied, provided by OpenCV with CUDA support. The displacement for each pixel i (with $i \in \{1, \dots, N\}$) at position \mathbf{x}_i is denoted by

$$\mathbf{w}^t(\mathbf{x}_i) = (u^t(\mathbf{x}_i), v^t(\mathbf{x}_i))^T$$

where $u^t(\mathbf{x}_i)$ is the horizontal displacement and $v^t(\mathbf{x}_i)$ is the vertical displacement. In this thesis, tracking of specific semantically coherent regions (e.g., objects) is not applied. Instead, flow information is used to identify shots in edited video content, shot comparison based on motion, and to assemble motion paths for creating artistic output images. Optical flow is also applied to calculate motion-compensated heat maps for eye-tracking data on video stimuli. This technique is described in detail in Chapter 5.1.1.

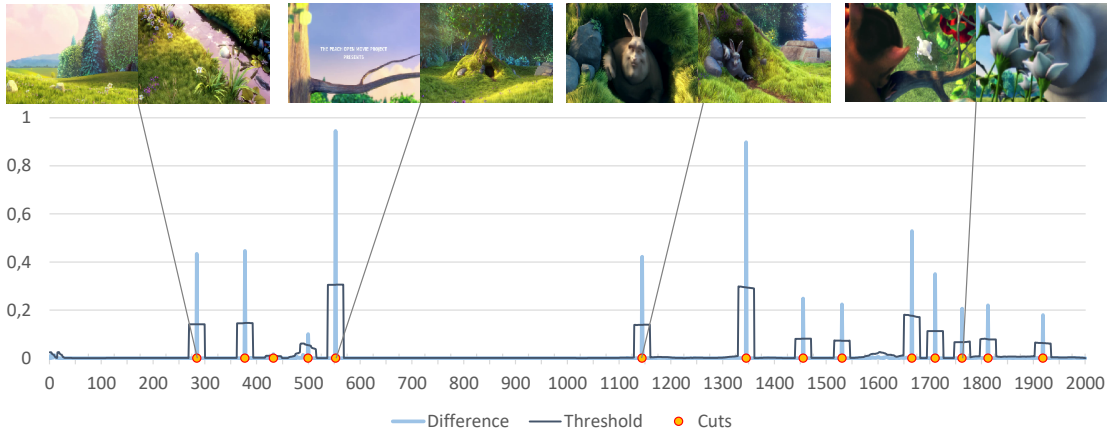


Figure 2.3: Shot detection based on difference D of the forward and backward optical flow field of consecutive frames.

Shot Detection

Optical flow information is applied for shot detection in edited video content. This approach is based on the assumption that the continuity of the flow is disrupted by abrupt cuts between video shots.

Temporal discontinuities in the optical flow, caused by cuts at shot boundaries result in high differences in the flow fields in forward and backward direction:

$$\mathbf{w}_{\text{diff}}^t(\mathbf{x}_i) = \mathbf{w}_{\text{fwd}}^t(\mathbf{x}_i) - \mathbf{w}_{\text{bwd}}^t(\mathbf{x}_i)$$

Only the magnitude of the difference vector is used, independent of the direction:

$$f(\mathbf{w}_{\text{diff}}^t(\mathbf{x}_i)) = \|\mathbf{w}_{\text{diff}}^t(\mathbf{x}_i)\|_2$$

For shot detection, the normalized difference D between the displacement vectors in both directions is calculated:

$$D = \frac{1}{N} \sum_{i=1}^N \|f(\mathbf{w}_{\text{diff}}^t(\mathbf{x}_i))\|$$

Here the L_1 norm is applied because it is robust against outliers. In general, if a shot appears, D will be significantly larger compared to a context with regular motion, where the flow calculation in both directions yields to similar results. For the detection of shots, an adaptive threshold [135] is applied based on a time window of 30 frames. Depending on the frame rate of the movie, this corresponds to an approximate time span of one second. Figure 2.3 shows an example of calculated values for the first 2000

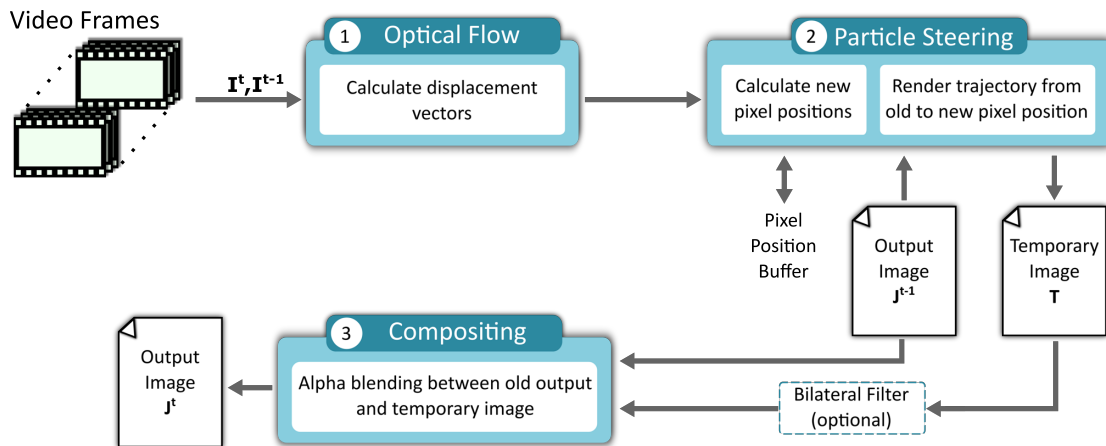


Figure 2.4: FlowBrush video processing steps: ① the optical flow is calculated, ② the pixel displacement is rendered as a trajectory into a temporary image, and ③ compositing of the temporary image and the output from the previous time step is performed.

frames of the *Big Buck Bunny*¹ video. Temporal discontinuities result in high peaks of the difference values that can be detected easily. Note that the motion-based algorithm is not only chosen due to its performance on shot detection [55] but also to extract the motion fields that are required for other analysis steps. This technique is applied for movie analysis (Chapter 2.3) and eye tracking of videos (Chapter 5.1).

Motion Similarity

With the optical flow available, a similarity metric for scenes based on motion is calculated. The motion histogram for $H_{B,t}$ for a frame t contains a number of bins B . The binning is calculated for the angle of the motion vectors. The length of the vector is included as a weighting factor (similar to Schöffman et al. [258]). The histograms are compared using the Pearson correlation coefficient ρ , as described before.

Art with Optical Flow

Optical flow information is also utilized to depict video motion in artistic representations with an approach named *FlowBrush* [28]. The technical procedure to create an image with *FlowBrush* is based on three steps, depicted in Figure 2.4: (1) calculation of the optical flow, (2) particle steering, and (3) compositing. Trajectories for individual video pixels are directly rendered on the canvas, the artist can influence the compositing by an additional bilateral filter step and by adjusting parameters to change the depiction of the trajectories.

¹ <http://www.bigbuckbunny.org>, last checked: October 13, 2018

In the following, let I^k be a series of input images at time steps k and let $I^k(\mathbf{x}_i)$ denote the color of pixel i (with $i \in \{1, \dots, N\}$) at location \mathbf{x}_i in the input coordinate system. Furthermore, let J^k be a series of output images, where J^1 is a blank image, and let T be a temporary image.

Each pixel i from the input coordinate system is assigned a particle in the output coordinate system. Hence, there are N particles. At the beginning, all particles reside in a common seed point \mathbf{a} . Afterward, particle i is steered by the displacements $\mathbf{w}^k(\mathbf{x}_i)$ that are estimated in each time step k at the fixed location \mathbf{x}_i . These displacements usually do not form the trajectory of any object in the input images but belong to different objects that move through location \mathbf{x}_i over time. The trace of particle i in the output image at time step t is the aggregation of the independent motions $\mathbf{w}^1(\mathbf{x}_i), \dots, \mathbf{w}^{t-1}(\mathbf{x}_i)$ at location \mathbf{x}_i in the input image.

More formally: Using the seed point \mathbf{a} in the output image, the origin is set $\mathbf{y}_i^1 := \mathbf{a}$ of all visualized traces of the particles i . For each time step t , the displacement vectors $\mathbf{w}^t(\mathbf{x}_i)$ for all pixels i in the input are calculated, and for each i , they are finally aggregated in an output pixel position buffer:

$$\begin{aligned} \mathbf{y}_i^t &:= \mathbf{y}_i^{t-1} + \gamma \mathbf{w}^{t-1}(\mathbf{x}_i) \\ &= \mathbf{y}_i^1 + \gamma \sum_{k=1}^{t-1} \mathbf{w}^k(\mathbf{x}_i) \end{aligned}$$

where γ is an amplification weight. The temporary image T is initialized with J^{t-1} and afterward the path increment of particle i is rendered into T by a line between the positions \mathbf{y}_i^{t-1} and \mathbf{y}_i^t . The color of the line is determined by the color $I^{t-1}(\mathbf{x}_i)$. The procedure is depicted in Figure 2.5, where red boxes correspond to \mathbf{x}_i and blue boxes correspond to \mathbf{y}_i^k at the respective time steps k .

Let us have a look at time step $t = 2$ in Figure 2.5. The temporary image T is initialized with the blank output image J^1 . The motion \mathbf{w}^1 between the first two input images at the pixel with the red box is now considered. As it moves upright and its color in the first image is blue, a blue line is rendered upright starting at the seed point in the temporary image T . Afterward, T is blended with J^1 giving J^2 . At the next time step $t = 3$, T is initialized with J^2 . The red-boxed pixel is orange and moves left by 2 pixels (\mathbf{w}^2). Hence, an orange line is drawn into T going to the left by 2 pixels and starting at the end of the last line. Finally, T is blended with J^2 , creating J^3 .

The displacements only affect the output image while the positions \mathbf{x}_i in the input remain unaltered (see red box in Figure 2.5). Since both coordinate systems are different, the resolution of the output image is independent from the input. Hence, the presented approach can generate high-resolution images from low-resolution input.

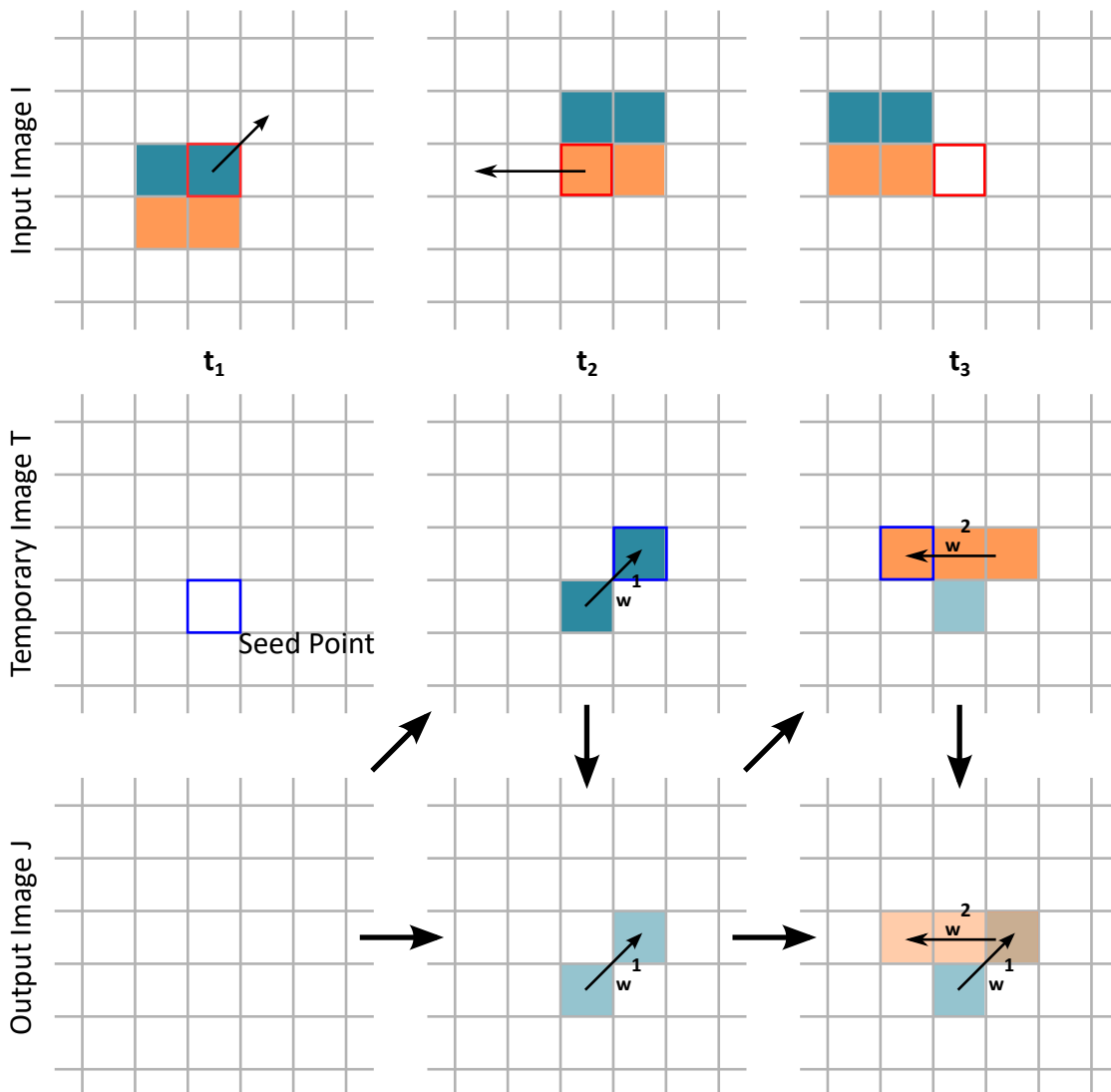


Figure 2.5: Particle trace for one pixel of the original video (red border) and its corresponding particle (blue border). If an object moves through this pixel position, the displacement is drawn as a line with the current color of the pixel.

The compositing step blends the previous output image J^{t-1} with the adjusted temporary image T from the current step

$$J^t = \alpha T + (1 - \alpha)J^{t-1}$$

with a blending weight $\alpha \in [0, 1]$. This iterative alpha blending approach is also applied for interactive vector field visualizations. Only the previous and the current time step are required for computation, which makes this method efficient for real-time



Figure 2.6: Two examples created with the *FlowBrush* technique. Depending on the video input, the user has countless possibilities to create a unique picture from the derived motion patterns.

applications [304]. Figure 2.6 shows two examples of resulting pictures from *FlowBrush*. Both pictures were created using a webcam. Influenced by the motion and objects with different colors, the results represent individual artwork created by the user.

2.2 Visualization of Video Data

Processed video data in semi-automatic systems requires an appropriate representation of results for human interpretation. Statistics and automatically generated textual summarizations provide valuable information but often require experienced users to interpret the results. Hence, visualization can provide effective representations of important features to make sense of the data. In the following, established principles of the visualization reference model and the sensemaking process are discussed for video visualization and video visual analytics.

2.2.1 Visualization Reference Model

The process from raw data to task-specific data visualization for analysis purposes is depicted in Figure 2.7. Following the visualization reference model described by Card and colleagues [70], raw data in idiosyncratic format is transformed to relational data tables extended to include metadata (e.g., optical flow). A visual mapping step transforms these data tables in visual structures (e.g., arrow glyphs) consisting of spatial substrates, marks, and graphical properties [200]. View transformations such as positioning, scaling, and clipping mark the final step to provide the views a human can interpret and interact with. Interaction with the different transformation steps is crucial for data exploration and analytical reasoning.

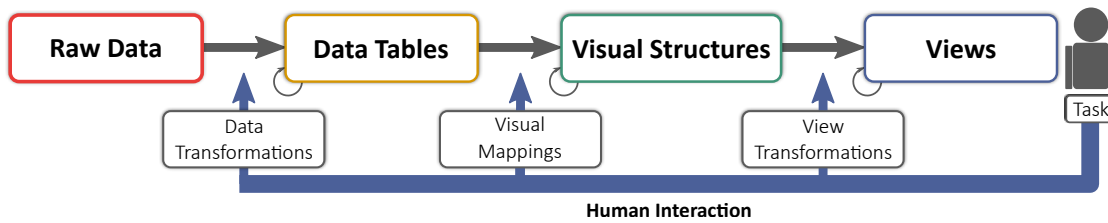


Figure 2.7: Reference model for visualization according to Card et al. [70].

The presented reference model provides a simplified overview of the principles of data visualization. Since the main purpose of visualization is to derive insight from data, the sensemaking process has to be considered as well.

2.2.2 Sensemaking Process

One of the main goals of visualization is to make data understandable to the human user through visual representations. Therefore, sensemaking plays an important role in the design of new visualizations and analytical frameworks. In this context, it can be defined as follows:

“ Sensemaking is the process of searching for a representation and encoding data in that representation to answer task-specific questions. ”

Russel et al. [249]

Such representations are realized by visualization. One important scenario of sensemaking in computer science is intelligence analysis. In the context of this scenario, the sensemaking process can be summarized as a sequence [234]:

Information → Schema → Insight → Product

Information is gathered and summarized in a representation schema for analysis support. Through manipulation of this representation, insight is derived and summarized in a knowledge product. Within this sequence, sensemaking consists of cyclic procedures of searching for representations and encoding information in these representations [249]. Figure 2.8 depicts the notional model of the sensemaking process consisting of bottom-up processes that are often data-driven (e.g., search and filter, read and extract) and top-down processes, driven by the analyst’s knowledge (e.g., reevaluate, search for support). The whole process is framed by two main loops, the *foraging loop* and the *sensemaking loop* [234].

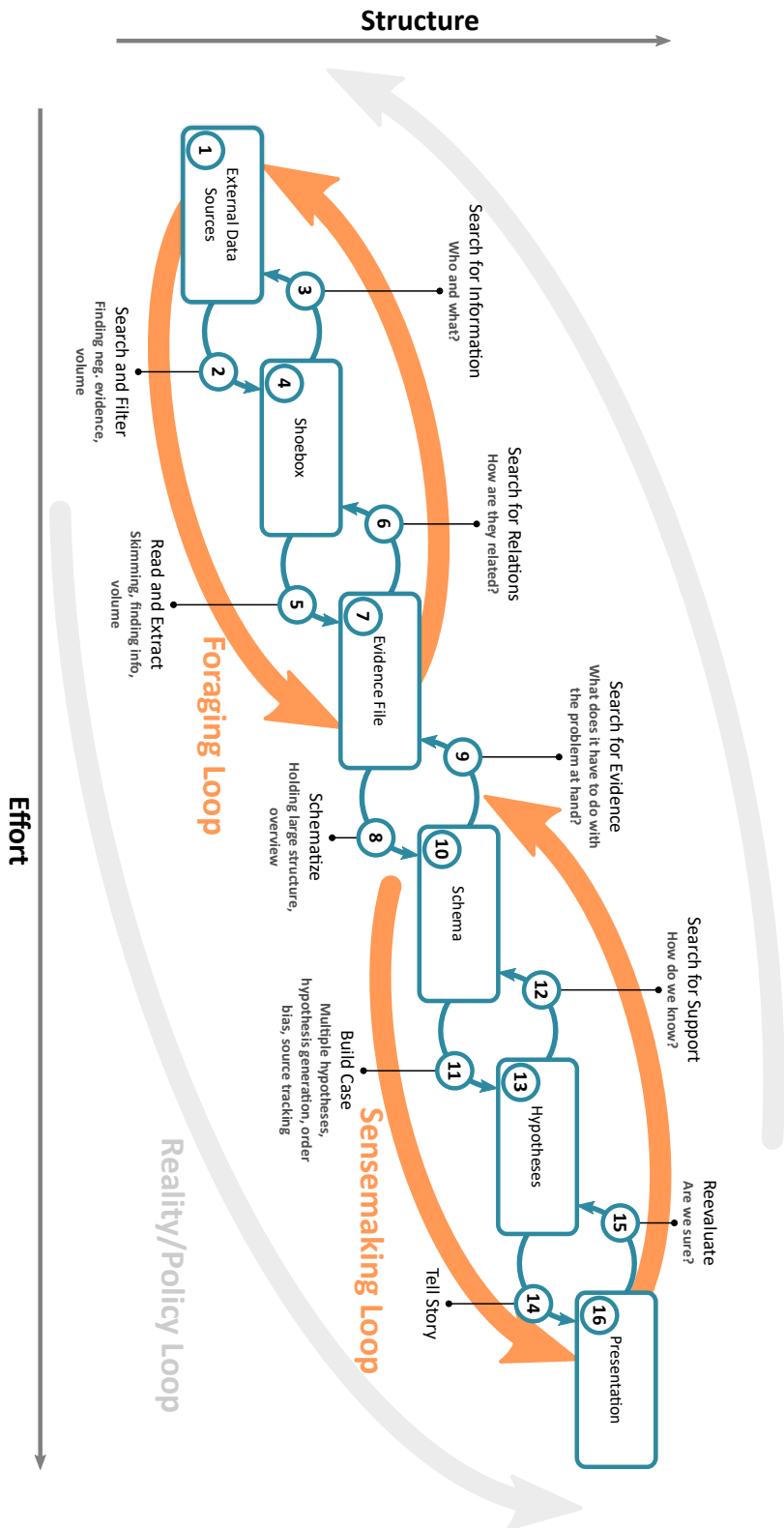


Figure 2.8: Sensemaking process for intelligence analysis according to Pirolli and Card [234].

Foraging Loop This loop involves information seeking and filtering to derive a schema. Analogous to the data transformation step in the visualization reference model, raw data from *external sources* is reduced to a smaller subset, the *shoebox*, for processing. The *evidence file* consists of extracted information from the shoebox items.

Sensemaking Loop This loop concerns the building of a mental model from the schema fitting the evidence. By structuring the information of the evidence file, a *schema* is derived to draw conclusions. *Hypotheses* summarize these conclusions with supporting arguments and are finally condensed in a *presentation*.

Although it focuses on the analysis of text documents for intelligence analysis, this model was partially adapted for numerous data domains. Visualization plays an important part for the development of schemata and the presentation of results, but can also help identify important information in the foraging loop. This thesis presents numerous approaches based on the principles of the visualization reference model and the sensemaking process focusing on the data domain of video and eye tracking. Examples comprise the analysis of movies (Chapter 2.3) and a model for the visual analysis of eye-tracking data (Chapter 4.1).

2.2.3 Video Visualization

Visual representations derived from video material as an input can be divided into video-based graphics that mainly focus on the artistic manipulation and rendering of videos, and video visualization. While the presented *FlowBrush* approach (Chapter 2.1.2) is an example of video-based graphics, video visualization focuses on data analysis:

“ Video visualization is concerned with the creation of a new visual representation from an input video to reveal important features and events in the video. It typically extracts meaningful information from a video and conveys the extracted information to users in abstract or summarized visual representations.

”

Borgo et al. [57]

Video visualization is an emerging research field that focuses on analysis tasks that cannot be solved by automatic processing solely. According to Daniel and Chen [92], two problems remain for automatic processing:

- **Communication of results:** If decision making is in the responsibility of a human operator, processing results have to be communicated accordingly. Statistical results require training to understand, and sequential viewing of results might be time-consuming, revoking the advantage of efficient automatic processing.

- **Reliability under changing circumstances:** Automatic approaches that adapt to changes, for example, changing light conditions or unexpected behavior, are hard to implement and often restricted to specific scenarios.

Visualizations in this category ease the analysis of video data without the need to skim through each video. This concept is not restricted to surveillance videos [92, 146], it is also applied in many other domains, e.g., for sports [145, 185] and movie analysis [163]. This thesis extends the application scenarios for video visualization by the analysis of eye-tracking data. To depict gaze and video data together in interpretable visual summarizations, existing techniques such as space-time cubes [42] and slit-scans [282] are extended to meet domain-specific requirements. Furthermore, new visualizations emphasize the connection of eye-tracking data and the underlying visual stimulus.

Examples

To further exemplify the application of video visualization, two implemented approaches are briefly discussed. The fast-forward visualizations [8] represent video content at increased playback rates and were implemented and evaluated as part of my diploma thesis. The second example, the attention-guiding visualizations [12] aim for a directed distribution of attention and were developed in the context of this thesis. Both examples are applied to the i-LIDS dataset², showing everyday traffic on a street.

Fast-Forward Video Visualization [8] Video recordings (e.g., from CCTV) require much effort to investigate in cases when an automatic analysis is not possible or not trustworthy enough. In such cases, a human expert has to watch the video to find and interpret important events. To solve this task more efficiently, videos are typically watched in fast-forward with increased playback rates. Visualization can be applied to enhance the information depicted in the video. Figure 2.9 shows four different methods for fast-forward in videos: (a) Frame skipping depicts original frames from the video without changes, it reduces the number of shown images by skipping frames, according to the desired playback rate. (b) Temporal blending summarizes frames between the depicted ones, causing a motion blur effect. (c) Object trails improve on the blending approach by preserving the current time step and showing past motion with a ghosting effect. (d) Predictive trajectories are the most abstracted visualization added to the video, showing past and future motion with trail and arrow glyphs.

Attention-Guiding Video Visualization [12] In the second example, the visualization aims to influence the distribution of attention. In search tasks, important events

² Imagery Library for Intelligent Detection Systems (i-LIDS), <https://www.gov.uk/guidance/imagery-library-for-intelligent-detection-systems>, last checked: October 13, 2018

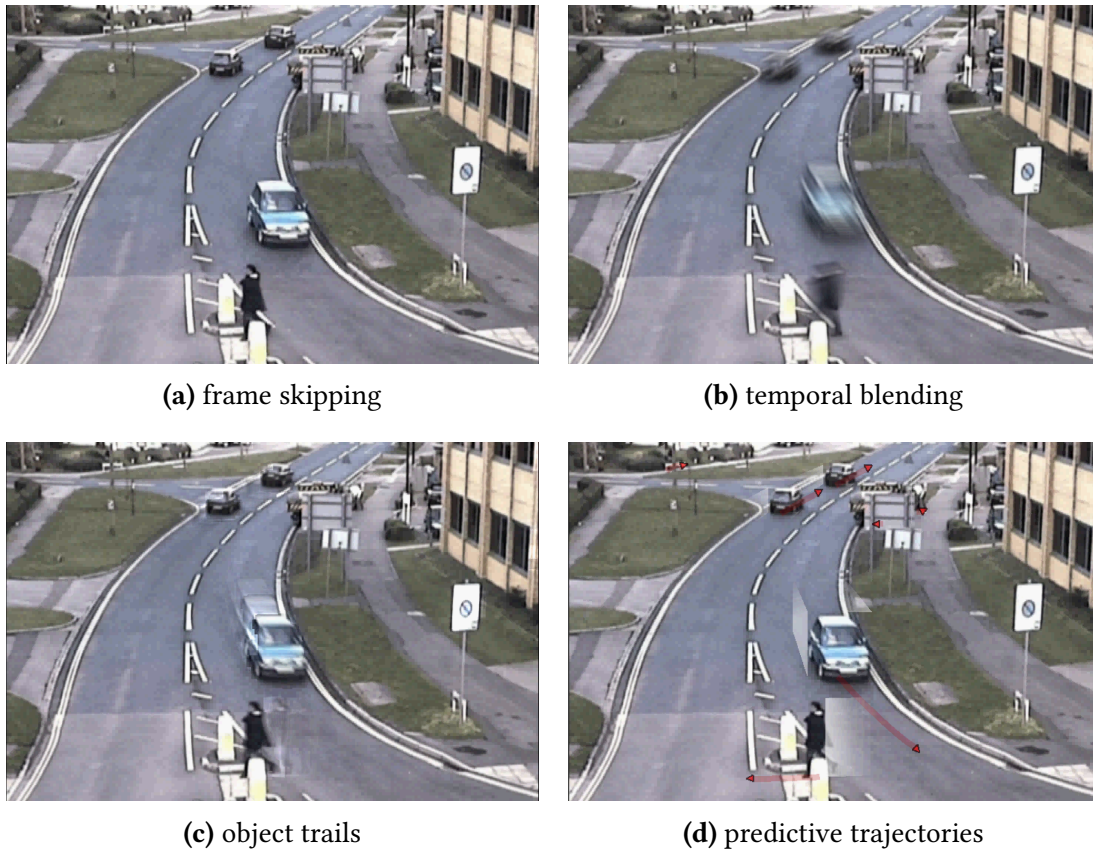


Figure 2.9: Four visualization techniques for fast-forward in videos. In addition to established techniques (a,b), alternative visualizations such as (c) object trails and (d) predictive trajectories depict information about past and future movement without obscuring the current image.

might be missed due to issues with inattentional and change blindness [241]. Such effects happen if a person focuses too much on one specific object, or does not pay attention at all. For this scenario, the attention-guiding visualizations emphasize potential objects of interest to distribute the users' attention equally among the objects. Hence, four different approaches were implemented (Figure 2.10): (a) Bounding boxes of appearing objects highlight where the user should look at. (b) This approach is further extended by equalizing the area size of all objects to reduce the visual saliency due to the size of an object. Potential overlaps of bounding boxes are solved by a force-directed approach that slightly shifts objects away from their original position until the overlap issue is solved. (c) The top-down view replaces the background with a static map and applies a perspective transformation. (d) All annotated objects are separated from the video and placed in a separate grid. Each object receives an individual border for a fast recognition in the original video.

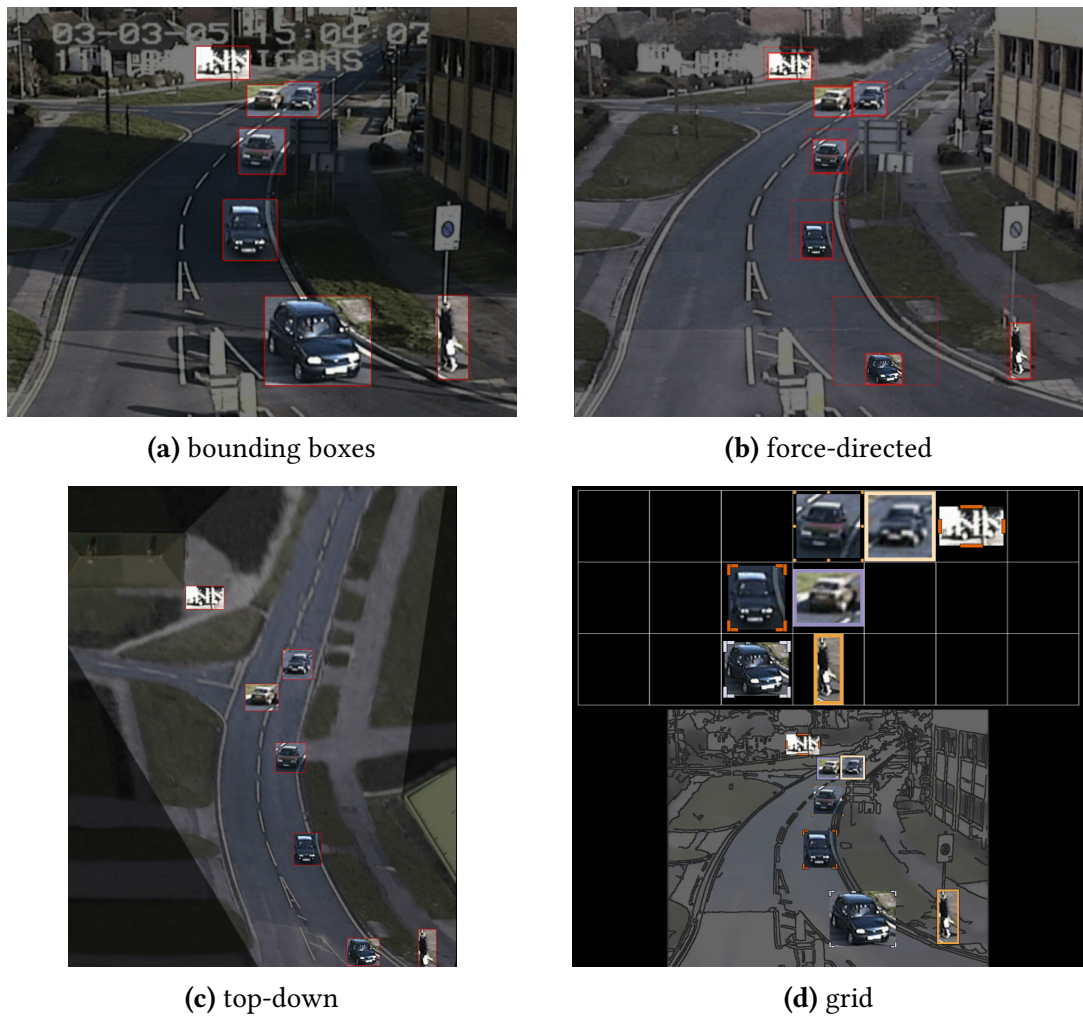


Figure 2.10: Attention-guiding video visualizations. (a) Bounding boxes emphasize objects while the background is faded out. (b) Objects are equalized in area size and overlaps are solved with a force-directed approach. (c) With a perspective transformation, objects are projected on a top-down map. (d) Objects are represented in a compact grid above the video.

In Chapter 3.2, it is further discussed how these visualization techniques influence the performance of participants looking for a specific target. Both publications focus on the direct enhancement of video material, interaction with the visualization is not required. The results are rendered into a new video of either shorter, or the same length as the original. In cases where interaction and automatic processing are applied together with the video visualization, the field of visual analytics is entered.

2.2.4 Video Visual Analytics

With appropriate representations for video analysis, it becomes necessary to interact with the visualization for exploration and reasoning purposes. Visual analytics aims to fuse algorithmic processing, interaction, and visualization for analytical reasoning:

“ Visual analytics is the science of analytical reasoning facilitated by interactive visual interfaces. People use visual analytics tools and techniques to synthesize information and derive insight from massive, dynamic, ambiguous, and often conflicting data; detect the expected and discover the unexpected; provide timely, defensible, and understandable assessments; and communicate assessment effectively for action. ”

Thomas and Cook [285]

To this point, visualization in video applications was discussed as compensation for the issues of automatic processing. With visual analytics, the advantages of automatic processing and visualization can be harnessed through tight interactive coupling of both aspects. This influences the aforementioned sensemaking, as presented in the visual analytics process model presented by Keim and colleagues [171] that explicitly integrates data mining and interaction techniques in the sensemaking process. In the context of video visual analytics, the foraging and the sensemaking loop are supported by three knowledge extraction methodologies [144]:

- **Exploratory Data Analysis (EDA)** [296] In contrast to experiments that apply data analysis for statistical hypothesis testing, exploratory data analysis focuses on the generation of hypotheses through identifying and describing patterns in the data. Interactive visualization supports such exploration in numerous ways.
- **Knowledge Discovery in Databases (KDD)** [110] From a data-driven perspective, KDD processes aim to apply data mining techniques to extract patterns or models from the data to assist an analyst with extracting knowledge. Such patterns require appropriate visual representations for interpretation and for providing feedback to the processes to refine the underlying models.
- **Information Retrieval (IR)** [262] Information retrieval techniques allow the analyst to bring existing knowledge into the analysis. Popular techniques in combination with visualization are search queries, either by filter rules (e.g., keywords) or similarity search (e.g., query by example). This approach can be applied for exploration and confirmation purposes.

EDA and **KDD** processes work on a data-driven basis (bottom-up) while **IR** can be described as knowledge-driven (top-down). By combining the support of all these

methodologies, visual analytics provides powerful means to solve a multitude of analytical tasks. Video analysis tasks (e.g., object annotation) require much time and effort, even for short time spans under investigation. Harnessing the advantages of automatic processing and interactive visualization, many tasks can be solved more efficiently, or a task becomes possible to solve in the first place. As an example, it is demonstrated how text processing, video feature analysis, and interactive visualization can be combined in a visual analytics approach for the semantic annotation of feature films.

2.3 Example: Visual Movie Analytics

Apart from the entertainment value of full-length feature films, the analysis of their content and inherent structure plays an important role. Be it to teach aspiring film students [250] the principles of basic techniques (e.g., shot composition, narrative) or for the analysis of the depiction of social and historical events (e.g., the portrayal of conflicts) for research purposes. In general, the presented approach addresses expert analysts with basic knowledge in movie content analysis as potential users.

The direct approach to such content analysis is watching the movie and taking notes. However, with the technological advances in the last decades, there are numerous semi- and fully automatic systems to help with the annotation and summarization of movies. For searching video content in large databases, retrieval systems based on different similarity metrics exist. Such systems help the analyst identify similar content based on reference videos to formulate a query. These approaches often require users to know in advance what exactly to look for. Other approaches that summarize video content, for example with storyboards [56, 119, 120] or short video skims [190, 267], provide an overview of specific content. They help analysts search for interesting time spans.

Video summarization should be based on four aspects: *who* (W_1), *what* (W_2), *where* (W_3), and *when* (W_4) [78, 195]. These aspects provide spatial and temporal information in the context of personal constellations and events in a movie. In other words, the summarization should enable the analyst to answer questions about “*who* was involved in a scene?”, “*what* happened in the scene?”, “*where* does the scene take place?”, and “*when* does the scene take place?” In the presented approach, the composition of scenes is interpreted as a linear order of time. Consequently, answers to *when* questions relate to the position of the scene within the movie. Flashbacks or temporally resorted scene structures as in the movie *Pulp Fiction* will have to be annotated by the analyst to refer to the temporal order of the content itself.

Furthermore, the four aspects can be combined and descriptive features might be derived to identify relevant scenes. Hence, a comprehensive approach to analyzing the content of a movie should provide both, an overview of important descriptive features and an integrated query interface to search for potentially interesting scenes with similar

content. To define descriptive features, the data is interpreted on two different levels: the image and the semantic level.

Image Level Movie content can be described by structural image and video features. Quantitative measures, such as shot frequencies or motion vectors, can be applied to compare time spans of a movie without providing much information to the four questions discussed above. However, for analyses that consider, for example, stylistic elements (e.g., camera motion), this is valuable information that is also used by most retrieval systems.

Semantic Level The semantic level refers directly to the four questions mentioned above. For a thorough analysis of a movie, the interpretation of what is happening is crucial. Although some of the four questions might be answered with computer vision (e.g., recognizing *who* is in a scene), the semantic gap [266] is a barrier that prevents an analysis solely based on video content. To bridge this semantic gap, two additional text-based data sources are included: the movie script and the movie's subtitles. In contrast to subtitles, a movie script contains not only spoken dialogs, but also scene descriptions, information about locations, and typically a detailed list of characters in a scene. Without the temporal alignment between the script and the final movie, the semantic information can only be interpreted on a textual level. A comparison between the movie script and the subtitles is applied to perform this alignment.

With visual analytics, it is possible to create an analytical environment that supports knowledge discovery on multiple levels of abstraction, i.e., on an image and a semantic level. By incorporating information from multiple text sources, valuable meta-information about person constellations, scene descriptions, and semantic frames is derived for movie scenes. In contrast to existing approaches, the analytical reasoning process is supported by an iterative annotation and analysis concept. Explorative and query-based analysis strategies are supported by multi-layer timelines that depict the data on different levels of detail and allow to compare multiple scenes directly.

2.3.1 Related Work

According to a recent survey on video interaction tools [257], existing approaches can be classified for video annotation, browsing/navigation, editing, recommendation, retrieval, and summarization. The presented approach combines features from summarization, browsing, retrieval, and annotation. One of the major issues to address is that many of the summarization and retrieval approaches focus on automatic algorithms alone to provide results, neglecting the human user. Answers to what the user wants typically depend on the task and cannot be fit by a single retrieval model [310]. Combining the

principles of visual analytics and multimedia analysis [84], the approach aims to ease the investigation of movie content for various analysis tasks.

Video Summarization

Over the last decades, many systems were developed to browse large databases with image and video content [152, 314]. For summarization of the video content, there are representations by keyframes or short video skims that provide an overview of the data and represent query results by video abstraction [190, 293]. A general overview of video summarization techniques is given by DelFabro and Böszörmenyi [93] and Money and Agius [213]. The latter ones differentiate between external, internal, and hybrid summarization techniques. External techniques use information that is not derived directly from the video stream, internal techniques use image, audio, and text features directly related to the video. For example, Jänicke et al. [163] analyze the audio structure of movies to extract specific events and visualize them on a *SoundRiver*. Hence, the presented approach can be interpreted as a hybrid: for internal summarization image- (e.g., [104, 120]) and text-based (e.g., [191, 267]) techniques are applied, and for external summarization, semantic information derived from the movie script is included. Audio data is not included so far, but due to the generic approach, it could be included along with other features without much effort.

Video and Text

The detection of scenes in a movie is typically performed in two ways: either by the analysis of image/video content (e.g., [79, 189, 300, 306]) or by including external information, typically text. The second approach is chosen due to the rich semantic information provided by movie scripts. Wactlar et al. [267, 299] use textual term frequencies along with audio and visual features to create video skims. Their approach focuses on the creation of the skims for browsing video databases. However, interactive incorporation of the human user for analysis purposes is not supported. Sang and Xu [254] describe how to extract and summarize characters from the script and movie data. The summarized data is also represented by video skims. Cour et al. [89] propose a method to align video and movie script text, using closed captions as anchor points. They depict extracted scenes and keyword search-queries by thumbnails as storyboards. Lienhart et al. [193] describe a concept to combine visual and audio features to higher-level operators, to formulate new queries (e.g., for finding commercial breaks). The authors do not include textual information for semantic features.

The ideas of these approaches are extended by (1) increasing the scalability through multi-level abstraction of extracted content information, (2) a more in-depth content analysis by semantic frames, and (3) including the human user into the analysis process through interactive annotation of query results.

Video and Visualization

For a visual representation, basic techniques for video visualization [57] based on timelines are applied. The content is abstracted on multiple layers with different levels of detail (Chapter 2.3.4). Scenes and shots are depicted by timelines with color-coded bars, indicating the presence of a feature over time. Keyframes, as in a storyboard visualization, provide a glimpse into the movie content on the finest level. Similar depictions of multivariate content information can be found in other work: Liu et al. [196, 195] present a hierarchical framework for movie analysis based on interactive combinations of features for search queries. The authors depict shots and scenes by thumbnails. However, depicting only thumbnails can lead to scalability issues for showing the content of a complete movie. Their approach is adapted to integrate the expert user in the analysis process to answer questions concerning *who* (\mathbf{W}_1), *what* (\mathbf{W}_2), *where* (\mathbf{W}_3), and *when* (\mathbf{W}_4). With the multi-layer abstraction of results, better scalability is provided considering the temporal depiction of multiple descriptive features for a more efficient analysis of the movie.

Ponceleon and Dieberger [235] depict multivariate features of videos as a matrix on multiple hierarchy levels or so-called *movieDNA*. Matrix cells display either binary or quantitative information of a feature's appearance by color coding. This concept is adopted for the overview of selected features and fast navigation in the video. However, the authors mainly focus on navigation aspects and do not include an advanced query and annotation process for analytical reasoning. Schöffmann et al. [258] provide an interactive system to browse and search multiple videos for similar content, based on image and motion features. The authors do not include a comparison of content on a semantical level. The approach in this thesis applies a similar depiction of motion over time as one descriptive feature in the visualization.

Related to the presented work are systems for annotating videos (e.g., [134, 176]). These approaches are typically generic, allowing the user to annotate time spans with self-defined concepts, often not restricted to the video content only [114]. However, due to this generic approach, the overview and navigation on unannotated time spans are less supported, and the user has to annotate the data before it can be browsed efficiently. The presented approach in this thesis aims to maintain the generality for annotation but improves the possibilities to identify relevant events more efficiently without having to perform a sequential search through the complete video.

In summary, this thesis presents a visual analytics approach that combines automatic algorithms for text and video analysis with interactive visualization. It includes an overview and details for query results to better analyze, compare, and annotate the structure and content of movies.

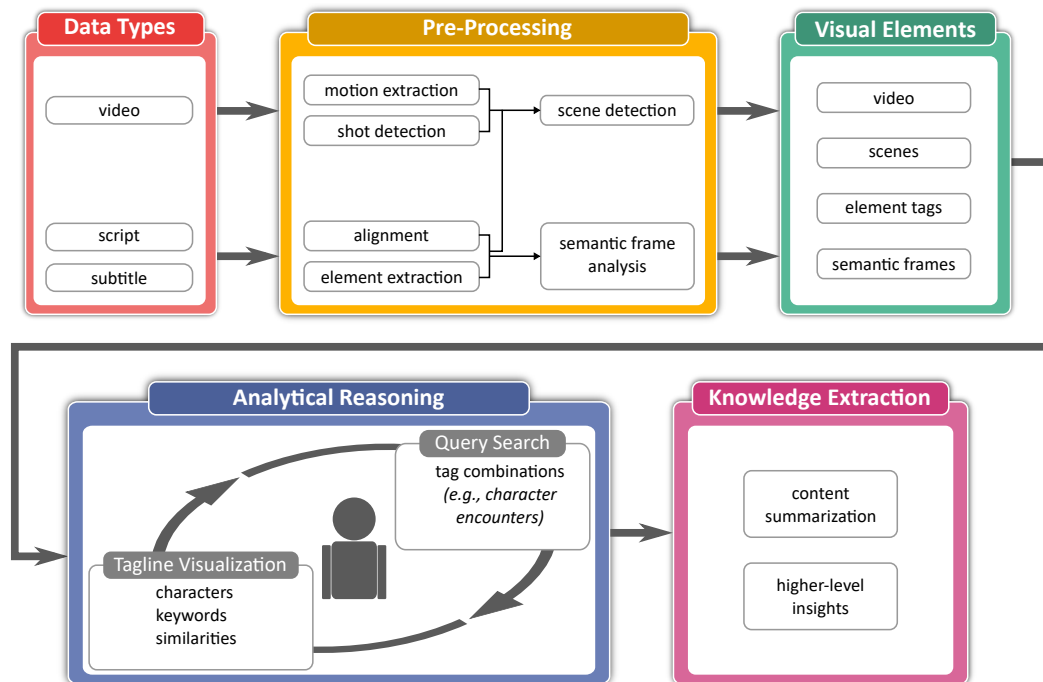


Figure 2.11: Pre-processing is performed automatically on the video and text data to provide an overview of extracted low-level content. Analytical reasoning is then performed in a loop by combining search queries and annotated element tags of different types (e.g., keywords, similarities) to derive higher-level insights.

2.3.2 Visual Analytics Approach

To derive insights from movie content, an analytical approach is proposed that takes advantage of automatic data processing to support the analyst. The reasoning process is supported by an analytics environment that allows for an interactive investigation and comparison of movie scenes. The analysis process is depicted in Figure 2.11.

Data Types In the current implementation, the data types relevant for the analysis are the video itself, the corresponding subtitles, and the movie script. The first two data types are typically available to the analyst. Movie scripts are available for a wide range of popular movies. However, for specific movies, this data source might be not obtainable. In these cases, the approach could also be applied, but without valuable information about scene boundaries.

Data Pre-Processing As a pre-processing step, the data is analyzed automatically to summarize information for visualization. Video (e.g., shot detection) and text (e.g., extracting scenes from the script) are processed separately and fused in a consecutive processing step for alignment of scenes extracted from the script and the video.

Visual Elements The result of the pre-processing is a set of elements that are included in the visualization to provide answers to the relevant analysis aspects (W_1 – W_4). The video itself provides the most detailed level of information and should always be available in the visualization. The temporal dimension of the video is now segmented by scenes and can be represented by an appropriate visual metaphor. Extracted element tags (e.g., persons in a scene) convey relevant information about *who* (W_1) appeared *when* (W_4) in the movie. Semantic frames [113] are relevant because they provide information about *what* (W_2) and *where* (W_3) something happened.

Analytical Reasoning The extracted information from the pre-processing step (e.g., scenes, characters) is presented in a visual analytics environment that provides an overview of the data and allows the analyst to formulate search queries and annotate the results for an iterative extraction and documentation of insights. The possibility to search directly for extracted information (e.g., keywords) is included for single or combined features, or a similarity search on textual and visual features can be performed.

Knowledge Extraction By combining search queries and including the annotated results back in the analysis process, higher-level insights (e.g., identifying conflicts) can be derived that provide the analyst with detailed knowledge about the movie.

In the following, it is further outlined how the pre-processing of video and text data is performed, how the visual analytics environment is designed, and how the analytical reasoning process is supported by the approach.

2.3.3 Data Pre-Processing

The accessed data is split into the two main categories of image-based and text-based data. Optical flow calculation and shot detection are performed as described in Chapter 2.1.2. The text-based information requires additional processing to retrieve semantic content. The main motivation here is to fuse the data sources, harnessing the information from both sources to complement each other.

Element Extraction Text-based data comes from two different sources: the subtitles from the movie and a descriptive movie script. Figure 2.12 shows an example of how a movie script section with various structural elements looks like. The (A) scene heading describes the location and time of a scene, providing answers to the *where* and *when* questions. For example, the abbreviation “Int.” stands for interior and means that the scene acts in a closed room, “Ext.” marks outdoor scenes. The (B) action element describes the narrative description of the events of a scene. The next elements provide information about the (C) acting characters and their (D) dialogs. Optionally, there are

(A) INT. CAR
 (B) John drives the car and Barbara sits next to him.

 (C) JOHN
 (D) Two more hours until we arrive.

 John looks at his wristwatch and shows it to Barbara.

 BARBARA (O.S.) (E)
 Yes, but you know mother cannot drive this
 long to visit us.

Figure 2.12: Example of a movie script including: (A) heading, (B) narrative description, (C) character, (D) dialog, and (E) shot details.

extensions, placed after the character’s name, to indicate how the voice will be heard onscreen. For example, if a character is speaking as a voice-over, it would appear as “V.O.”, or if the character is not visible as (E) off-screen “O.S”. Movie scripts consist of plain text and there are no standardized formatting rules. However, they have a similar inherent structure, which allows automated processing of the script.

Text Alignment After the subtitles and the movie script have been successfully parsed, the next step is to synchronize both. Comparing the two text sources, two differences are apparent: the order of words or whole sentences can vary, and the script can contain scenes that do not exist in the movie and vice versa. Natural language processing methods are applied to extract the contained meta data and match between the text sources. For matching, each subtitle is assigned to a script dialog according to the highest similarity between text passages, based on string matching and the *term frequency–inverse document frequency* (*tf–idf*) weighting scheme [251].

Scene Detection With shot detection, the first abstraction layer is derived from the original video frames. The movie script contains the information which sentences belong to a scene but has no temporal information. The subtitles contain sentences with temporal information, but only for the time span they are displayed. Hence, the shots are summarized into scenes, based on the matches between subtitles and movie script, as depicted in Figure 2.13. The subtitle sentences are matched to the script so the first and last time stamp of a scene can be identified based on the shot boundaries.

Semantic Frame Analysis In cases where not the same words are chosen to express something semantically similar, an alternative measure is necessary. To achieve this, *semantic frames* [113] are applied. Semantic frames are a concept from linguistics that

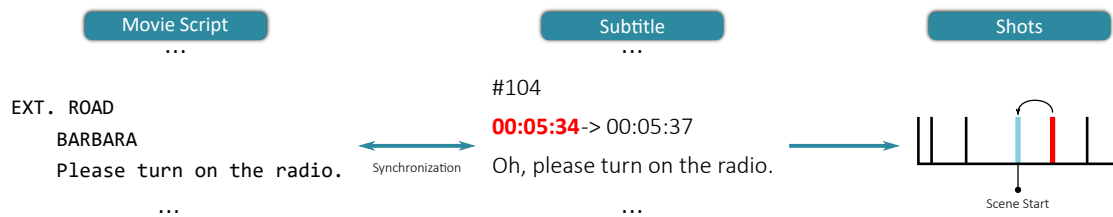


Figure 2.13: Data fusion for script/movie alignment: First, the script is matched with the subtitles. The corresponding time stamp of the subtitle in the video helps then identify the detected shot that marks the beginning of a scene.

describes prototypical situations described by words such as verbs, nouns, or adjectives. This approach provides a set of such situations (e.g., *shoot_projectiles*) for each scene. Comparing the overlap of two sets provides the second similarity measure.

The pre-processing provides a set of transformed data extracted from the raw data, the so-called data tables, according to the visualization reference model [70] (Chapter 2.2). For further details on the text processing, it is referred to the respective publication [25].

2.3.4 Analytics Environment

After the pre-processing stage, a set of *visual elements* is obtained that are incorporated in the final visualization: the video, detected scenes, element tags, and semantic frames. The main visualization consists of multi-layer timelines that represent this data on different levels of detail. This structure corresponds to the inherent hierarchy of a movie itself, from scenes to shots.

Visualization Design

The extracted elements are represented by individual timelines (Figure 2.14). A color and a label are assigned to each element. For the depiction of search results and annotations, this simple timeline visualization is proposed for two main reasons: (1) familiarity and (2) visual scalability. A timeline visualization is easy to interpret, as it is established in everyday life, e.g., in the form of schedules and requires only few screen space.

- **Layer 1** shows segmented timelines according to the length of the corresponding scenes. For scenes that could not be matched between the movie and the script, a dashed rectangle with a fixed width is included in the timeline to indicate that there is content relative to the previous and following scene matches.
- **Layer 2** can be displayed for multiple selected scenes. In general, the selection of a scene in Layer 1 creates a new layer that is stretched to screen width, enabling

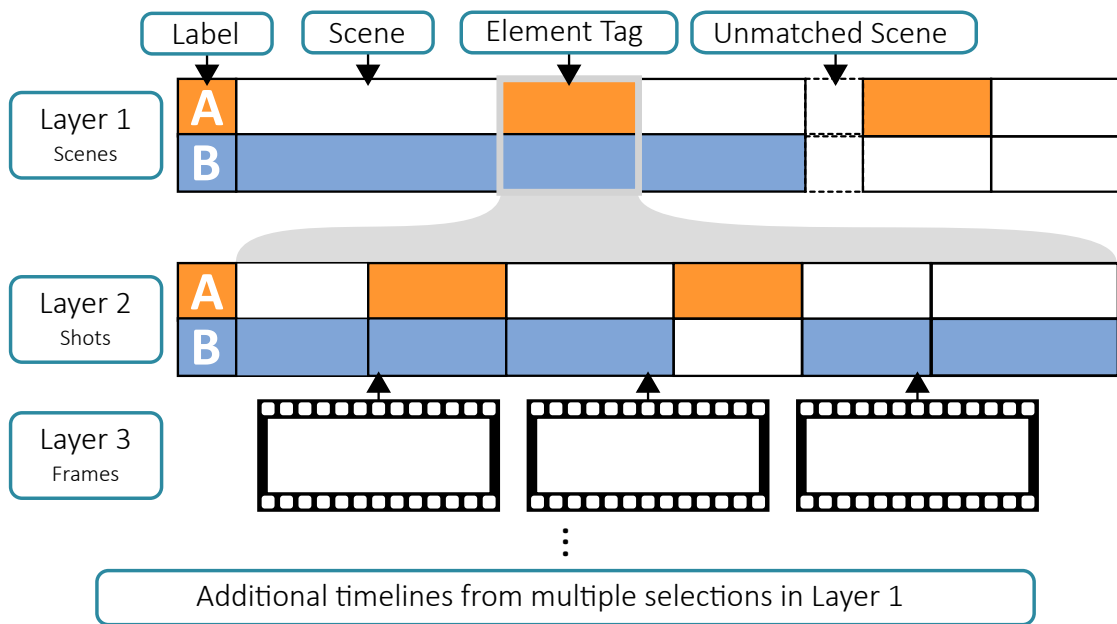


Figure 2.14: Multi-layered timelines: each labeled element’s appearance in scenes is depicted on an individual timeline in Layer 1. Selecting a scene creates a second layer that shows the shot-based appearance of elements. The third layer depicts movie content directly by corresponding frames. Selecting multiple scenes creates additional instances of Layer 2 to support comparisons.

a comparison of different scenes in relative time. In Layer 2, the segmentation of the timeline is based on the detected shot boundaries.

- **Layer 3** is an optional component that shows video content directly by example frames in a storyboard representation. The depicted frames are uniformly sampled from the corresponding time span.

In each timeline, element tags are displayed when the corresponding element appears in a scene. In the simplest representation, this can be achieved by coloring the corresponding part of the timeline. However, the visualization of timelines with rectangular shapes provides the possibility to encode additional information inside the scene rectangles. Figure 2.15 shows a set of possible visual encodings, suitable for numerous analysis tasks. Categorical tags can depict simple characteristics such as the occurrence of a person in a scene. Similarity tags depict the accordance of scenes with a selected one. Distribution tags depict quantities that may change over time, for example, the magnitude of motion over time. Event tags mark specific points in time when something happened (e.g., the beginning of a shooting).

The implemented prototype includes three visual encodings to depict different properties of the processed data:

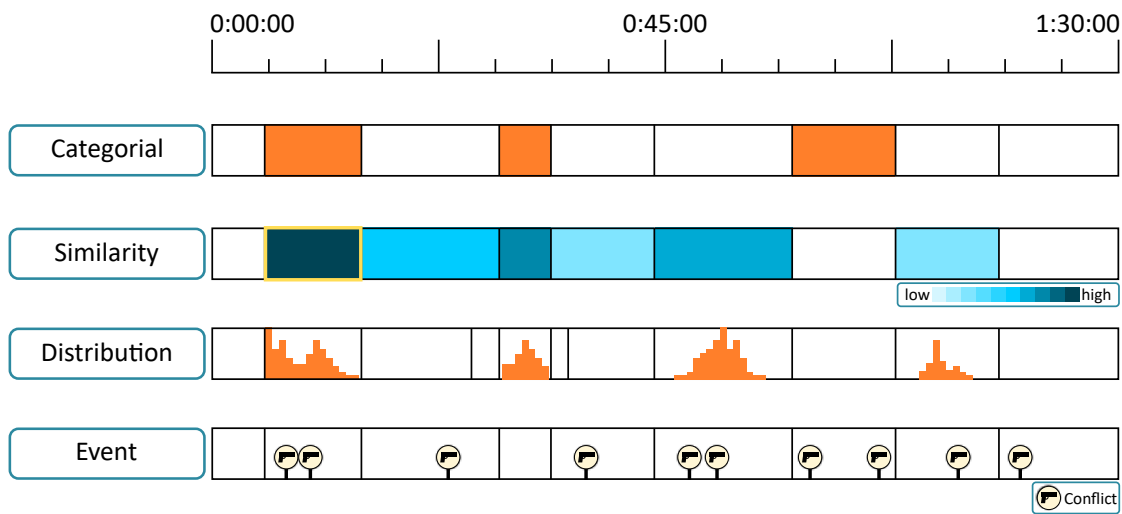


Figure 2.15: A tagline visualization offers numerous possible encodings of relevant data. Examples comprise categorical, similarity, distribution, and event tags.

Categorical/Occurrence Tags This kind of tag represents when an element appears in a scene. Additionally, the height is adjusted to depict the relative frequency of an element's appearance. Consequently, if a person in a scene has many lines to speak, the height of the occurrence tag will increase. If binary tagging is used, it shows whether an element appears, or not.

Similarity Tags Based on the described similarity metrics for motion and text (Chapter 2.3.3), the analyst can select a reference scene and compare it with the other scenes. In this case, a new timeline with the resulting normalized similarities will be created. For the depiction of the values, a sequential color map is applied.

Distribution Tags For the visualization of the extracted motion field, the average length, and direction of the motion vectors is calculated and the values are displayed on a separate timeline. The length of the motion vector is decoded by the height of the bar, direction by color. Figure 2.16 shows three examples of different camera motions. With this representation, similar panning and zooming motions of the camera can be identified visually in the timeline. Although the focus of this work is more on the semantic analysis of movie content, this feature is incorporated to provide a glimpse in further possibilities for movie analysis and how simple they can be integrated.

Implemented Framework

Figure 2.17 shows an overview of the resulting analytical environment. At the top, the [Ⓐ] multiple timelines with the different tags are displayed. Selected scenes are stacked

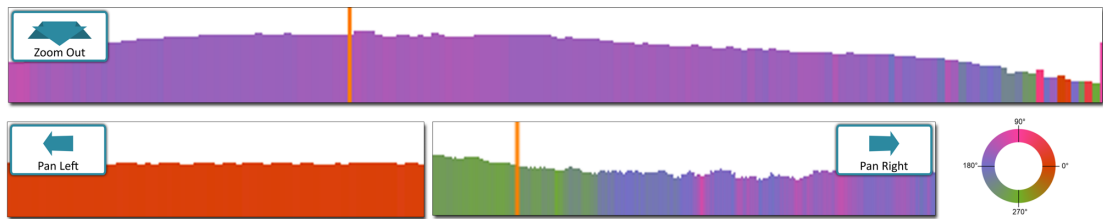


Figure 2.16: An example of a distribution/motion tag is the average motion over time calculated from the optical flow. The color coding for directions makes different camera motions visible.

below **Layer 1**. To improve the vertical scalability in order to compare more scenes at once, **Layer 2** can be reduced to a **(B)** compact representation and also **(C)** **Layer 3** can be removed. The **(D)** video player provides regular playback options for the video. Additionally, the analyst can create new elements for tagging, annotate the appropriate time spans, and write notes about specific findings. The **(E)** script viewer displays the text of the script with additional annotations of element labels and reference matches with the movie. In the list in the bottom-right corner, all **(F)** elements are shown. The analyst is free to select the elements to display as needed. Also, query results and manual annotations are appended to the list as new elements. In an additional list, all extracted semantic frames can be selected for a query search.

2.3.5 Analytical Reasoning

With the multi-layer timelines (Figure 2.17), the user has the advantage to investigate movie scenes in detail, while keeping the overview of all scenes available. However, to analyze and explore the structure of a movie, more interaction concepts are required than a static representation of scene timelines. Two main operations, *identify* and *compare*, have to be supported by the system for exploratory data analysis [41].

Identify In general, the identification of relevant elements for the questions (**W₁–W₄**) is important. Although a compact overview of individual elements is included, the analyst should be able to filter the data and group elements that belong together. To filter the data, queries on the extracted elements can be formulated. An additional search for keywords in the script is also possible. Query results are represented as new timelines integrated in the overview (**Layer 1**). By this approach, derived insight can be assessed for further analysis.

Compare The second important operation is the comparison of task-relevant scenes. Relations between scenes are typically investigated by similarities of certain aspects. Let us assume the analyst identifies an important scene and wants to find other scenes that are similar. In this case, an additional query dialog is provided that allows specifying

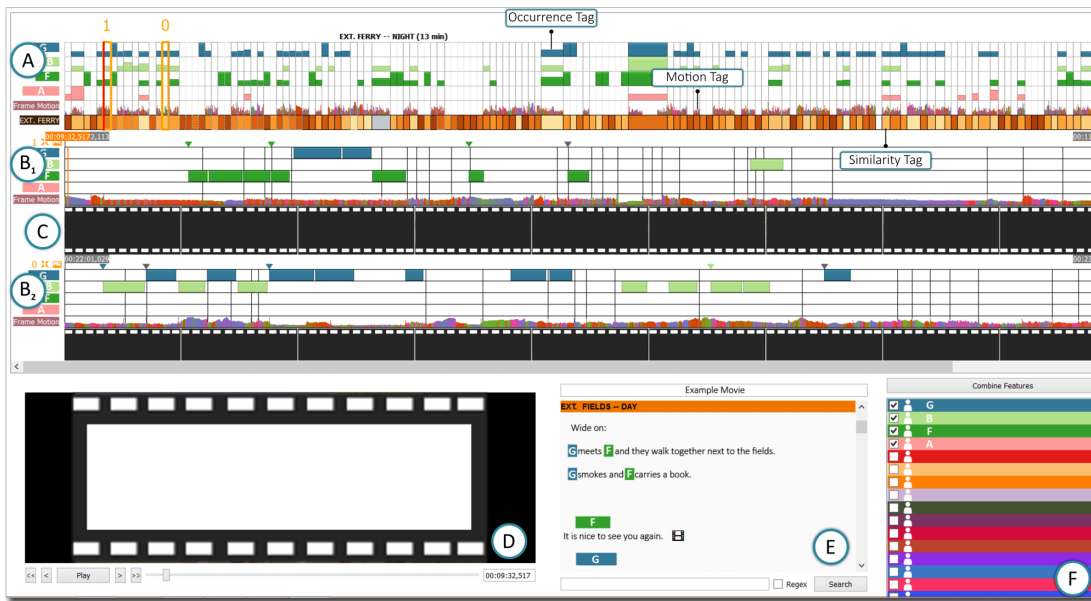


Figure 2.17: Analytics environment: (A) Timeline overview (**Layer 1**) on extracted movie scenes showing the appearance of individual element tags and the results from similarity queries. (B₁) & (B₂) Details about selected scenes (**Layer 2**) are displayed on separate timelines. (C) A storyboard representation of individual scenes (**Layer 3**) shows the movie content directly. Two linked views, (D) the video player and (E) the script viewer allow for detailed content analysis. (F) All defined elements are listed in the bottom-right corner.

the features on which the similarity between scenes can be determined. The dialog currently offers three methods to compare scenes. One of them compares scenes based on their motion histograms. The second method is based on the text similarity between the scripts and the subtitle. Finally, the third method compares the semantic frames of the scenes. Selecting multiple methods simultaneously aggregates the individual results with equal weights. Detailed comparisons of multiple scenes are also possible by multi-selections that will create new instances of **Layer 2**. The selected scenes can then be compared visually.

With the application of search queries and scene comparisons, the presented approach provides automatic mechanisms that aid to identify scenes of potential interest for a wide range of possible analysis questions. An analyst can formulate different queries, either based on specific knowledge (e.g., investigate all scenes with two important characters) or look for similarities, based on a reference (e.g., investigate all scenes that contain similar semantic frames as a selected scene). With the manual annotation function, the analyst can finally note derived insights from the assessed data on new timelines. An annotation can be performed by creating new element tags and marking

the corresponding time spans. Since this new tag is included in the element list, it can be used in new search queries for an interactive analysis loop (Figure 2.11). By this, higher-level insights and content summarizations can be extracted from the data.

This concludes the example of visual analytics for videos. By combining multiple established techniques from computer vision and natural language processing, an interactive visualization framework was created that supports the annotation and interpretation of movies. Further details and use cases are included in the publication [25].

User-Based Evaluation of Visualization

Effective visualization design is based on numerous factors. Along with best practice advice, many design aspects are derived from theoretical frameworks with respect to human perception and cognition. Nevertheless, evaluation is often necessary to provide empirical evidence for the effectiveness and efficiency of a visualization. Evaluation can be applied in different stages of development and deployment.


Typical scenarios in visualization research are divided into the evaluation of data analysis processes and the evaluation of the visualizations themselves [182]:

- **Evaluation of processes:** The goal of such evaluation procedures is to provide a holistic view of the experience and the role that visualizations play in an analysis scenario. These scenarios comprise understanding environments and work practices, evaluating visual data analysis and reasoning, evaluating communication through visualization, and evaluating collaborative data analysis.
- **Evaluation of the visualization:** For testing of design decisions and usability, as well as for comparison with other techniques, the visualization itself is evaluated. Typical scenarios in this category comprise evaluating user performance, user experience, and visualization algorithms.

In the context of this thesis, both types of evaluation were performed. For the evaluation of processes, often including questions about *how* and *why* some behavior is observed, eye tracking provides new possibilities to gain insights in contrast to classic performance studies. The evaluation of video visualizations in this thesis is focused on the techniques themselves and on user performance.

This chapter provides a general overview of user-based evaluation methodology for visualization (Chapter 3.1). Furthermore, evaluation techniques that are applied in the context of this thesis are discussed, i.e., quantitative performance studies (Chapter 3.2), interviews based on the repertory grid (Chapter 3.3), and eye tracking (Chapter 3.4).

This chapter is partly based on the following publications:

- K. Kurzhals, M. Burch, T. Pfeiffer, and D. Weiskopf. “Eye Tracking in Computer-Based Visualization”. In: *Computing in Science Engineering* 17.5 (2015), pp. 64–71 [14]
- K. Kurzhals, E. Çetinkaya, Y. Hu, W. Wang, and D. Weiskopf. “Close to the Action: Eye-Tracking Evaluation of Speaker-Following Subtitles”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2017, pp. 6559–6568 [27]
- K. Kurzhals, B. Fisher, M. Burch, and D. Weiskopf. “Eye Tracking Evaluation of Visual Analytics”. In: *Information Visualization* 15.4 (2016), pp. 340–358 [22]
- K. Kurzhals, B. Fisher, M. Burch, and D. Weiskopf. “Evaluating Visual Analytics with Eye Tracking”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2014, pp. 61–69 [21]
- K. Kurzhals and D. Weiskopf. “Exploring the Visualization Design Space with Repertory Grids”. In: *Computer Graphics Forum* 37.3 (2018), pp. 133–144 [19] 

3.1 Methodology

The evaluation of visualization design has become increasingly important, including different methodologies and guidelines when a method should be applied. Isenberg et al. [161] review evaluation methods applied in visualization research, based on a coding scheme by Lam et al. [182]. Sedlmair et al. [263] present a methodological framework and practical guidance for conducting design studies. Brehmer et al. [61] discuss pre-design empirical methods for information visualization. Munzner [214] provides a four-level nested model for visualization design and validation. She also discusses when different evaluation methods should be applied. This model was further extended by McKenna et al. [207] and Meyer et al. [211]. In all these models and taxonomies, quantitative and qualitative evaluation methods are listed to provide further insights into the visualization design process.

3.1.1 Quantitative Evaluation

A typical example of quantitative evaluation in visualization is the analysis of user performance. This performance is often measured through the number of errors and the completion time a participant needs to solve a specific task. By comparing visualization techniques, this measure provides empirical evidence if one visualization is better than another under the experimental conditions.

The general experimental procedure is depicted in Figure 3.1. First, the research question and appropriate hypotheses have to be defined. Independent variables mark the aspects that are studied and manipulated to inspect their influence on the results. The complexity of independent variables can become high, for example, in complex visual analytics applications. Hence, typical laboratory studies focus on a low number of dependent variables for precise results.

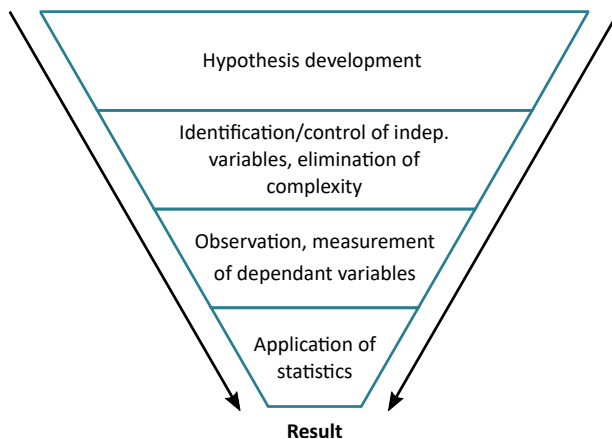


Figure 3.1: Experimental procedure for quantitative evaluation, according to Carpendale [72].

The validity of results can be compromised by issues concerning conclusion-, internal-, construct-, external-, and ecological validity [72]. Conclusion validity pertains the relation between independent and dependent variables. Typical issues arise from *Type I errors* (false negative) and *Type II errors* (false positive) considering a null hypothesis H_0 (no difference between two measures). Internal validity describes the causality of the relations. This is an important factor for correlation analysis where high correlations might occur without a reasonable relationship between the measures (e.g., high correlations between the stork population and human birth rate [205]). Construct- and external validity concern the generalizability with respect to the intended question (construct) and the applicability of the results to other groups/situations (external). The ecological validity of study results is about the relation between the experimental setting and a real-world application. One major issue of quantitative evaluation is the fact that complete validity for all of the mentioned aspects is often not possible. Experiments are usually designed to ensure conclusion validity by restricting the setting to a small number of controlled variables. This contradicts a real-world application in which numerous other factors can influence a study participant.

In this thesis, quantitative analysis is applied for traditional performance studies. For different video visualizations, participants identify specific search targets. Error rates and the time between onset of a target until reaction are measured. The user studies were conducted under laboratory conditions, excluding confounding factors in order to measure the best possible gain of the investigated visualizations. Additionally, eye-tracking metrics are applied for an extended analysis of how the participants' gaze behavior changes under different visualization conditions. Furthermore, not all aspects of visualization can be measured with quantitative methods. If the evaluation aims to better understand the user experience, subjective impressions, and analysis processes, qualitative methods provide additional means to be included in a user study.

Consequently, other factors have to be eliminated or kept constant. Dependent variables comprise observations and measurements. These variables are investigated for changes as a consequence of manipulations on the independent variables. The application of statistics provides further information on the influence of independent on dependent variables, for example, by statistical inference (i.e., finding significant differences between the results of two conditions) and correlation analysis.

3.1.2 Qualitative Evaluation

Qualitative evaluation comprises observation and interview techniques [72]. Observations are less obtrusive and often result in notes and recordings of a participant's behavior using a visualization. Interviews require the person leading the interview to interact with the participant, providing a more target-oriented method where questions can be asked directly, but possibly influencing the results. From the numerous methods for qualitative evaluation, the work in this thesis focuses on think-aloud protocols, questionnaires, and interviews. Qualitative evaluation is mainly conducted for expert feedback on implemented visualization techniques.

Think Aloud This method [105] encourages participants to verbalize their thoughts during the interaction with a visualization. Audio and written protocols are captured and can be annotated with an appropriate coding scheme to help identify common strategies or issues with the visualization. One issue with this method is that loud speaking while solving a task is an atypical situation for most participants and their workflow might be different to a real situation.

Questionnaires Questionnaires are capable of capturing subjective opinions on a topic [225]. Qualitative feedback is often collected by free-text forms, or with Likert scales [194]. The advantages of questionnaires are that for many visualization and usability-related questions, standardized questionnaires exist (e.g., the *Questionnaire for User Interface Satisfaction (QUIS)* [83]). Additionally, online surveys can be created to collect a large number of participants' opinions.

Expert Interviews Asking domain and visualization experts in an interview about their opinion can help to identify flaws that were not considered by the person who designed a visualization. As a specific type of interview, expert reviews [290] are often applied in which visualization experts evaluate a design based on heuristics.

In cases where free-text reports are the result of the evaluation procedure, further annotation of the data based on grounded theory [77] is necessary to identify commonalities between participants. For questionnaires with Likert scales, this scheme is predefined, but for all verbal statements recorded with these methods, one has to read through the protocols and annotate the text accordingly. Hence, an alternative approach is presented that provides qualitative research results in a structured, quantified form (Chapter 3.3). As an example of the application of multiple evaluation techniques, Chapter 6.2 describes a visual analytics approach for the annotation of eye-tracking data. The approach allows one to solve annotation tasks in multiple ways. With a combination of performance analysis, questionnaires, and think aloud, different analysis strategies and their efficiency are determined.

3.2 User Performance Studies

Evaluation based on user performance provides objective measures, typically in the form of error rates and completion times. According to Lam et al. [182], the two types of goals for such studies are: (1) Finding the limitations of visual perception and cognition for visual encodings or interaction techniques and (2) comparing different visualizations based on human performance. For the first case, experiments for the identification of *Just Noticeable Differences (JND)* [275] are a popular method to identify thresholds for visualization parameters that influence performance. For the second case, the user performance for the different visualizations is compared. As an example of typical user performance studies in controlled lab experiments, the video visualizations introduced in Chapter 2.2 are evaluated. In both cases, it is important to find out if the visualization impairs detection tasks, which are typically performed on such video material.



Figure 3.2: The cartoon character appears multiple times and participants have to confirm each detection by pressing a button.

User studies were conducted to compare how participants perform detection tasks with the different visualizations. The task for both studies is identical: A cartoon character (Figure 3.2) is edited into video material from surveillance cameras and participants have to search for this figure. In the fast-forward videos, the character appears as an individual, in the attention-guiding videos, the character fades in and out on existing persons and in cars.

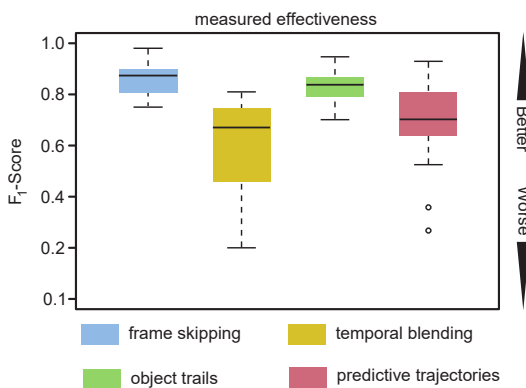


Figure 3.3: Performance for fast-forward visualizations [8].

Fast-Forward Video Visualization In the fast-forward study, cartoon figures appear at fast playback rates, exacerbating perfect detection rates for all video visualizations. Temporal blending and predictive trajectories result in detection rates significantly lower than with frame skipping and object trails (Figure 3.3). The results support the assumption that frame blending, often applied as an alternative to frame skipping, impairs search tasks for video surveillance. Although the detection rate for predictive trajectories is reduced, the subjective impression of perceived motion is better than with frame skipping.

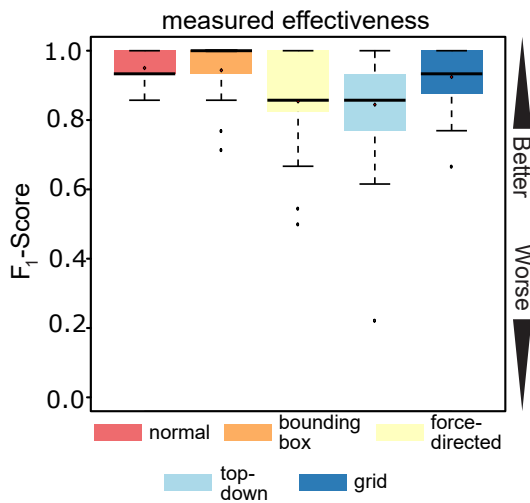


Figure 3.4: Performance for attention-guiding visualizations [12].

Attention-Guiding Video Visualization In contrast to the fast-forward videos, the stimuli are played back at regular speed. The cartoon character fades in and out, so participants have to overview all objects in the scene. Only the top-down approach performs significantly worse than the normal video representation (Figure 3.4). The performance in this task is of less priority because the distribution of gaze is the focus of this user study. The gaze distribution is measured with eye tracking and evaluated separately, see the respective publication [12] for details.

Both examples provide insights into how video visualizations influence object detection tasks. In cases where a technique impairs the task, a trade-off between performance and other advantages (e.g., an even distribution of attention) has to be made.

In addition to the performance analysis, the attention-guiding video visualizations aim to distribute the user's gaze more evenly between the appearing objects. This is measured with eye tracking. The bounding boxes, the top-down view, and the grid achieve a more even gaze distribution. To collect information about the subjective experience with the visualization, questionnaires were handed out. As an alternative qualitative evaluation step, the repertory grid poses a good means to assess the individual experience.

3.3 Repertory Grids for Visualization

As a contribution to extending the set of methodologies for qualitative evaluation in visualization research, the repertory grid is discussed in the following. The repertory grid is an interview technique with its origin in psychology. It allows researchers to quantify objective and subjective features in a setting that does not dictate specific terms for rating. The interviewee is free to formulate individual opinions in a structured grid. This chapter discusses the methodological approach of this technique, which has been applied in numerous research fields, but, to this point, is rarely utilized in the context of visualization design. The repertory grid technique is based on the personal construct theory developed by Kelly [172]. This theory essentially says that how a person construes the world depends on a large set of personal constructs that can be expressed with bipolar terms. With the repertory grid technique, individual constructs can be elicited and related to specific visualization features.

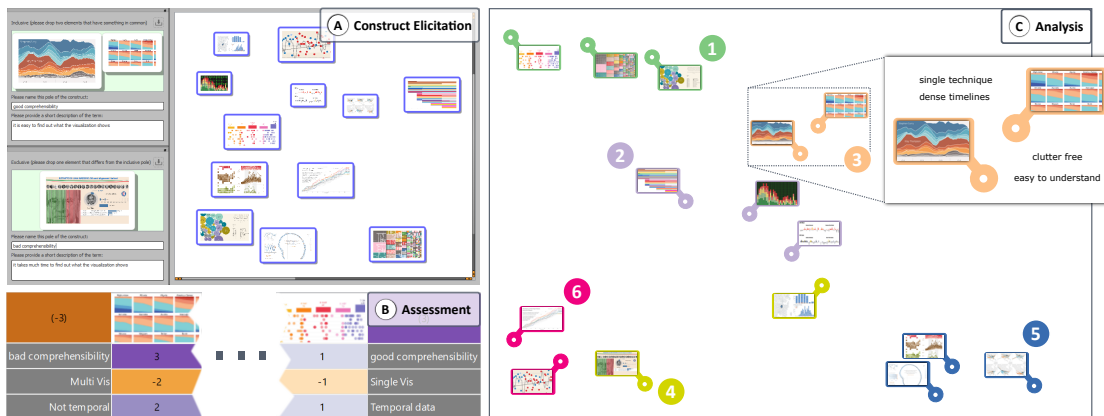


Figure 3.5: Repertory grid technique applied to information visualization: (A) Construct elicitation by formulating bipolar terms to describe the visualizations. (B) Each visualization is assessed according to the formulated terms. (C) Qualitative and quantitative analysis methods provide insights into important factors for visualization design.

For example, Figure 3.5 (A) shows a set of visualizations used in a showcase interview. The construct *good comprehensibility* – *bad comprehensibility* is elicited with two stream graphs (*good*) and one visualization containing several pictograms (*bad*). (B) After eliciting constructs, each visualization is assessed individually. (C) The resulting matrix can be analyzed, for example with clustering to identify commonalities between visualizations. Depending on the set of elements and the research questions, this technique can be applied in different scenarios to evaluate visualization design. This chapter first discusses the repertory grid and how it can be applied, followed by a general discussion of the technique in the context of visualization. Due to its maturity as a methodology and versatile applicability, the repertory grid has the potential to serve as a means of evaluating visualization during the design and the application stage.

3.3.1 The Repertory Grid Technique

The assumption of the personal construct theory is that every person creates “own ways of seeing the world” [172]. Construing the world is performed by building personal constructs. These constructs can be expressed on bipolar axes with opposing terms on both ends (e.g., *ugly* – *beautiful*), based on the assumption that whenever we affirm one thing, we simultaneously deny another thing. Thus, an object cannot be beautiful and ugly at the same time. To elicit these constructs, the theory is accompanied by a methodological procedure (Figure 3.6). The repertory grid is a form of a structured interview [115] that can help explore and formalize another person’s construct system. In a conversation about the investigated topic, the interviewee formulates constructs to assess the topic, while the interviewer tries to understand what the construct terms

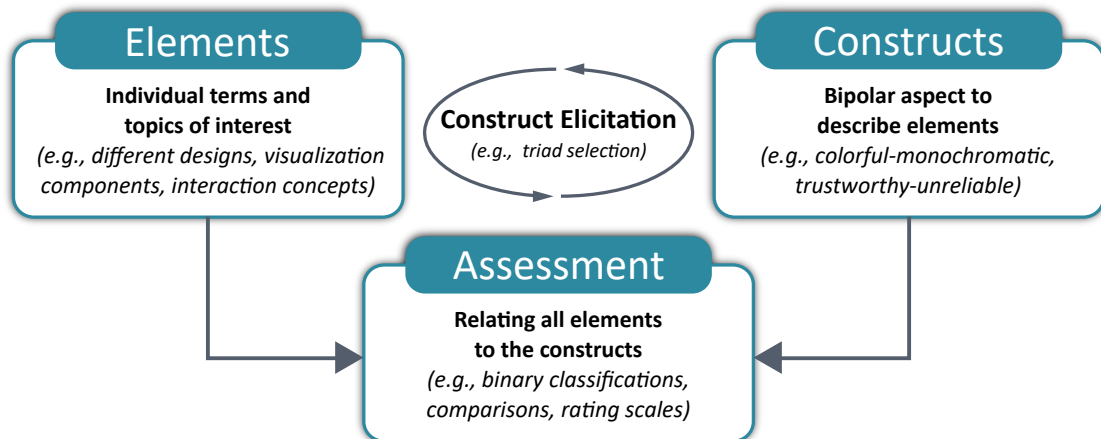


Figure 3.6: Procedure of the repertory grid technique. From a set of elements, constructs are elicited and assessed on a scale.

mean to the interviewee. Initially developed for psychotherapeutic application, this technique has been extended to numerous research fields since.

Elements According to Kelly [172], “the things or events which are abstracted by a construct are called elements”. In the specific case of visualization, these can be different types of visualizations, components of a visualization, or interaction techniques. The interviewer either provides elements or the interviewee formulates them. Which approach fits best depends on the research question. For an assessment of different design approaches for a specific task, the interviewer provides the elements. For the general exploration of design possibilities, questions are provided (e.g., “*how would you visualize multivariate data?*”) and the interviewee decides on the elements.

Constructs In other approaches such as questionnaires or interviews with predefined questions, the interviewer’s assumptions influence the results. Eliciting constructs from the interviewee provides the possibility to identify aspects that are not expected by the interviewer. Established methods for elicitation are the selection of dyads or triads. Dyads require a selection of two elements and the interviewee states how these elements differ. For the triad approach, three elements are presented to the interviewee and (s)he has to name some aspect that two of them have in common. This is the term for one pole of the construct. For the other pole, the interviewer either asks what makes the third element different or asks for the opposite of the stated aspect. In general, the following assumptions are made for elicited constructs [115]: (1) They should be permeable, this means being applicable to new elements, as well as to the elements from which the construct has been elicited. (2) Pre-existing constructs with a certain degree of permanence will mainly be used; occasionally, new constructs could arise during the

elicitation process. (3) Constructs should be labeled with communicable verbalizations; the interviewer should discuss the meaning of labels with the interviewee without implying specific answers.

Element Assessment Elicited constructs are already interesting to identify important aspects of visualization, but the repertory grid further provides the possibility to relate all investigated elements to the constructs. Different methods have been proposed, most common are binary assignments of elements to one of the poles of a construct, ranking, and rating of elements. For visualization, binary classifications are appropriate for many aspects (e.g., *temporal* – *nontemporal data*), but some constructs also require a more differentiated rating of elements (e.g., *aesthetic* – *ugly*).

3.3.2 Related Work

Since Kelly's introduction of the methodology, the repertory grid interview was modified and applied in many different application scenarios. Apart from its use in psychotherapy, it was applied, for example, to management studies [74], product evaluation [117, 148], software engineering [94, 101, 286, 287], information systems [208, 280], and design studies in human-computer interaction (HCI). Since the latter topic is most similar to visualization, the discussion is focused on related work in this context.

Repertory Grids in Visualization

Hogan et al. [147] discuss the *elicitation interview*, a qualitative technique for a non-inductive but directive interview approach that requires additional coding (e.g., based on grounded theory [88]). The important difference to the repertory grid is that the elicitation interview aims to describe subjective aspects in the experience with a visualization, explicitly avoiding judgments and rationalizations, whereas the repertory grid technique provides a quantified representation of the relation between elicited constructs and the investigated visualization elements. Other comparison methods for visualization techniques apply a ranking of elements according to predefined criteria (e.g., Lawonn et al. [183]). Similarly, a ranking scheme can be incorporated for repertory grids by adjusting the assessment phase.

There is some work that either discusses repertory grids as a possible evaluation method or applies it in a visualization context. Mayr et al. [206] discuss measures and evaluation procedures for mental models. They list the repertory grid, together with sketching and concept maps, as suitable for understanding content, structure, and coherence of mental maps. Compared to the other mentioned techniques, the repertory grid is more structured and provides quantifiable results that are easier to compare between participants. Meng [210] mentions the repertory grid as a means to capture personal preferences and personally perceived characteristics of geo-visualizations. McNamara

and Orlando-Gay [209] apply comparison-contrast debriefing questions derived from the repertory grid method to investigate how intelligence analysts analyze documents. Baum [45] presents the application of the repertory grid in the context of aesthetics criteria for software visualizations. The author focuses on the qualitative summarization of constructs in a categorization scheme for software structures. Ab Aziz [32] evaluates the user experience of visualizations for navigation purposes. She argues that analyzed grid data can provide groupings of constructs for the classification of visualization features, for example, to derive design guidelines.

The above papers are the first concrete examples of how the repertory grid can be applied to visualization techniques. In contrast, this thesis discusses the general considerations of the repertory grid's application for visualization in detail, as well as the possible application scenarios. Furthermore, a general comparison with other established methods for qualitative research is provided.

Repertory Grids in Other Research Fields

Hassenzahl et al. discuss and apply the repertory grid technique for the evaluation of the parallel design of prototypical interfaces [139] and websites [138]. Van Gennip et al. [123] investigate the design space of technologies for supporting remembering, Môtus et al. [199] the aesthetics of interaction, and Kwak et al. [181] the design space of shape-changing interfaces. Fallman and Waterworth [108, 109] examine the user experience of using mobile information technology. Hogan and Hornecker [148] propose a blended approach for repertory grids with focus groups, comparing visual, auditory, and haptic interfaces. The authors focus on the categorization of elicited constructs and establish clustering and projection methods to investigate the data. This categorization method is adopted in this work for visualization-specific context, identifying overlaps and differences between categories. This thesis also discusses how to further evaluate the data with descriptive statistics and visualization-specific modifications of the method.

3.3.3 Visualization-Specific Requirements

Given the assumption that elements and constructs have a range of convenience [172], it has to be considered how familiar interviewees are with a specific visualization. If a person does not understand how a visualization displays underlying data, certain constructs will not be applicable to elements or constructs will become superficial. Therefore, an initial training phase for the applied visualization elements is important, as well as a documentation of the interviewee's subjective assessment of understanding.

Visualization inherently requires the inclusion of visual aspects in the interview. Abstract concepts (e.g., multi-dimensional data representation) can be applied in the context of visualization, but often the interviewee will require a visual representation

of an element. If interaction and dynamically changing components are involved, a simple textual representation is cumbersome to convey the required information about elements to elicit meaningful constructs. Consequently, a visual interface for the conduction of a grid interview was developed to provide visual representations of the elements, help with data collection, and provide grid data ready for analysis.

The random selection of elements for construct elicitation is also applicable to visualization. However, especially for the exploration of the design space, it is important that all included elements are involved in the elicitation phase. As systematic randomization of dyad or triad combinations would require far too many participants, a modification of the full context form [115] is suggested where all elements are available and the interviewee can decide which elements to pick for a construct. With this approach, counting the frequency of an element's application provides potential information about its significance (*Was an element frequently involved in construct elicitation? Was it related to positive or negative terms?*). In order to cover all elements, the visual interface indicates which elements have already been used and highlights unused elements.

3.3.4 How to Conduct the Interview

Conducting the grid interview can be learned quite quickly. Important steps are outlined in Figure 3.7 for the conduction, analysis, and dissemination of the interview. Similar to Hogan et al.'s [147] description for the elicitation interview, five steps necessary to conduct and analyze a repertory grid can be formulated. Furthermore, it is discussed how documentation, computer support, and the interviewer influence individual steps.

Informed Consent

As with any other user-centered evaluation, participants have to be informed about the conditions, restrictions, and confidentiality of the procedure. A static document providing everyone with identical information is required.

Task Description, Tutorial

It is also important to make the interviewee familiar with the procedure, explaining it in detail and how data will be acquired. If examples are provided, it is possible that these examples might influence the type of elicited constructs [115]. Here, the interviewer also has to specify the research question for the elicitation procedure. If different visualizations are used, especially when comparing visualizations containing multiple views, some time to familiarize the interviewee with the elements is necessary. Because all information has to be provided consistently between interviewees, a static document is the preferred means of communication.

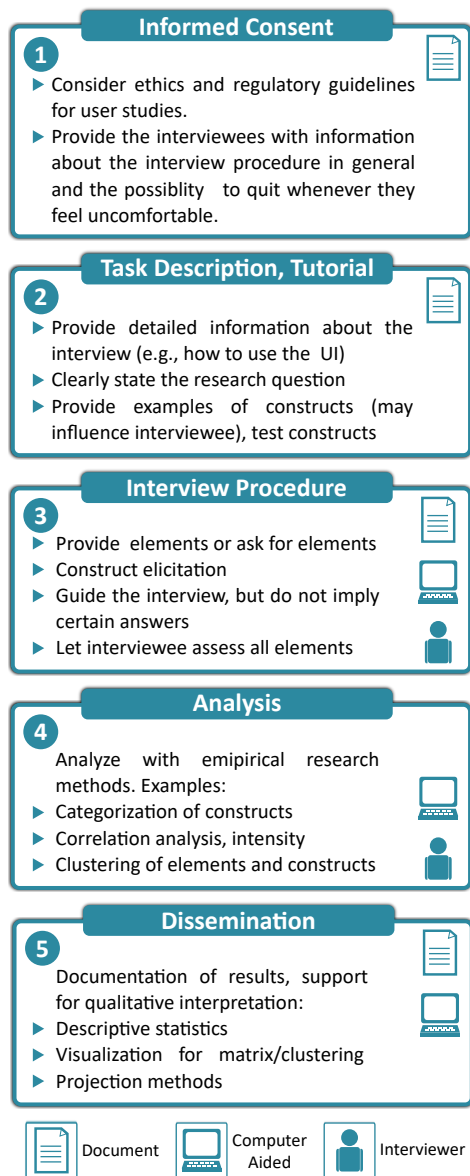


Figure 3.7: Important steps to conduct, analyze, and disseminate a repertory grid. The icons indicate which steps are conducted by a protocol document, with computational/interactive aid, and with guidance from the interviewer.

Interview Procedure

The repertory grid technique can be applied according to the elicitation and assessment procedure. The goal of the interview is to understand how the interviewee construes the topic. Interviewees are the experts for their subjective point of view and interviewers are the experts for the methodological procedure [116]. Therefore, the interviewer has to guide the interviewee, help elicit constructs by reminding about the rules, and ask for the meaning of constructs. The interview procedure is a social situation that requires the interviewer to comply with some rules to guarantee the freedom of articulation for the interviewee [116]:

- ▶ **Avoid contentual judgment:** Negative and positive assessment of stated constructs (*This is a very nice/bad term...*) does not appertain to the interviewer and has to be avoided.
- ▶ **Avoid surrogate wording:** Providing alternative terms (*Did you mean?*) undermines the expert role of the interviewee. Instead, the interviewer can ask for examples to help articulate.
- ▶ **Be open for corrections:** The articulation should be subjectively satisfying for the interviewee. Different tryouts and revisions of construct terms should be possible.
- ▶ **Adjust to the tempo of the interviewee:** The time, especially during the elicitation, can vary between interviewees. Some constructs come up spontaneously, others require time to think.
- ▶ **Adjust to the mood of the interviewee:** Exhaustion, tiredness, and a changing state of concentration during the interview might require some breaks the interviewer should be aware of.

Furthermore, the interviewer asks for the specific meaning of terms to understand their subjective meaning to the interviewee. To help elicit new constructs, *laddering* [115] can be applied. With this approach, the interviewer asks the interviewee to further evaluate on a pole of a construct (*You have formulated the construct “beautiful – ugly”. What exactly makes a visualization beautiful/ugly for you?*), which leads to new constructs. Asking for new constructs that explain *why* a term was chosen provides deeper insight into potential issues of a visualization.

A grid interview can be performed simply with a pen and paper. An interactive, digital version has some advantages over its analog counterpart: The collection and analysis of data is simplified, measures such as completion times and use frequencies for individual elements can be captured, information about the elements can be integrated, a visualization can be explored in detail, and constructs, as well as ratings, can be edited without effort.

Some software suites (e.g., GridSuite¹, Idiogrid²) have been introduced to perform the interview digitally. However, these applications are often only commercially available or support a subset of the functionalities required for the application to visualization. Hence, an open-source visual interface is provided, similar to a paper-based version of the test including the advantages mentioned before.³

Figure 3.8 shows how to apply the repertory grid with the developed interface. First, the interviewee is asked to select two visualizations that have something in common by drag-and-drop interaction (Figure 3.8a). Then, one visualization that differs is dropped in the second area (Figure 3.8b). For both poles, verbalizations are entered in a text field. The second text field is for documenting the meaning of the terms. The interviewer asks specifically for explanations (*What does this term mean to you?*) that help understand the subjective view of the interviewee. The repertory grid with all elicited constructs is displayed on demand for the rating of elements (Figure 3.8c).

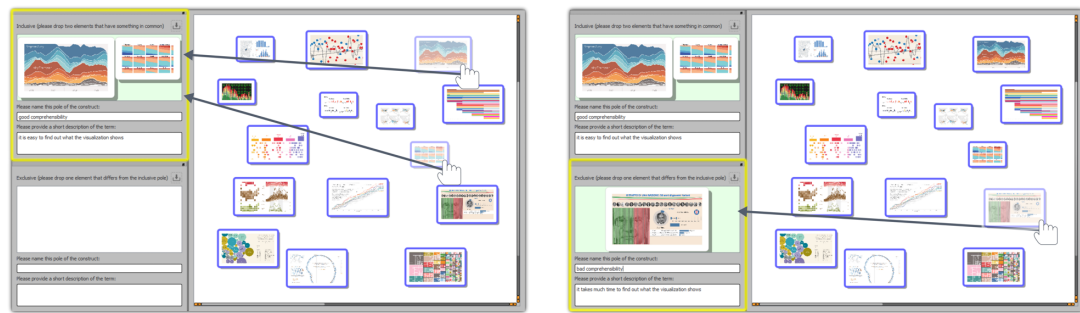
Analysis

The analysis of the captured grid data can be performed qualitatively and quantitatively. One important first step is the summarization of the dimensions describing the design space. Constructs can be categorized and counted to provide an overview of the dimensions and their importance. To this point, this procedure requires the interviewer—or optimally multiple analysts for inductive coding [284]—to interpret the similarities between constructs. Here, the descriptions for individual terms are crucial to identify how different interviewees interpreted the visualizations.

¹ <http://www.gridsuite.de>, last checked: October 13, 2018

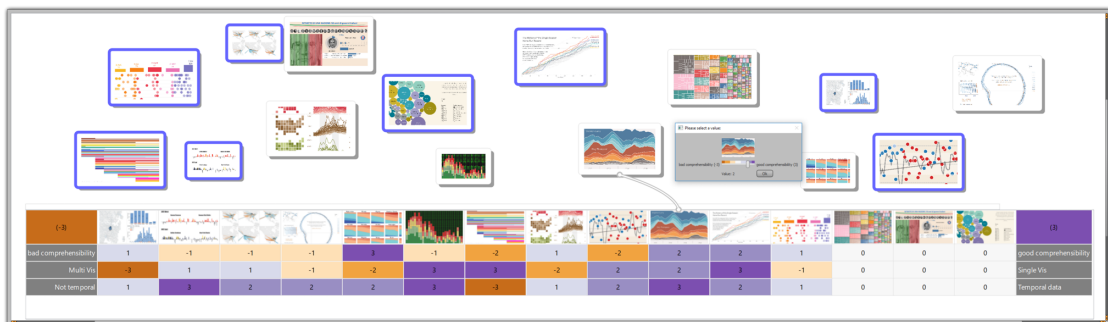
² <http://www.idiogrid.com>, last checked: October 13, 2018

³ <http://go.visus.uni-stuttgart.de/repgrid>, last checked: October 13, 2018



(a) Selection of two elements that have something in common.

(b) Selection of one element different from the previously selected.



(c) The repertory grid can be displayed on demand, elements are rated according to the defined constructs.

Figure 3.8: Visual interface for conducting a repertory grid interview.

Descriptive statistics on the grid data can be generated automatically and typical matrix analysis methods such as clustering and correlation analysis can be applied. *Multidimensional scaling* [170] and *Principal Component Analysis (PCA)* [230] are established methods to help interpret construct dimensions. Further information can be found in Fransella et al. [115] and Fromm [116], providing a general overview and discussion of how to interpret grid data.

Dissemination

For the dissemination of the results, documentation is the main way of distribution. Categorization tables, matrix visualizations, and two-dimensional projection methods are established methods to support statistics. Statistical software like **R**⁴ provides basic visualization techniques for repertory grids. Since the resulting data is a quantity matrix, additional visualization techniques for such data could be potentially helpful to communicate insights. For example, Onoue et al. [223, 224] present a graph layout that is suitable to analyze the hierarchical structure of constructs resulting from laddering.

⁴ <https://www.r-project.org/>, last checked: October 13, 2018

Table 3.1: Comparison of qualitative evaluation methods. The criteria are rated as (○) not supported, (◐) partially supported, and (●) supported for each method accordingly.

	No Extra Coding	Quantitative Analysis	Comparability	Objectivity	Exploration
Observations	○	○	◐	◐	●
Think Aloud	○	○	◐	◐	●
Questionnaire	●	●	●	●	○
Expert Review	○	○	◐	●	○
Repertory Grid	◐	●	◐	◐	●

A showcase interview that exemplifies the elicitation and analysis of a repertory grid is discussed in the respective publication [19]. It is compared how experts and non-experts describe important visualization aspects based on objective and subjective criteria.

3.3.5 Comparison with other Qualitative Methods

Qualitative studies are often conducted as part of the design process to derive design and evaluative criteria. Based on the qualitative evaluation approaches discussed by Carpendale [72] (Chapter 3.1.2), the repertory grid is compared with methods typically applied in experiments with a similar purpose (Table 3.1). The criteria consider how coding is handled, if direct quantitative analysis is possible from the data, how comparable results between participants are, to which degree objectivity of the results is possible, and if an exploration of the design space is supported by the method.

No Extra Coding Post-test open coding is often necessary to systematically categorize the content of observations or verbal statements (e.g., statements from a think-aloud session). This is often one of the most time-consuming parts in the analysis procedure. In comparison, a questionnaire based on Likert scales is designed with a specific coding scheme in advance, restricting answers to the determined aspects. The main advantage of the repertory grid lies in the coding scheme that is directly established during the test. Structuring expressions according to the rules of the method, provides constructs that can be summarized into concepts, or construct categories, more easily than interpretations of verbal statements.

Quantitative Analysis Most of the qualitative approaches require an additional coding phase to structure and count specific aspects of the participants' statements. The structure of the repertory grid provides quantitative results. Not only countable

constructs but also element assessments from participants are immediately available because the rating scheme is often identical with Likert scales in a questionnaire.

Comparability The standardization of answers, necessary for a direct comparison between participants, is only attainable with a questionnaire. All other methods require coding of answers to compare them. In the repertory grid, coding is performed by the participants and the interviewer identifies groups of constructs. With the other approaches, protocols have to be interpreted and coded by the interviewer to achieve a similar degree of comparability.

Objectivity Observations, think-aloud protocols, and repertory grids include subjective results, resulting from the interviewee's statements and the interviewer's interpretations. With the use of heuristics, derived from studies and expert knowledge, objectivity can be increased, primarily in questionnaires and expert reviews.

Exploration The detection of unexpected aspects of a visualization is an important feature for the exploration of the design space. In comparison to questionnaires and expert reviews, the other techniques support this exploration by putting fewer restrictions on the terms how participants can express their opinions.

Although a questionnaire supports the majority of the mentioned aspects (Table 3.1), it lacks options for exploration. In practice, multiple techniques (e.g., think aloud and questionnaires) are often applied together to combine the presented aspects. However, due to their different structures, merging the resulting findings requires additional effort. The repertory grid provides comprehensive results for all mentioned aspects. Additionally, the elicited constructs can be applied to derive new elements for a questionnaire. In summary, the repertory grid introduces a structured scheme for the results that other methods can only achieve by extensive coding of observations or recordings after the experiment. The element assessment with respect to the elicited constructs allows quantitative analysis that is otherwise only achievable by questionnaires with Likert scales. The interviewee is free to formulate individual opinions, which supports the exploration of design aspects without the restrictive properties of the questionnaire. Hence, the advantages of the repertory grid render it a versatile tool for the application in the design process.

3.3.6 Application Scenarios

Munzner [214] describes the design process by four nested layers: domain problem characterization, data/operation abstraction design, encoding/interaction technique design, and algorithm design. Arising threats can be validated by immediate and

downstream approaches. The repertory grid can be integrated into multiple stages of this model, for exploration and validation purposes, except for algorithm design.

Domain Problem Characterization A new visualization has to address the appropriate problem. As mentioned by Munzner, immediate validations are mostly qualitative, including semi-structured interviews and grounded evaluation [160]. The repertory grid interview fits in this category and has also been applied for requirements analysis in other fields [219, 220]. Possible scenarios could provide a set of important aspects as elements, or let the target user define important elements that specify the problem. For example, the interviewee is asked to state a set of important analysis tasks a visualization should support for a dataset. Incorporating these tasks as elements in the grid interview might provide further insights into their relevance for the domain problem. Downstream validation is achieved by letting the user rate the same grid from the immediate phase after the deployment of the visualization.

Data/Operation Abstraction Design Identifying if the chosen operations and data types solve a problem properly is, according to Munzner, mainly restricted to downstream validation because target users must test a system first. The repertory grid can be applied as a downstream validation, helping the user structure experiences with a system, and it can help formulate and quantify insights for a specific analysis task (e.g., using different levels of data aggregation or filtered datasets as elements).

Encoding/Interaction Technique Design Visual encoding is the stage with the most useful application of the repertory grid because it supports expert reviews and decisions based on guidelines. Different visualization designs can be used as elements and design guidelines formulated as constructs to help a visualization designer justify the decisions in a structured, replicable way. As another example, different visualization designs can be compared with respect to their suitability to solve the identified relevant analysis tasks from the domain problem characterization stage.

To this point, the focus was mainly on the investigation of commonalities of the interpretation of visualization design between participants. However, in cases where multiple techniques can be applied to solve a problem, individual differences in experience and other factors exacerbate decisions that declare one visualization technique as generally better than another. Hence, an application scenario respecting the individual preferences of a user is worthwhile inspecting.

Visualization Recommendation System If the repertory grid is regarded as the personality test it originally was, the application as a recommendation system for visualization is conceivable. For a set of visualization designs and an analysis task,

individual preferences might be different. One user prefers *Scatterplot Matrices (SPLOM)*, the second parallel coordinates, the third a glyph approach, and so on. If a pool of constructs is available from earlier interviews, the user can either perform a rating of the grid or select personally important aspects to identify the best-suited visualization. Although for some users this might be a clear choice in the first place, the repertory grid helps them structure the reasons why they prefer one visualization over another.

In summary, the methodological approach of the repertory grid interview provides new means for evaluation in the context of visualization. This approach can be applied in multiple stages of the design and deployment of new visualization techniques. Due to the relatively small number of required interviewees to cover the majority of constructs of a domain, the repertory grid is suitable to evaluate rapid prototypes and design concepts, but it is also useful in later stages of evaluation. The interview can provide insight into possible design flaws, acceptance of techniques, and it helps explore the design space for specific visualization problems. The individual freedom of interviewees to formulate constructs can reveal potential objective and subjective factors that should be considered when designing a visualization. Elicited constructs can be used to design questionnaires for new studies on the topic.

3.4 Eye Tracking for Visualization

In addition to the established methods for quantitative and qualitative evaluation, the inclusion of eye tracking became popular over the years. Known from applications in psychology and marketing [99], the eye-tracking methodology was adapted in other research areas such as visualization. As mentioned in the beginning of this chapter, evaluation beyond regular performance analysis requires methods that provide more information on task-solving processes during the use of visualization. Statistics of measured gaze data provide detailed information about which part of a visualization was investigated for how long, how often participants switched between different areas, and if important parts have been ignored. Furthermore, investigating recorded gaze data as spatio-temporal sequences provides insights into visual task-solving strategies which is otherwise hard to achieve.

3.4.1 Foundations of Eye Tracking

One important assumption for the analysis of gaze data is the eye-mind hypothesis [169] derived from experiments on reading behavior. It states that what people look at has a strong correlation to what they think of. While this assumption has some limitations, e.g., for cognitive retrieval processes [37], it is commonly accepted for most

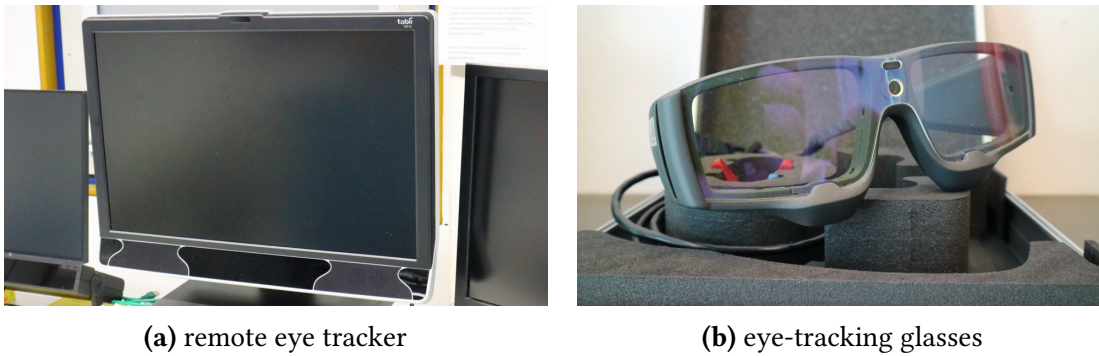


Figure 3.9: Eye-tracking devices used in this thesis: (a) remote eye tracker for showing stimuli with constant conditions; (b) eye-tracking glasses for mobile applications.

application scenarios. Hence, detected fixations in gaze data are often interpreted as an approximation for visual attention.

The recording of gaze data, typically accompanied by mapping the gaze to a specific coordinate system (e.g., monitor coordinates) is achieved by different devices, nowadays often with video-based detection systems. All these systems have in common that a video image of the eye is recorded and computer vision is applied to detect the pupil and the orientation of the eye. Figure 3.9 depicts two video-based devices that are used to record the data for this thesis:

- A remote eye tracker (Figure 3.9a) attached to a monitor records gaze data for scenarios in which participants watch or interact with stimuli under controlled conditions. For the data presented in this thesis, remote eye tracking is mainly applied to present videos. Data recorded with such an experimental setting results in an easier comparison between participants than with data from mobile devices.
- Mobile eye-tracking glasses (Figure 3.9b) count into the category of head-mounted devices. In addition to cameras that recorded the eyes of a participant, a world camera covers the current field of view onto which the gaze can be mapped. Wearable devices ease the application for unconstrained real-world experiments, but the data is difficult to analyze due to the high variability between recordings.

Next, some basic terminology is introduced, before it is discussed how eye tracking is included in evaluation methodology and what it is used for in visualization research.

Terminology

Recorded and mapped raw gaze data is further processed to detect a set of common types of eye movements: *fixations*, *saccades*, *glissades*, *smooth pursuit*, *microsaccades*, *tremor*, and *drift* [149]:

Fixations A fixation summarizes a time span (200–300 ms) when the eye remains relatively still. Although micro-movements happen, the respective gaze points are usually summarized for this time span. Fixation detection is supported by most software suites provided by the hardware vendor. Detection algorithms are often based on spatial or velocity thresholds [253].

Saccades Between consecutive fixations, the eye performs rapid jumps to adjust the gaze to a new position. It is assumed that people are temporarily blind during a saccade. Post-saccadic eye movements that adjust the eye to the target are called *glissades*. The typical duration of a saccade is between 30–80 ms. Due to their increased speed, saccades are easier to detect with high sampling rates of the recording device.

Smooth Pursuit Smooth pursuits happen in situations when the eye follows a moving stimulus, for example, an object in a video. The detection of smooth pursuits for eye-tracking hardware with different sampling rates is still an important research topic. The investigated data in this thesis recorded from video is visualized with raw data, if not stated otherwise, to maintain the visual structures of smooth pursuits.

Microsaccades, *tremor*, and *drifts* are summarized as micro-movements that are not considered in the context of this thesis. Furthermore, some additional terms are repeatedly used in the following chapters:

Scanpath In the following chapters, the full sequence of fixations and saccades is referred to as scanpath. Since the applied detection algorithms are based on the detection of fixations, saccades are approximated by the spatio-temporal gaps between fixations.

Area of Interest Semantic regions on a stimulus can be annotated as *Areas of Interest (AOIs)*. With AOIs available, comparisons between participants become easier because the scanpath is further abstracted to sequential visits on different AOIs. The annotation of AOIs is a time-consuming step in the analysis process that requires human input. An automatic solution of this task is only possible in a few scenarios (e.g., in marker-based environments [232, 233]).

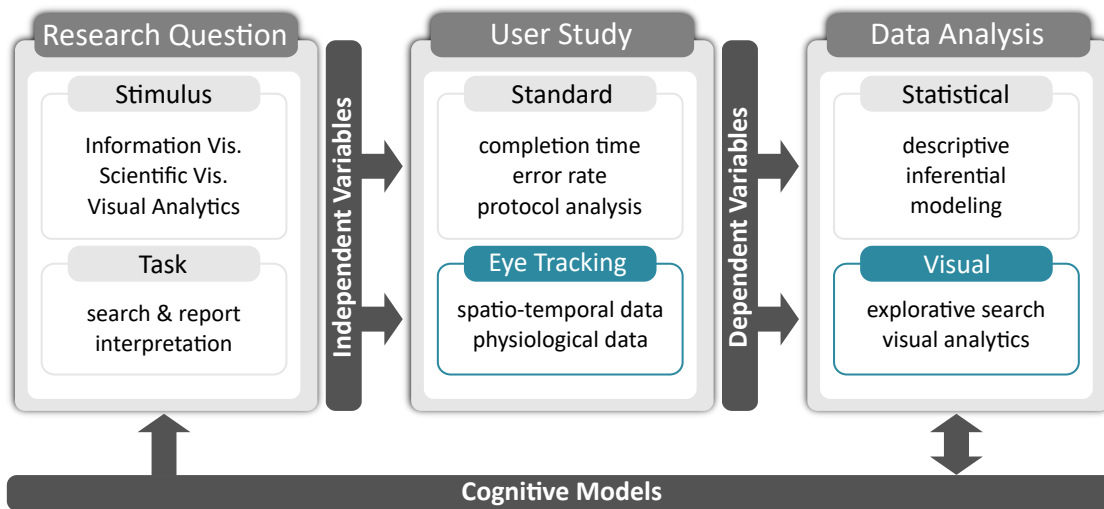


Figure 3.10: Evaluation pipeline depicting a regular laboratory experiment (gray) with visualization. Eye tracking is easily integrable into the procedure (blue).

Eye tracking can be included in many established evaluation procedures. For standard desktop visualization systems, a remote eye tracker requires only minor effort to calibrate before the experiment. Even collaborative scenarios can be solved by using multiple head-mounted eye trackers. In the following, the integration of eye tracking in evaluation methodology is further discussed.

3.4.2 Including Eye Tracking in Evaluation Methodology

Embedding eye tracking into existing evaluation procedures can be achieved without significant changes on the procedure itself. A typical user study for visualization techniques is described by the pipeline in Figure 3.10 (gray parts). A controlled laboratory experiment is assumed, even though many aspects carry over to other variants of user studies. The visual stimuli and choice of tasks serve as independent variables of the study. In this context, different visualization techniques and/or variations of one technique provide the basis for the visual stimuli. The task often requires the user to search and report certain aspects, or interpret the stimulus. The performance with the task is assessed in the form of dependent variables. The data acquired through the dependent variables is analyzed, eventually leading to conclusions regarding the study.

With eye tracking, the evaluation pipeline is extended (Figure 3.10, blue). The recorded gaze data provides additional dependent variables, in particular, spatio-temporal information about the participant's viewing behavior or physiological data by the pupil diameter, which can be an indicator of cognitive load [36, 177]. In visualization and visual analytics systems, the analyst is typically confronted by a difficult task that consists

of several stages and subtasks, demanding interaction with one or more visualizations. Consequently, the traditional error rates and completion time variables are insufficient for a thorough analysis of viewing behavior.

This thesis targets the upper part of the pipeline from Figure 3.10 for the classical evaluation of visualization techniques. For future evaluation of visual analytics, the more complex distributive cognitive system that includes the user and the machine needs to be assessed as well. To this end, cognitive modeling of the user has to be considered in future work. Cognitive models will have an influence on the task design and the stimuli, which will have to fit the properties of the underlying models. Additionally, the cognitive models will influence the data analysis in both directions, as models can be derived as well as be evaluated with data analysis. With raw and processed gaze data available, the analysis of viewing behavior can be separated into two different approaches: statistical and visual analysis.

Statistical Analysis

An important class of analysis approaches is based on eye-tracking metrics computed from the (pre-processed) gaze data. With AOIs, fixation data can be mapped to the areas and individual statistics can be calculated for each AOI. The common eye tracking metrics can be separated into three categories, according to Poole and Ball [237]:

Fixation-Derived Metrics Fixations with or without AOI information can be processed. A common metric is defined by the number of fixations per AOI, which indicates the relevance of the AOI for the participants. To compare the distribution of attention between AOIs, the sum of fixation durations may be used.

Saccade-Derived Metrics The characteristics of the saccades may indicate the quality of visual cues in the stimulus or the extent of visual searching. For example, large saccade amplitudes can indicate meaningful cues that draw the attention from a distance, or a high frequency of saccades could come from much visual searching. Therefore, saccade-derived metrics can serve to indicate difficulties with the visual encoding.

Scanpath-Derived Metrics The scanpath consists of the full sequence of fixations and saccades. Therefore, scanpath-derived metrics can acquire information about visual reading strategies or pinpoint specific problems with the visualization design during the task. The transition matrix is the common approach to analyzing transition patterns between AOIs, albeit it does not represent the full sequence but only the collection of pairs of fixations from the sequence.

Once values from any of these metrics are available, statistical methods are applied directly, including inferential or descriptive statistics as well as statistical modeling. Therefore, these metrics can serve as a basis for hypothesis testing.

Eye-tracking data contains much more information than represented by the above, aggregated metrics. Statistical analysis can also be applied to data that is closer to the original gaze data. In particular, statistical modeling to predict and classify scanpaths on stimuli provides a promising approach for a more complete analysis of visualization stimuli. Here, one issue is to generate the appropriate model for the scanpath (e.g., define important AOIs) and employ the appropriate statistical methods. In this context, one can use data-mining techniques such as scanpath clustering [124], layered hidden Markov models [90], or measures for the similarity between aggregated scanpaths [129]. Additionally, the evaluation of the participants' experience and gain of insight plays an important role [221]. From this perspective, other quantitative measures derived from eye tracking (e.g., cognitive load [177]) could help quantify complex cognitive aspects [36]. The metrics summarized in this chapter are just the most common that can be found in the evaluation procedures for visualization. For the evaluation of visual analytics, some metrics provide valuable information, such as the distribution of attention between multiple views, but none of them alone captures all the cognitive processes that are involved. Therefore, cognitive models for visual analytics will be required. Which metrics are suitable for the application to visual analytics and the development of new models is still an open research topic.

Eye-tracking metrics have to be interpreted with caution because they can be ambiguous indicators for certain characteristics of cognitive or perceptual processing. In fact, they provide a rather coarse and aggregated perspective on a participant's viewing behavior. Hence, metrics are best accompanied by complementary indicators such as visualization.

Visual Analysis

Visualization complements statistical analysis by providing confirmatory and additional insight into the data by exploratory search, helping with hypothesis building, or presenting analysis results [261]. The same is true in the case of eye-tracking data analysis. In particular, visualization is a good means of examining the spatial, temporal, and spatio-temporal aspects of the data [4].

The most common visualization techniques are heat maps (Figure 3.11a) and gaze plots (Figure 3.11b). Heat maps display the spatial distribution of eye-tracking data on a stimulus. The data can be aggregated over time for one participant or multiple participants. Although heat maps can provide a good overview of important AOIs on a static stimulus, the temporal component of the data is lost. In contrast, gaze plots provide a spatio-temporal perspective on fixation sequences and can be investigated to identify potential reading strategies. With increasing length of the scanpath, or with

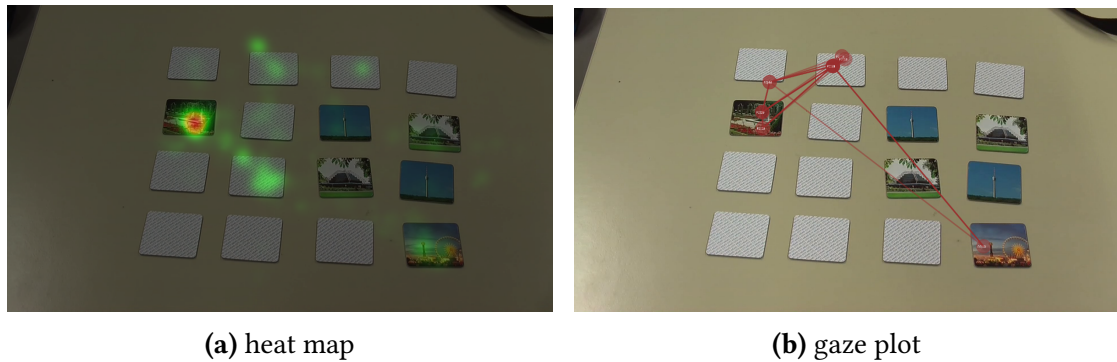


Figure 3.11: Standard methods for visual analysis: (a) The heat map shows the aggregated gaze distribution of multiple participants. (b) Gaze plots depict the scanpaths of individuals.

scanpaths from multiple participants, the visualization becomes cluttered and hard to interpret. Alternatively, transition matrices are applied to analyze gaze patterns but lack the interpretation of longer transition sequences (beyond just pairs of fixations). In summary, the traditional visualization techniques are well prepared to provide a qualitative picture of the gaze distribution aggregated over time (heat maps) or of the short scanpath of a single participant (gaze plot)—both for static stimuli. In these cases, they can also be used for eye-tracking experiments with visualization or visual analytics, in particular, for exploratory data analysis and hypothesis building.

Scanpath Comparison

One important question for eye-tracking analysis considers how similar the viewing behavior between participants is. A typical example is the comparison of experts and novices [164]. For such comparisons, the combination of algorithmic processing and visualization for the interpretation of results is efficient. Established approaches to determine the similarity of two scanpaths can be separated in either trajectory-based or AOI-based methods [38].

Trajectory-Based Comparison

Based on the sampled gaze data, several metrics can be derived. For example, fixation overlap and spatio-temporal correlation of gaze points are automatically computable without AOIs. Furthermore, scanpaths and geo-trajectories have many properties in common. Both consist of a temporal sequence of locations, either on a visual stimulus or in real-world coordinates. Due to this similarity, comparison methods applied to geo-trajectories can also be used for gaze data. Techniques such as *Dynamic Time Warping (DTW)* [47] and the *Fréchet distance* [35] are two popular examples that have been applied for scanpath comparison.

As an additional approach, this thesis compares image-based metrics derived from attended stimulus regions with the established metrics (Chapter 5.3.3), contributing a novel approach for scanpath comparison without annotation [10].

AOI-Based Comparison

Methods based on AOIs all utilize a string representation of scanpaths. String editing methods such as the *Levenshtein distance* [187] and the *Needleman-Wunsch algorithm* [216] are popular approaches to compare scanpaths. Furthermore, methods based on the gaze distribution and on pairwise transitions between AOIs are applicable for comparing viewing behavior [15] (Chapter 5.2.2).

- **Sequential:** Levenshtein's algorithm calculates a distance between two strings by counting edit operations to transform one string into the other. These edit operations are (1) insertion, (2) deletion, and (3) substitution of a character. In the Needleman-Wunsch algorithm, this approach is extended by a weight matrix to penalize edit operations differently. The Levenshtein-based similarity measure focuses on local and temporal coherence of the scanpath strings and penalizes similar object transitions that have a low temporal correlation.
- **Gaze distribution:** The second similarity measure puts no emphasis on temporal coherence and focuses on the gaze distribution of each viewer. This measure aggregates the overall gaze distribution of each participant on the AOIs by counting the number of video frames during which a participant was looking at the respective AOI. It then normalizes this value with the maximum of all AOIs. To quantify the difference between the resulting attention maps the squared difference between each of the components is calculated, which is normalized with the overall number of AOIs and subtracted from 1 to obtain a normalized similarity value. A similar measure for the attention map difference for still images is mentioned by Holmqvist et al. [149].
- **Transitions:** The third similarity measure focuses entirely on pairwise transitions between AOIs. Similar to the gaze distribution method, a transition matrix is calculated for each participant. In addition to the transition between two AOIs, two special states are added: the initial and the final state. This has the effect that the initial and the final AOI of a scanpath are incorporated in the measure. Each of the values of the transition map is normalized by the maximal number of transitions for a pair of AOIs. Again, the similarity is quantified by the sum of squares of pairwise differences in normalized transition frequency, which is normalized with the overall number of pairs of AOIs.

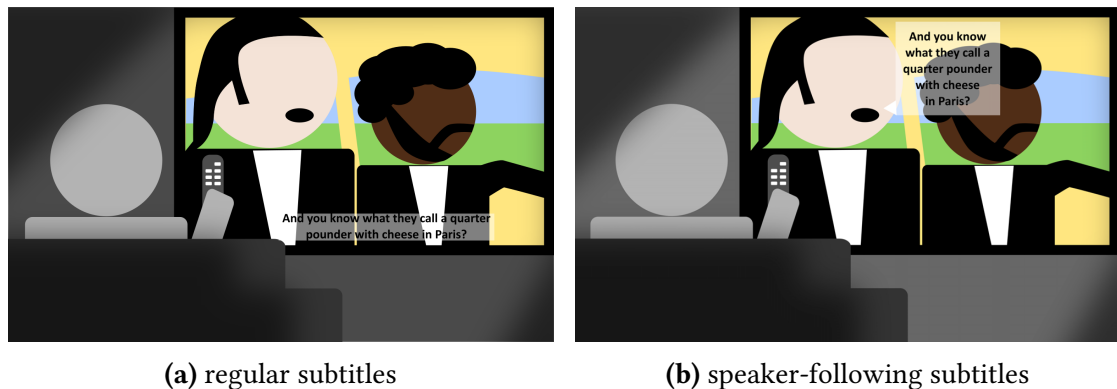


Figure 3.12: Regular subtitles are presented at the center-bottom of the screen. Speaker-following subtitles are displayed with speech bubbles sensitive to the current speaker’s position.

Eye-tracking analysis provides numerous challenging research questions, especially the ones involving video stimuli and many participants. To address these questions, neither statistical nor visual approaches are sufficient on their own. Consequently, this thesis contains improved visual analytics approaches (Chapters 5, 6), combining both aspects to advance the current state of the art in eye-tracking analysis.

To further exemplify how a classical eye-tracking experiment looks like, a user study conducted in the context of this thesis is briefly discussed [27]. It examines the influence of an alternative subtitle layout for videos on the user’s gaze distribution.

3.4.3 Example: Evaluation of Subtitle Layouts

Subtitles in multimedia such as movies and TV shows are important to communicate content for hearing-impaired persons and as an affordable method to translate information into other languages (see also Chapter 2.3). Established approaches present subtitles at the center-bottom of the screen (Figure 3.12a). This position leads to a high visual angle between the subtitle text and the image content, i.e., the current speaker. As a consequence, people watching a video with subtitles constantly have to switch their focus between text and image, leading to increased eye strain and higher chances to miss important content.

This issue is addressed by *speaker-following subtitles* [153], a technique that displays subtitles sensitive to the presented content (Figure 3.12b). Incorporating automatic speaker detection and positioning constraints, it is possible to rearrange subtitle text in speech bubbles close to the speaker, similar to representations in comic books. As a contribution to this thesis, a user study was conducted that compares how viewing behavior changes between regular and speaker-following subtitles [27].

User Study

The user study was conducted, applying eye tracking as an objective measurement of gaze distribution, and a questionnaire to evaluate the subjective impressions of the participants. The study summarizes the results of 40 participants (17 female, 23 male) with an average age of 23 years.

The participant's task was to watch 10 videos (between 1–3 minutes) with alternating layout and summarize the content after each video was over. This task served as motivation to read the subtitles and all participants were able to recapitulate the content in 2–3 sentences. To ensure that all participants would solve the task by reading the subtitles and not by listening, the audio track was removed from the videos.

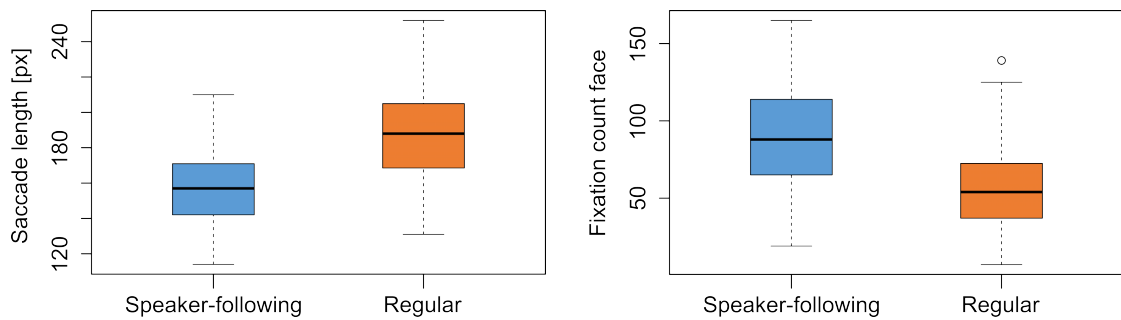
The study was designed to investigate six hypotheses derived from a preliminary pilot study. Within the scope of this example, two of the hypotheses are discussed in detail. Please note that for this example, the numbering of the hypotheses has been changed.

H₁ **The average saccade length for regular subtitles is higher.** This results from the distance between text and image. When participants switch between text and image, they have to overcome longer viewing distances. Longer saccades (increased amplitude) are an important factor in causing fatigue effects.

H₂ **The average fixation count on faces is higher with speaker-following subtitles.** With subtitles being close to the speaker, the participants can better focus on the image content. As a consequence, the fixation count on the speaker should increase.

Results

The results (Figure 3.13) support both hypotheses. With significant differences in saccade length between the layouts, hypothesis **H₁** is supported. The viewing angle between subtitles and important image content is decreased with the speaker-following subtitles. Also, the subjective impression of the participants was that they could investigate the content better with speaker-following subtitles. Considering the fixation counts on AOIs showing subtitles and faces, significant differences could be found that support hypothesis **H₂**. Vice versa, a significant decrease of fixations on subtitles was identified for the alternative layout. The speaker-following subtitles change the gaze distribution in favor of the image content. This means that participants spent more attention on the scene, as they would if they were watching a movie with regular subtitles. Their subjective impressions also reflect that fact. Although less attention was spent on the subtitles, the participants did not have the impression that the readability was impaired in the alternative layout.



(a) Average saccade length (in pixels): speaker-following (median = 157.0, mean = 156.1, sd = 18.5), regular (median = 188.0, mean = 187.4, sd = 24.9). Significant difference according to t-test ($t(398) = -14.3$, $p < 0.01$), H_1 supported.

(b) Average fixation count on faces: speaker-following (median = 88.0, mean = 88.6, sd = 31.9), regular (median = 54.0, mean = 58.0, sd = 26.3). Significant difference according to U-test ($U = 9269$, $N = 200$, $p < 0.01$), H_2 supported.

Figure 3.13: Resulting boxplots of the measures for (a) saccade length and (b) fixation count.

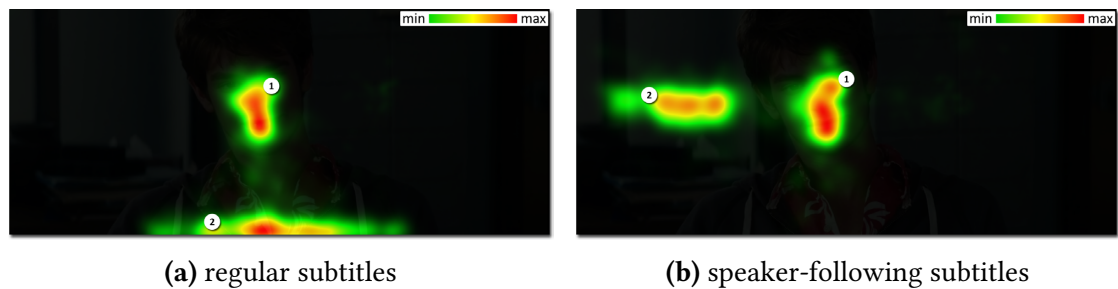


Figure 3.14: Heat maps of approximately 10 seconds of a video. (a) Regular subtitles show many gaze points on a horizontal line at the bottom (②); the face (①) is investigated occasionally. (b) Speaker-following subtitles show fewer gazes on the text and the two hot spots are closer.

To support the statistical results by visualization, Figure 3.14 shows two heat maps that depict how the gaze distribution changes for one shot with respect to the subtitle layout. Regular subtitles show a typical horizontal pattern for text reading at the bottom. For speaker-following subtitles, this horizontal extent is reduced and consists of fewer gaze points. The heat maps summarize only a short time span of the investigated video. To visualize the change of gaze distribution over time, an alternative visualization is necessary. Chapter 5.1.3 presents this data in a space-time cube, showing that the depicted heat map patterns are representative for the general viewing behavior.

This concludes the example of a classical evaluation for eye-tracking data. The investigated subtitle layouts pose a common visualization issue related to label placement. The next section discusses other scenarios in visualization and related research where eye tracking was applied.

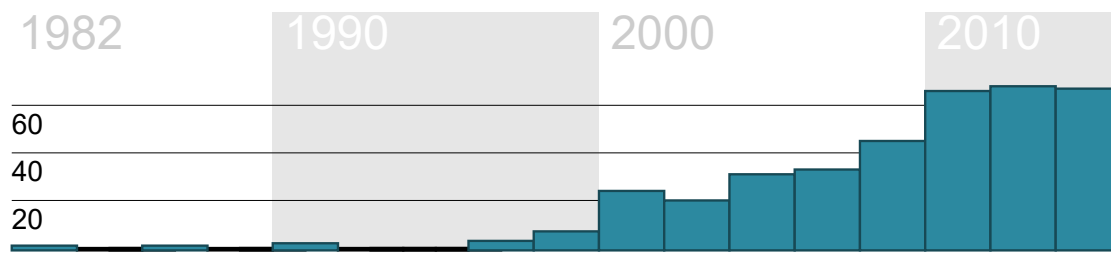


Figure 3.15: Histogram of all 368 publications from visualization and related communities.

3.4.4 Eye-Tracking Evaluation in the Visualization Community

This chapter aims to provide an overview of the current application of eye tracking in visualization research, which is important with respect to the second research question on how to leverage eye tracking to evaluate visualization techniques. Furthermore, future research directions for the evaluation of complex visual analytics frameworks are discussed. Based on a systematic review of publications from the main journals and conferences on visualization [22], as well as related fields from HCI and eye tracking, an increasing number of papers including eye tracking as an evaluation methodology was discovered. Note that also publications in the visualization community exist that apply eye tracking for interactions with applications. These papers investigate how gaze data can be used as an input device, e.g., to replace mouse input. Figure 3.15 shows a histogram of all investigated publications, displaying the increasing importance of this research field. The surveyed publications investigate different visualizations. Based on the established statistical and visual analysis methods for eye tracking data, all publications contain at least one of the aforementioned methods. Hence, the approaches can be summarized, based on the investigated metrics. Three main approaches to evaluate visualizations are identified: evaluating the distribution of visual attention, evaluating sequential characteristics of eye movements, and comparing the viewing behavior of different participant groups.

Distribution of Visual Attention The investigated visualizations are static node-link graphs [156, 217], matrices [174], parallel coordinates [265], 3D meshes with various rendering styles [175], and different user interfaces [277]. For dynamic stimuli, eye tracking is applied to measure the distribution of attention on objects with different video visualizations [12], and to create perceptual motion blur for rendered scenes [274]. The visualization techniques are compared by fixation metrics for the attention on different regions to investigate how the techniques are perceived and to identify possible usability issues. Heat maps are applied to visualize the spatial distribution of attention on the stimuli and support the statistical results. The majority of these publications investigates the spatial distribution of attention directly on the stimulus. If applied, AOIs are defined for rather coarse regions on the screen (i.e., multiple views). For

visualizations that contain small regions of interest (e.g., nodes in a graph), the definition of AOIs can be difficult since the accuracy of current eye tracking hardware might be insufficient to retrieve such small areas. Therefore, the person evaluating a study of a complex visual analytics system has to decide if it is reasonable to investigate small visual components or to consider a coarser scale (e.g., individual views).

Sequential Characteristics of Eye Movements The analyzed stimuli include node-link diagrams [65, 67, 157, 167], linear and radial charts [126], and visualizations with multiple coordinated views [128, 290]. In addition to fixation-related metrics on AOIs, the transition frequencies between AOIs with transition matrices [67], transition graphs [290], and visual scanpath analysis [126] were analyzed to gain insights into how users investigate a visualization (e.g., as an explanation for a decrease in task performance). Also, gaze analysis by visual analytics was applied to identify reading strategies in tree diagrams [65]. As mentioned above, the definition of AOIs in complex visual analytics systems might be problematic but is often necessary to perform most of the analysis related to sequential characteristics. For multiple views, the view itself can be considered an AOI, but also the content of a view could contain multiple AOIs. For such complex structures, the definition of hierarchical AOIs [2] could be considered to investigate the behavior between and within different views. Since many visualizations consist of rendered content with known geometry, the definition of potential AOIs based on this content can be considered.

Comparison Between User Groups The visual stimuli in this category are virtual character models [46] and cross-sectional medical images [270]. Complementary to the previous two points, the distribution of attention between different groups is investigated. Group comparisons are performed between healthy and mentally disordered persons, or between novice and expert groups. Comparisons are based on a statistical analysis of AOI fixation metrics [46] or visual comparison of gaze point distributions [270]. For visualization analysis tasks, the expertise of a participant also plays an important role. For the application to visual analytics, one point that should be considered more in the future is the influence of the visual span of participants. For example, Reingold et al. [240] investigate the viewing behavior of chess players with different levels of expertise. As a result, they observe that novice players fixated more on individual pieces, whereas expert players have a greater proportion of fixations between chess pieces, indicating a larger visual span to investigate more pieces at once. For visualizations, this behavior needs to be investigated in more detail. As a consequence of an increased visual span, the accuracy of the eye-tracking device is less problematic, because a much larger area on the screen with potential AOIs has to be considered. Approaches that count fixation hits on AOIs might not be sufficient for evaluations with expert participants. Therefore, an uncertainty factor could be applied to distribute the visual attention between potential AOIs.

These eye-tracking studies mainly rely on the statistical analysis of AOI-based fixation metrics. The main focus of these studies is on static visualizations where the definition of AOIs is less complicated than with dynamic content. As discussed, the proper definition of AOIs and the influence of the visual span are two important points that have to be considered during the design process of a study. If performed, visual data analysis is often limited to the investigation of heat maps and gaze plots. For the identification of visual reading strategies, more advanced visual analytics techniques are applied. However, none of the above studies investigate the full sequence length of scanpaths or any complex spatio-temporal characteristics of eye tracking for dynamic stimuli, let alone any cognitive aspects related to the mixed-initiative distribution of cognition in visual analytics. Considering the applied hardware, the main part of the studies is conducted with a remote eye-tracking system, which should be sufficient for studies with one participant. In collaborative scenarios, for example, a visualization expert working with a domain expert, the application of wearable eye-tracking glasses for each expert are required to capture eye movements from both participants. In addition to these important points, related communities are inspected to obtain further inspiration of how eye-tracking evaluation of visual analytics might be performed in the future.

Related Communities

Publications of the *Conference on Human Factors in Computing Systems (CHI)* and the *Symposium on Eye Tracking Research and Applications (ETRA)* provide extended evaluation methods using eye tracking. Technically, not only video-based eye tracking, but also electrooculography and head tracking are applied to estimate a participant's point of regard. Because these approaches are of limited suitability for an application to visual analytics scenarios, this survey focuses on the video-based systems.

In the CHI literature, eye tracking is applied for two main reasons: hands-free interactions with computers and for usability testing. Evaluations mainly focus on websites, text, and graphical user interfaces. Also, gaze behavior during driving simulations, on mobile devices, and in code programming is investigated several times.

The ETRA literature contains publications with similar research because various authors publish work at both conferences. Investigated stimuli comprise those from CHI with more work investigating videos and photographs, as well as artificial stimuli from psychological research. The evaluation of standard metrics can be found in most of these publications. In addition to these metrics, further purposes and approaches were presented to analyze eye-tracking data. Those could also be beneficial for the evaluation of visualizations.

Cognitive Modeling and Machine Learning Eye-tracking data is analyzed to infer statistical models to predict and classify human behavior. Examples comprise research to identify time spans of visual search and reading behavior, as well as visual saliency models that predict regions of interest in interactive environments [68, 102, 151, 231, 252]. Similar approaches can also be found in smaller numbers in the visualization community [142]. Such models could also be applied to visual analytics. A predictive model could influence the design process of a system, telling the developer how the layout of visual components could be optimized. Additionally, it could be used during the analysis to guide attention to relevant parts of a visualization.

Correlation of Gaze and Mouse Data Another important aspect of usability evaluation with eye tracking is to find out how mouse input and visual attention work together in different scenarios and tasks. The main focus of these publications is on the interaction behavior with websites [130, 155]. The application to complex graphical user interfaces such as visual analytics systems is limited to a single publication [54] and will provide a challenge for future research.

Pupil Dilation Measurements Physiological data from pupil dilation is often available from the recorded eye-tracking data. The identified work on this topic considers the data as an indicator for cognitive load, arousal, and vigilance [159, 227, 229]. A direct application of these measurements to other stimuli seems reasonable, but to this point, such evaluation procedures are seldom in the visualization community [112].

Retrospective Think Aloud As a variant of think aloud, eye-tracking data is included in a *retrospective analysis (RTA)* [103, 131]. Gaze data is either displayed to the participants as visual cue during the replay of their task performance or applied to check the validity of protocols. Although this combination still requires further investigation, a general application of the RTA method to a visual analytics context might be a good approach to produce reliable results, since eye-tracking data and task performance are influenced by the think-aloud method if performed during the task.

CHI and ETRA publications contain much more work on the sequential analysis of scanpaths. Here, the quantitative analysis of common transition sequences between AOIs and similar scanpath patterns is also applied for the analysis of viewing strategies.

Future Directions

With the availability of cheap eye-tracking hardware and its ease of use, there are no longer any technological obstacles for using eye tracking in user-based evaluation; in particular, in controlled laboratory studies, gaze data can essentially be recorded for

free, along with any traditional study procedure that aims to test task performance. Therefore, the big overall challenge is to make sense out of the eye-tracking data and relate this data to something we want to learn about the visualization tested and the cognitive processes involved. As discussed before, there are already several examples of eye-tracking studies in visualization: they mostly work with statistical analysis of aggregated data, for well-defined hypotheses, and with traditional visual analysis by heat maps and gaze plots. In fact, many other laboratory studies could adopt these approaches to testing and data evaluation, adding a better understanding of reasons for task performance. Therefore, the general recommendation is that eye tracking should be considered as a testing method whenever a laboratory study is planned and designed.

However, the real value of eye tracking goes beyond what is possible now. Based on the reflections on the state of the art, relevant directions for future research on evaluation methodology are discussed, beginning with more technologically oriented research questions asking for short term action, and ending with long term grand challenges.

Study Design

The study design for future evaluation procedures in visual analytics will have to consider some changes for the applied stimuli and tasks. The visual stimuli (i.e., interactive visual analytics systems) should include the possibility to produce data to identify AOIs on the screen. Given that the rendered content is known, dynamic changes of position and size of a visual component can be tracked and logged. This preparation step will help increase the efficiency of the evaluation. The study design should already consider the granularity and type of potential AOIs. For future research, the classical task performance analysis will not be sufficient to evaluate the insight gain of a participant using a visualization or visual analytics system [221]. Referring to the evaluation pipeline (Figure 3.10), this means that the task section will significantly differ from classical performance analysis. New classes of tasks will be required that are less restrictive than classical *search & report tasks*. Approaches that leave more freedom to the participant to explore a dataset increase the difficulty for the evaluation later on. In addition to the qualitative, open-ended protocol approach suggested by North et al. [221], the analysis of eye-tracking data, for example, the identification of reading strategies, could provide a quantitative component on the way to measure insight.

Exploratory Data Analysis and Hypothesis Building

Statistical methods can be applied once clearly defined hypotheses exist and an eye-tracking experiment was set up accordingly. The interesting question is how such an eye-tracking experiment can be designed, in particular, for the complex visual representations and tasks in applications of visualization and visual analytics. Here, great potential lies in improved data analysis methods that could work on eye-tracking

data acquired in less constrained preliminary studies. Visual analytics will certainly play a major role here [40], in particular, for the complex spatio-temporal nature of the eye-tracking data and the (dynamic) stimulus data, and by combining data mining, statistical, and interactive visualization methods.

Scanpath Comparison One analysis aspect is most relevant, albeit difficult: improved *scanpath analysis*. So far, the studies in the visualization community focused mainly on the spatial aspect of the recorded gaze data. Temporal aspects of the data, such as AOI sequences, provide important information about reading strategies but were often neglected entirely or only partially covered through transition matrices. More work in this field was performed in related communities, often applying algorithms for statistical analysis. For a full understanding of common scanpath patterns, a combination of automatic algorithms for processing these patterns and visualizations for interpreting the patterns could be the best solution. Therefore, better visual analysis techniques for long sequence information are required. Because gaze plots tend to cause visual clutter with an increasing number of participants and scanpath length, a visual comparison becomes problematic with standard approaches. Hence, a visual analytics approach seems to fit best for analyzing eye-tracking data recorded from using visual analytics systems.

Data Fusion A third aspect is the *combination of eye-tracking data with additional time-oriented data*. For example, the temporal evolution of the dynamic stimuli needs to be understood to build the context for the gaze data. Or, the eye-tracking data can be combined with information about logged interactions such as mouse or key-stroke data, to obtain deeper insights into the usability of interactive visualization and visual analytics systems. The evaluation of interactive systems solely based on gaze data and performance measurements might lack details for a full interpretation of the participants' cognitive processes. Continuing the preliminary work from other communities, the fusion of multiple data sources (e.g., eye tracking with *interaction logs* [3, 52]) could provide this missing data for the interpretation. In the field of visual analytics, in which evaluated systems are often far more complex than simple menus and websites, this approach opens a new research field where only a few works exist to this end. Another trend is to include other physiological measures into an eye-tracking experiment. For example, electroencephalography (EEG) measures can already be included in the software suites of known eye-tracking vendors. Including such measures could help to understand the interrelation between these components. Because *pupil dilation* is already recorded by many eye-tracking devices as an additional measurement, current research from other communities could also be applied for the evaluation of visual analytics and visualization techniques. For example, in long testing sessions when using a complex analysis tool, participants could get tired and time spans

when they just stare at the screen might occur. During these time spans, long fixations would be identified without any cognitive processing of the participants. Hence, a temporal measure for vigilance and cognitive load would increase the reliability of the gaze analysis afterward.

Evaluation Tools A practical aspect is concerned with making the newly developed analysis methods available to other researchers. Reflecting a general discussion in the visualization community, *disseminating codes, tools, and systems* is necessary so that improved analysis can be adopted quickly. One way is to have advanced analysis methods included in professional software by the vendors of eye-tracking hardware; however, this approach might not always work due to the latency in the software development process and because not all visualization-related analysis problems will be sufficiently relevant for the broader eye-tracking audience. Therefore, there should also be dissemination of software (prototypes) developed, including complete analysis systems but also partial codes.

In conclusion, eye tracking becomes increasingly important as a means for evaluation in many areas, including visualization research. The high complexity of gaze data recorded from dynamic stimuli such as videos requires new approaches for an effective analysis of data from multiple participants. Hence, the following chapters will discuss how visualization and visual analytics can provide such approaches and present the technical contributions of this thesis for the analysis of eye-tracking data.

Visualization of Eye-Tracking Data

The previous chapter mentioned eye tracking as a means for the evaluation of visualization and visual analytics. Vice versa, visualization and visual analytics help interpret gaze data from user-based evaluation. The main contributions in this thesis focus on the development of new techniques to support a better understanding of eye-tracking data, especially in the context of dynamic stimuli. Based on extensive literature review, the state of the art for the visualization of gaze data is surveyed and a taxonomy is derived under the aspects of common analysis tasks [26] and technical aspects of visualization approaches [1, 4].

This chapter discusses eye tracking in the context of the visualization pipeline (Chapter 4.1). It further presents a taxonomy for eye-tracking visualizations (Chapter 4.2), derived from the current state of the art. The techniques developed in this thesis are categorized with respect to the taxonomy (Chapter 4.3). Furthermore, the technical contributions in this thesis are mainly presented with a benchmark dataset (Chapter 4.4) for eye tracking and visualization.

This chapter is partly based on the following publications:

- T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl. “State-of-the-Art of Visualization for Eye Tracking Data”. In: *Proceedings of EuroVis State of the Art Reports*. 2014, pp. 63–82 [1]
- T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl. “Visualization of Eye Tracking Data: A Taxonomy and Survey”. In: *Computer Graphics Forum* 36.8 (2017), pp. 260–284 [4]
- K. Kurzhals, C. F. Bopp, J. Bässler, F. Ebinger, and D. Weiskopf. “Benchmark Data for Evaluating Visualization and Analysis Techniques for Eye Tracking for Video Stimuli”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2014, pp. 54–60 [20]
- K. Kurzhals, M. Burch, T. Blascheck, G. Andrienko, N. Andrienko, and D. Weiskopf. “A Task-Based View on the Visual Analysis of Eye-Tracking Data”. In: *Eye Tracking and Visualization – Foundations, Techniques, and Applications (ETVIS 2015)*. Ed. by M. Burch, L. Chuang, B. Fisher, A. Schmidt, and D. Weiskopf. Springer, Cham Switzerland, 2017, pp. 3–22 [26]

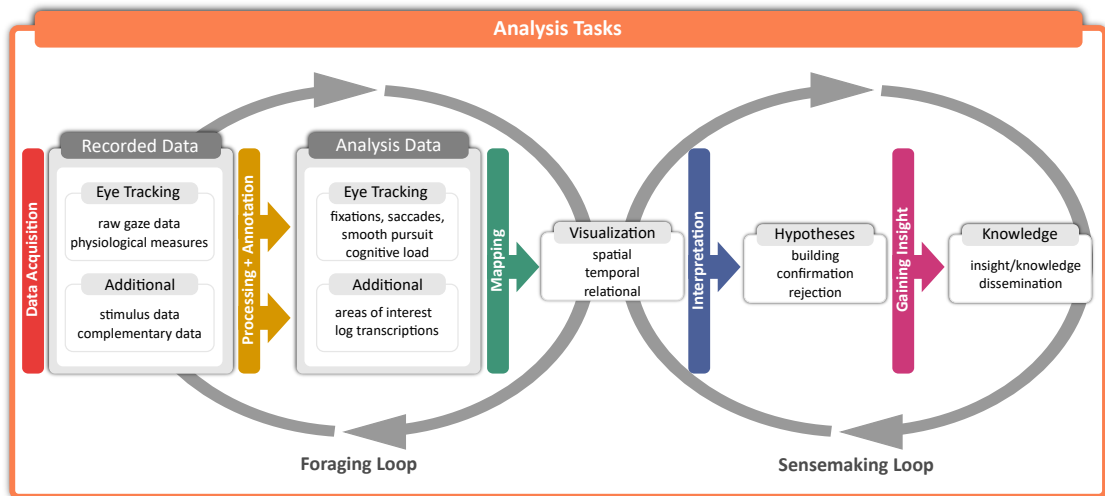


Figure 4.1: Extended visualization pipeline for eye-tracking data: The recorded data passes multiple transformation steps before knowledge is extracted. Each step from data acquisition, processing, mapping, interpretation, to gaining insight is influenced by the analysis task.

4.1 Eye-Tracking Visualization Pipeline

From a data perspective, eye-tracking recordings primarily consist of spatio-temporal information, optionally enhanced by the semantics of the stimulus and complementary data sources. The visualization reference model can be interpreted in the context of gaze data. From the data acquisition to the extracted knowledge from the data, the analysis task influences the choice of appropriate visualization techniques.

The procedure from conducting an eye-tracking experiment to gaining insight can be generalized in the form of a pipeline (Figure 4.1) that is an extended version of the generic visualization reference model (Figure 2.7). The acquired data consists of eye movements and complementary data. It is processed and optionally annotated before the visual mapping step. As discussed in Chapter 2.2, by interacting with the data and the visualization, two loop processes are started: a foraging loop to explore the data and a sensemaking loop to interpret it and to confirm, reject, or build new hypotheses from where knowledge can be derived [234]. For all steps, the analysis task plays an important role, determining which actions to take and which visualization fits best.

Data Acquisition

Eye tracking combines several data dimensions. It comprises dimensions directly stemming from the recorded eye movements (raw gaze, physiological measures) and additional data sources serving as complementary data that can help achieve more reliable analysis results when combined with gaze data. The displayed stimuli are an

additional data source that is often included in the analysis. Other data sources provide complementary data such as verbal feedback, EEG data, and keypress protocols. The analysis task defines how the experiment is designed and which data will be recorded. Most scenarios predefine also the visual stimulus. Exceptions are, for example, *in-the-wild* experiments with mobile eye tracking where it becomes more difficult to control the experiment parameters.

Processing and Annotation

From the time-varying sequence of raw gaze points, more data constructs can be derived in a processing step. Automatic data-mining algorithms are applied to filter and aggregate the data. Clustering and classification are prominent processing steps. For example, raw gaze points are clustered into fixations and labeled. As another example, the convex hull of a subset of gaze points can be extracted to identify AOIs automatically. In general, the annotation of AOIs plays an important role in this step. Especially for video sequences, this annotation is a time-consuming step that often takes more effort than the rest of the whole analysis process. Recorded protocols and log files are derived from the additional data sources. It should be noted that each additional data source requires synchronization with the recorded gaze data, which can be difficult considering different sampling rates and not regularly sampled data (e.g., think aloud) [3]. The processed data is finally mapped to a visual representation.

The analysis task influences what filters are applied to the data and what AOIs are annotated. For explorative scenarios in the context of visual analytics, the visualization and the processing are tightly coupled in a foraging loop, where the analyst can identify relevant data artifacts through interaction with the visualization.

Mapping

The mapping step projects the analysis data to a visual representation. According to the introduced taxonomy (Chapter 4.2), the main categories of state of the art visualization techniques for eye tracking are spatial, temporal, and relational data representations. Therefore, this task categorization follows a similar scheme and appropriate visualizations are selected according to the main data dimension that is required to perform the corresponding task. It may be noted that only a few visualization techniques for gaze data also take into account the additional data sources for an enhanced visual design in order to explore the data. Those data sources may build meaningful input for sophisticated data analyses if they are combined with gaze data.

The analysis task plays the most important role in choosing the appropriate visualization technique. In the foraging, as well as the sensemaking loop, the visualization has to convey the relevant information and should provide enough interaction supported by automatic processing to adjust the visualization to the specific needs of a task.

Interpretation

Two strategies can be distinguished for the interpretation of the visualization: (1) Applying visualization to support statistical measures and (2) performing an explorative search. In the first case, hypotheses are typically defined before the data is even recorded. Therefore, inferential statistics are calculated on appropriate eye-tracking metrics, providing p -values to either support or reject hypotheses. Here, visualization has the purpose to support these calculations additionally. In the second case, the explorative search, hypotheses might be built during the exploration process. Filtering and re-clustering data, adjusting the visual mapping and reinterpreting the visualization can lead to new insights that were not considered during the data acquisition. This explorative approach in the context of eye-tracking studies is particularly useful to analyze data from pilot studies. Building new hypotheses, the experiment design can be adjusted and appropriate metrics can be determined for hypothesis testing in the final experiment. The interpretation of the data strongly depends on the visualization.

With a single visualization, only a subset of possible analysis tasks can be covered. For an explorative search where many possible data dimensions might be interesting, a visual analytics system providing multiple different views on the data can be beneficial.

Gaining Insight

As a result of the analysis process, knowledge depending on the analysis task is extracted from the data. As discussed before, this knowledge could be insights that allow the researchers to refine a study design or conduct an entirely new experiment. In the cases where visualization has the main purpose to support statistical analysis, it often serves as dissemination of the findings in papers or presentations. In many eye-tracking studies, this is typically the case when inferential statistics are performed on metrics and heat maps are displayed to help the reader better understand the statistical results.

4.2 Taxonomy

With respect to the presented visualization pipeline, a taxonomy is derived to classify existing techniques. Accordingly, the two main categories separate task-related and technical aspects of a visualization (Figure 4.2). Task-related aspects consider the possible research question a visualization tries to answer. The technical category comprises aspects of the gaze data, the visualization, and the stimulus. Both categories complement each other and are essential for the choice of an appropriate technique to address a research question. However, a single technique often provides answers to multiple questions and is therefore difficult to categorize from a task-related perspective. Hence, the task-related categories are discussed first, including examples of appropriate

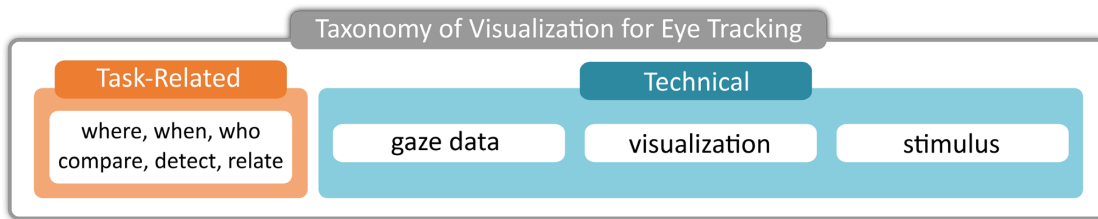


Figure 4.2: Main categories of the taxonomy of visualization for eye tracking, consisting of task-related (orange) and technical categories (blue).

visualization techniques. Then, the technical categories are discussed. These categories were applied to survey the current state of the art.

4.2.1 Task-Related Categories

The visualization pipeline for eye-tracking data (Figure 4.1) shows the steps in which analysis tasks play an important role. For the experienced eye-tracking researcher, the first two steps—*data acquisition* and *processing*—are usually routine in the evaluation procedure. In the context of this chapter, *mapping* is the most important step in which the analysis task has to be considered. When the analysis task is clear, the chosen visualization has to show the relevant information. Hence, a categorization of analysis tasks is necessary to help with the choice of an appropriate visualization. The main properties of the involved data constructs are discussed as well as typical measures for these questions. To provide a systematic overview of typical analysis tasks, the three independent data dimensions (following the questions discussed in Chapter 2.3) in eye-tracking data are:

- **Where?** For these tasks, space is the most relevant data dimension. Typical questions in eye-tracking experiments consider where a participant looked at.
- **When?** Tasks in which time plays the most important role. A typical question for this dimension is: when was something investigated the first time?
- **Who?** Questions that investigate participants. Typical eye-tracking experiments involve multiple participants and it is important to know who shows a certain viewing behavior.

With these three independent dimensions, visualizations is applied to display dependent data constructs (e.g., fixation durations). Since many visualization techniques are not restricted to just one of these dimensions but facilitate different combinations between them, categories are discussed with the techniques where the name-giving dimension can be considered as the main dimension for the visualization.

Additionally, the data is related to general analytical operations that can be found in other taxonomies (e.g., the KDD process [111]):

- ▶ **Compare:** Questions considering comparisons within one data dimension.
- ▶ **Relate:** Questions considering relations between data dimensions and constructs.
- ▶ **Detect:** Questions about summarizations and deviations in the data.

This categorization is based on the surveys written in the context of this thesis [4, 22] and the work of Andrienko et al. [40]. An overview of current state-of-the-art visualization and visual analytics approaches for the analysis of eye-tracking data is presented after the task categorization.

Where? – Space-Based Tasks

Typical questions considering the spatial component of the data are often concerned with the distribution of attention and saccade properties. Statistical measures such as standard deviations, nearest neighbor index, or the Kullback-Leibler divergence provide an aggregated value about the spatial dispersion of gaze or fixation points. If a saccade is interpreted as a vector from one fixation to another, typical *where* questions can also be formulated for saccade directions. With AOIs, measures such as the average dwell time on each AOI can be calculated and represented by numbers or in a histogram.

Space-based tasks for dynamic stimuli, such as videos and interactive user interfaces require a visualization that takes the temporal dimension into account, also considering the changes of the stimulus over time. With AOIs, questions about *when* and *where* are tightly coupled. An example of a visualizations with focus on spatio-temporal analysis, i.e., a space-time cube [16] is presented in Chapter 5.1.2.

When? – Time-Based Tasks

Gaze data has a spatio-temporal nature often demanding for a detailed analysis of changes in variables over time. Questions in this category typically focus on a certain event in the data (e.g., smooth pursuits) and aim at answering when this event happened. Considering the detection of specific events over time, many automatic algorithms can be applied to identify these events. Automatic fixation filtering [253], for example, calculates when a fixation started and ended. For semantic interpretations, AOIs are included to answer questions *when* was *what* investigated.

Without AOI information, the visual analysis of the temporal dimension is rather limited. Statistical plots over variables such as the x- and y-component [125], or acceleration of the eye can provide useful information about the physiological eye-movement process.

However, combined with the semantic information from AOIs, visualizations help to better understand when attention changes appear over time. Timeline visualizations are a good choice to answer questions related to this category. Chapter 5.2.2 discusses an approach where multiple timelines for different AOIs are stacked on top of each other [15]. Colored bars on the timelines indicate when an AOI was visible. Alternatively, this binary decision could also be applied to depict whether a participant looked at the AOI [273, 303]. In general, timeline representations depict an additional data dimension, allowing one to combine relevant data its temporal progress.

Who? – Participant-Based Tasks

Typical questions raised when looking at recorded participants' data can be categorized into those concerning only a single individual or a larger group of people. Inspecting the viewing behavior of participants provides insights into the visual task solution strategies applied by them [65]. Generally, most visualization techniques for multiple participants work fine also for an individual participant. Comparisons are facilitated by similarity metrics. To interpret the results, a visual scanpath representation that supports the similarity measure is helpful. For visualization, timelines for individual participants with color-coded time spans can be created, commonly known as scarf plots [15, 242] (Chapter 5.2.2).

Compare

Comparison, in general, can be seen as one of the elementary analysis operations performed during the evaluation of eye-tracking experiments. In fact, statistical inference is calculated by comparing distributions of a dependent variable. However, inferential statistics can only provide the information that a difference exists. To identify what the difference between the conditions is, a visual comparison is usually a good supplement to the statistical calculations.

Comparison tasks are typically supported by small multiples visualizations. An example of such visual comparisons can be found in a seminal eye-tracking experiment conducted by Yarbus [312], in which participants investigated the painting *The unexpected visitor*. To compare the different viewing behavior during alternating tasks, the resulting gaze patterns were depicted by rudimentary gaze plots, allowing an easy interpretation of how the task influenced the eye movements. A more direct and supportive way to perform comparison tasks is by the principle of agglomeration. In this concept, two or more data instances are first algorithmically compared and the result is encoded in a suitable visual metaphor, for example with a dendrogram. This approach was applied in multiple techniques presented in this thesis (Chapters 5.2.2, 5.3.1).

Relate

In most analysis scenarios, not only a single dimension is in the research focus. Correlations between data dimensions in eye-tracking research are often analyzed statistically, while the interpretation of the data can be achieved visually. Typical examples are scatter plots or parallel coordinate plots.

Investigating relations between AOIs is another important analysis task. Relations between AOIs are often examined by counting transitions between them. Transition matrices or Markov models provide valuable insights into the search behavior of a participant [149]. Alternative techniques for showing relations between elements are graphs and trees. A transition graph depicts AOIs or meta information about AOIs as nodes and transitions as links [53]. Trees are typically used to depict the sequence of transitions [2]. These trees can also be used to visually compare the sequences of different participants and depict common strategies in a visual form [13, 294, 307]. A tree-based approach to analyze sequential visits of AOIs is discussed in Chapter 5.2.3.

Detect

Detecting patterns of common viewing behavior is often achieved by summarization of the data. Calculating the average fixation duration, the variance of saccade amplitudes or the mean scanpath length are some examples. Box plots are typically used to represent these values and depict outliers as a simple-to-understand graph. Summaries can be created for the raw data points, for aggregated data using AOIs, or for the participants. Some visualizations are specially designed, or suitable, for detecting outliers and deviations in the data. Here, timeline visualizations [129, 24] showing one data dimension over time can be applied. As an alternative, an image-based technique for this task is presented in Chapter 5.3.1.

AOIs may also be used to find deviations in the data. For example, an AOI may not have been looked at during the complete experiment by one or multiple participants. This may be an indicator that the AOI was not needed to perform the experiment task or participants missed important information. AOI timelines can help answer this question. Presenting AOIs next to each other [238, 174] allows a direct comparison to inspect which AOIs have been looked at or not. Furthermore, individual participants may show different strategies, which can be found with scanpath comparison.

4.2.2 Technical Categories

With the increasing number of eye-tracking studies conducted, the need for new analysis techniques emerged. Visualization supports the aforementioned analysis tasks to help communicate results, extract insights efficiently, or provide answers to a research question in the first place. Even in scenarios where the stimulus does not change,

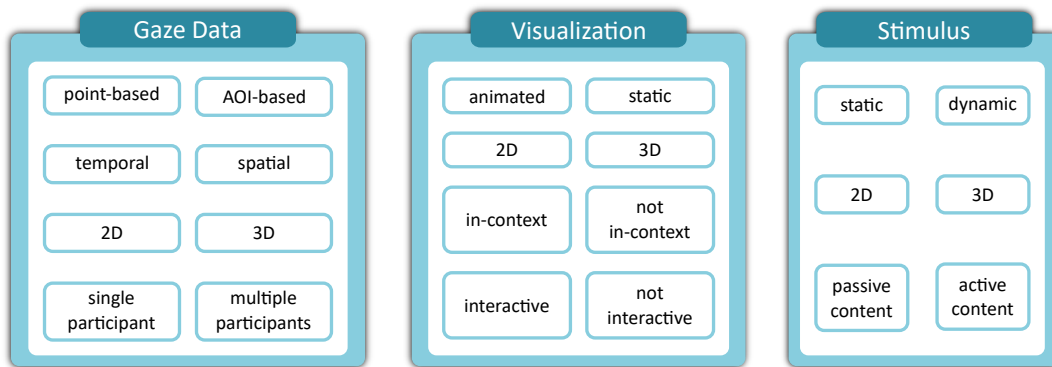


Figure 4.3: Visualizations for eye-tracking data can be classified by three main categories: aspects of *gaze data*, the *visualization* itself, and the *stimulus*.

complex questions (e.g., regarding relations between AOIs and participants) are hard to solve with basic techniques. Changes on the stimulus, as occurring in videos or interactive applications require one to watch a stimulus replay with the basic visualizations superimposed. One can imagine, if *The unexpected visitor* [312] were a video and not a painting, the comparison of different tasks would have been far more difficult. Due to changing positions of persons or motion that attracts attention, a representation by gaze plots would be less expressive than for the static picture. Consequently, the importance of visualization in eye-tracking research increased, yielding new data representations and modifying existing ones. Based on an extensive literature review [1, 4], visual representations for gaze data are investigated and categorized. The resulting taxonomy differentiates between three categories, i.e., aspects of the *gaze data*, the *visualization*, and the *stimulus* (Figure 4.3).

Gaze-Related Categories

Techniques are based on the type of investigated data, which is either point-based, AOI-based, or a combination of both. The multiple dimensions of gaze data concern temporal and spatial data, 2D/3D coordinates, and single or multiple participants.

Point-based/AOI-based Point-based data concerns raw data points and aggregated data (e.g., fixations), often mapped to the coordinate system of the stimulus. The spatial context of the stimulus plays an important role in such visualizations, as it is necessary for an interpretation of the results. Annotated data provides semantic information that can be utilized to create visualizations that abstract from the spatial context.

Temporal/spatial/spatio-temporal Analysis solely based on the spatial dimension, namely the x-, y-, and z-dimension, mainly considers the distribution of gaze points. The temporal dimension of gaze data allows the inspection of changes. Hence, a combined analysis of spatial changes over time is often preferable in visualization techniques.

2D/3D The spatial dimensionality of gaze data may vary between 2D and 3D. In most settings with a regular monitor, a 2D mapping on a plane is sufficient. Coupled with the dimensionality of the stimulus, an experiment might also require a calculation of 3D gaze positions as it is necessary in virtual and mixed reality [232, 233].

Single/multiple participants Depending on the task, the analysis of a single participant is often not sufficient. For comparisons and summarizations of viewing behavior, data from multiple participants is required. One issue with many visualizations depicting multiple participants is the increase of visual clutter [248].

Visualization-Related Categories

Taxonomies for visualization either consider data dimension or type [81, 288], interaction techniques [264, 313], task [60], or visualization types [69, 184]. However, for the specific case of eye-tracking visualization, these taxonomies are too general or restricted. Hence, the considered aspects for the visualization are:

Animated/static Static visualizations handle data based on a time-to-space mapping. This is often worthwhile to provide an overview without interaction necessary. Without AOIs, static visualizations with semantic context are hard to achieve. Animations use a time-to-time mapping, representing the data sequentially. If animated visualizations are designed as an overlay, the data and visualization are kept in the same domain.

2D/3D visualization The combination of visualized data dimensions in 2D provides multiple variations. The x- and y-coordinates of the data can be represented directly, as displayed in heat maps. Another possibility is the representation of time on one axis, and showing one of the remaining data dimensions on the other axis. Similarly, the extension to 3D does not necessarily require a representation of all spatial dimensions of the gaze data. For example, a space-time cube visualization [16] represents the x- and y-axis of the stimulus and adds time as the third dimension.

In-context/not in-context Visualizations that show the context of the stimulus are often designed as a visual overlay that requires animation for inspection. AOI-based representations usually abstract this context and do not depict it directly. Such abstractions might not show important details of an AOI.

Interactive/not interactive An interactive visualization enables the analyst to adjust parameters and views. With basic interactions such as zooming and filtering, data exploration becomes possible. For dissemination, a fixed parameter set can be used to extract static images for a protocol or publication. In this context, the term *interactive* mainly concerns techniques that actively support data exploration.

Stimulus-Related Categories

In addition to recorded gaze data, the stimulus provides important information. Tightly coupled with the other categories, the following aspects are differentiated:

Static/dynamic Static images played an important role for many years in eye tracking research. Dynamic content from watching videos or real-world scenarios is far more complex and poses new challenges for the visualization. Generally, a visualization developed for dynamic stimuli can also be applied to static content.

2D/3D stimulus As discussed for the gaze data, if a 2D coordinate system is sufficient for analysis purposes, the gaze data is typically mapped into the coordinate system of the stimulus. In scenarios where depth is important (e.g., stereoscopic displays, mixed reality), the third spatial dimension has to be considered.

Passive/active content This aspect considers the participant's mode of interaction. Participants can watch stimuli passively, for example, pictures or videos. With active content, each participant influences the stimulus individually. Examples are recordings of mobile eye tracking or interactions with a desktop application. Due to individual differences between recordings, a comparison is more difficult than with passive content.

4.3 Categorization of Visualization Techniques

With the presented technical categorization (Section 4.2.2), existing techniques can be classified. In the following, a summarization of existing techniques is discussed and how the techniques developed in this thesis fit in the classification scheme.

4.3.1 State of the Art

The first separation is made between point-based and AOI-based approaches. Figure 4.4 presents a summarization of all publications investigated in the survey [4] excluding the publications from this thesis.

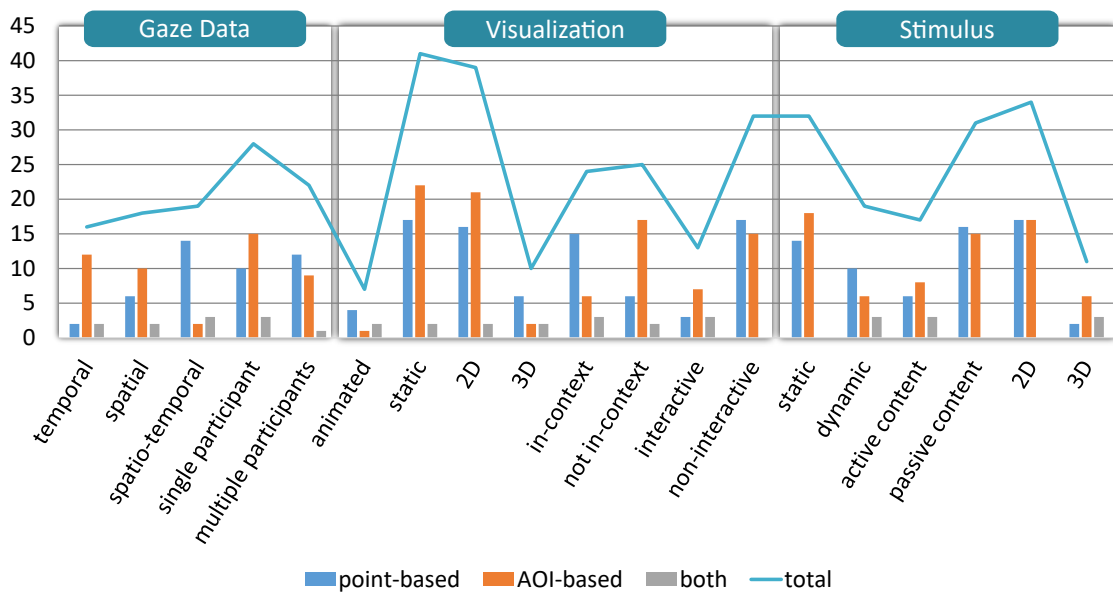


Figure 4.4: Summarization of categorized publications presented in Blascheck et al. [4] for techniques that are point-based, AOI-based, or both.

Gaze Data The categories related to gaze data show in total a similar number of techniques representing *temporal* and *spatial* aspects with slightly more techniques for *single participants*. In detail, the point-based approaches often consider *spatio-temporal* aspects and AOI-based techniques focus on either the *spatial* or the *temporal* dimension.

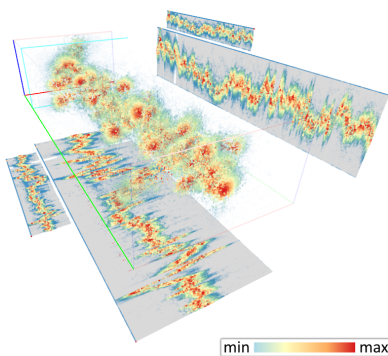
Visualization The visualization categories show a strong preference of *static* over *animated* techniques. This seems reasonable since animation for eye-tracking data is important to see details, but not for an overview of the data. Similarly, *2D* visualizations were preferred over *3D* techniques, mainly because the spatial domain of the stimulus was also often investigated in *2D*. The total sum of techniques shown *in-context* of the stimulus and *not in-context* is equal, but with a clear preference of *in-context* for point-based techniques and *not in-context* for AOI-based techniques. The majority of investigated techniques were classified as *non-interactive*, meaning that interaction was often reduced to parameter adjustment while *interactive* approaches provided support for data exploration.

Stimulus Regarding the stimulus, more techniques focused on *static* stimuli, regardless if they were point-based or AOI-based. *Passive* content was primarily investigated, containing *static* and *dynamic* stimuli. In most techniques, a *2D* stimulus was used. In cases where *3D* stimuli were analyzed, AOIs also played an important role.

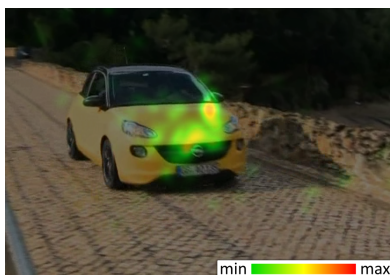
Some aspects that were less prominent in the classification have been neglected for good reasons. For example, animated visualization is often sufficient with a video replay of heat maps or gaze plots and 3D visualizations are typically not necessary for a 2D spatial domain. In contrast, the lack of interactive techniques with support for dynamic, active, and 3D stimuli can be interpreted as possible white spots in research. Furthermore, the combination of individual aspects plays also an important role, because depending on the supported aspects, different analysis tasks can be solved.

4.3.2 Contributed Techniques

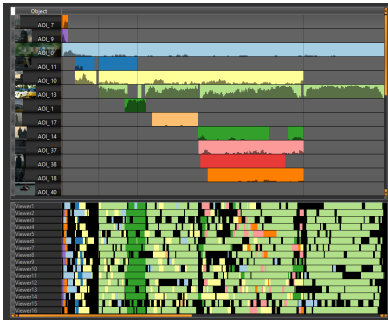
The following techniques were developed during this thesis and will be further discussed in the Chapters 5 and 6. This chapter aims to provide an overview of developed techniques and how they fit into the presented taxonomy. The techniques cover different aspects of this taxonomy, mainly focusing on the support of interactive data exploration and dynamic stimulus analysis.



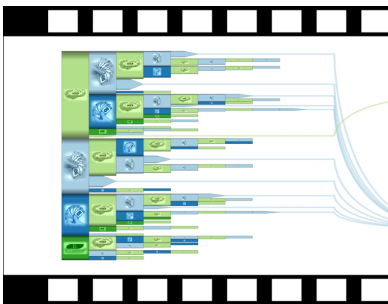
Space-Time Cube [16] The space-time cube represents *spatio-temporal* gaze data. The *2D* spatial gaze information is extended by time as the third dimension and data from *multiple participants* can be displayed. It provides a static overview of the data in *3D*, but the *context* of the stimulus is only visible by temporal skimming. *Interactive* kernel adjustment and clustering support an explorative analysis. The space-time cube is applied to *dynamic stimuli* with *passive* content.



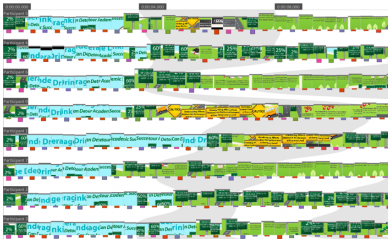
Motion-Compensated Heat Map [16] Motion-compensated heat maps present an approach to depict gaze data from *dynamic* stimuli on a *static 2D* heat map with no further *interaction* support. Optical flow is used to move gaze points with the objects the eyes are following.



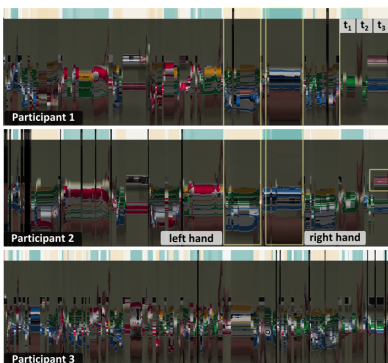
AOI Timelines and Scarf Plots [15] The visualization is *AOI-based* and depicts the *temporal* sequence of visited AOIs for *multiple participants*. A *2D static* overview shows color-coded AOI visits *not in-context* for *interactive* analysis. The techniques are developed for *dynamic 2D* stimuli with *passive* content. AOI timelines summarize the overall gaze distribution of all participants and the scarf plots depict each participant individually.



AOI Transition Trees [13] As another *AOI-based* technique, the transition trees depict *temporal* sequences of AOI visits from multiple participants. The specifications are identical to the AOI timelines. In contrast to the AOI timelines, the visualization depicts sequence frequencies of arbitrary length to identify common patterns.



Gaze Stripes and Fixation-Image Charts [23, 24] These two techniques are *point-based* for data from *multiple participants* and aim to preserve the *spatio-temporal* context of the data by incorporating thumbnails of the stimulus. Both visualizations are *static 2D* representations *in-context* of the stimulus and support *interactive* analysis. Gaze stripes focus on annotations of thumbnails for storytelling in eye-tracking protocols, fixation-image charts provide interactive filtering to detect and annotate patterns.



Gaze-Guided Slit-Scans [10, 18] Slit-scans are also *point-based* techniques with strong emphasis on image content of the regarded stimulus. Individual slit-scans show the scanpath of a *single* participant. However, similar to the other visualizations based on timelines, *multiple participants* can be compared by sorting of the timelines and dendrograms. Here, the interaction focuses on the exploration of different comparison metrics.



AOI Clouds [17] The last two techniques focus on a combination of two aspects that provide the biggest challenge in eye-tracking analysis: the examination of *active* stimulus content from *multiple participants*. AOI clouds provide a simple visualization of the time an AOI was watched in different recordings. *Interaction* is limited to fast navigation for all included data sources to watch the videos of important time spans.



Visual Analytics for Mobile Eye Tracking [29] While the former technique is based on AOIs, this approach aims at efficient annotation of AOIs, based on thumbnails. This is a *point-based* technique with image context, applied to *active* content from *multiple participants*. *Interactive* labeling is coupled with a direct analysis of the annotated results.

For the majority of the presented techniques, *dynamic* stimuli with *passive* content were investigated. Those techniques focus on different analysis tasks (Chapter 4.2.1). To provide a common dataset for the evaluation of new analysis techniques, a set of benchmark videos was recorded together with gaze data of people watching the videos.

4.4 Benchmark Data for Visualization Techniques

A benchmark of 11 videos was created to provide a dataset for the development and comparison of new visualization techniques [20]. As presented, the data analysis of spatio-temporal eye-tracking data in combination with video content was missing effective approaches for several analysis tasks. With this benchmark, data for research in this field is provided to the visualization community without needing an eye-tracking device to record new data. The content of the videos and the tasks are designed to induce typical viewing patterns. In this way, the benchmark data is designed to test a variety of analysis goals that one typically wants to perform with dynamic gaze data. To evoke these patterns, the content of the stimuli and the viewing tasks were controlled and a user study was conducted. In summary, the data suite consists of (a) 11 videos containing cars, persons, and card games, (b) raw gaze data from 25 participants, recorded with specific viewing tasks, and (c) AOI annotations for important objects in the videos with semantic naming.

Other Datasets

Outside the visualization and visual analytics community, there are some publicly available collections of datasets that contain stimulus material with eye-tracking data.

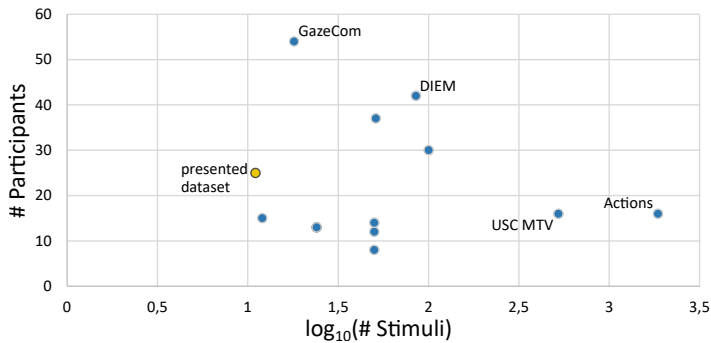


Figure 4.5: An overview of available datasets that include video stimuli and corresponding eye-tracking data (according to Winkler and Subramanian [308]).

Winkler and Subramanian [308] provide an overview of such data, summarizing 28 datasets of which 11 include video data with correlated eye-tracking data. Figure 4.5 displays how the presented dataset can be compared to these other video datasets regarding the number of stimuli and participants. These datasets were created with different intentions and objectives than the ones in this thesis.

For example, the datasets were designed for research on visual saliency [71, 204], video quality [34], and the natural viewing behavior for everyday video material [133, 212]. The tasks in these datasets include either one specific task per dataset and/or a free-viewing task. All datasets include gaze data from between 8 to 54 participants (*GazeCom* [96]). One exception is the *DIEM* dataset¹: since the overview of the datasets [308] was created, the number of records increased for some videos to more than 200 participants. The datasets *Actions in the Eye* [204] (1857 videos) and *USC CRCNS MTV* [71] (523 videos) contain the largest number of stimuli.

In the visualization and visual analytics community, two major datasets with gaze data were published: (1) Eye-tracking data for a user study on the readability of tree diagrams [67] and (2) a large dataset containing 393 different visualizations². However, this data is restricted to static stimuli and does not cover the dynamic content provided by the presented benchmark.

Viewing Patterns

The videos and tasks for the eye-tracking experiment were designed to evoke 3 different patterns that are most common in dynamic stimuli. These patterns either emerge from gaze distributed between different AOIs, or from gaze focused on individual AOIs.

¹ <http://thediemproject.wordpress.com>, last checked: October 13, 2018

² <http://massvis.mit.edu>, last checked: October 13, 2018

Table 4.1: The recorded stimuli with a description of the stimulus settings, the given tasks, and eye movement patterns that could be observed.

ID	Stimulus	Setting	Task	Induced Patterns
S1	Car Pursuit (0:25 min)	Panning camera follows a red car while it was going through a roundabout.	Follow the red car.	Potential smooth pursuit with long time spans of attentional synchrony on the red car.
S2	Turning Car (0:28 min)	Camera follows turning car. The movement of the car describes the shape of an eight.	Recognize the shape that is described by the movement of the car.	Attentional synchrony on the car with potential smooth pursuit eye movement.
S3	Dialog (0:19 min)	Two persons talk to each other in front of the camera.	Follow the dialog attentively.	Switching focus between the faces of both persons. Label on shirt (right person) attracts additional attention.
S4	Thimblorig (0:30 min)	A thimblorig with three cups and a marble.	Find the cup with the marble.	Attentional synchrony mainly on the cup with the marble.
S5	Memory (2:28 min)	A 4 × 4 memory game. Pairwise flipping of cards is performed until all pairs are found.	After one card is flipped, focus on the corresponding card of the pair.	Increasing attention on matching cards after several turns and switching focus during the search.
S6	UNO (2:01 min)	Two persons play UNO card game until the right player wins.	For each player's turn, focus on the playable cards on the hand.	Switching focus and attention mainly distributed between both hands and the stack of played cards.
S7	Kite (1:37 min)	Person on a meadow steers a kite. The kite repeatedly leaves the field of view.	Follow the flight path of the kite if possible.	Smooth pursuit if the kite is visible. Otherwise, the participants either tried to estimate the position of the kite, or focused on the person.
S8	Case-Exchange (0:27 min)	Various persons crossing the field of view while a text ribbon in the lower part is showing further information.	Task is provided by the text ribbon: Look for metal case.	Attentional synchrony on the text ribbon until the metal case appears and the task is readable.
S9	Ball Game (0:31 min)	Three players with orange shirts and one player with a white shirt pass a ball around.	Task group A: Count ball contacts of the white player. Task group B: Count passes between orange players.	Attentional synchrony often on the ball, independent from the task.
S10	Bag Search (2:13 min)	Various persons carrying different bags are crossing the field of view.	Look for a specific bag. Two groups (A,B) with two different search targets, presented before the video started.	Switching focus on new bags in the scene. Depending on the group, the search targets attract more attention.
S11	Person Search (2:52 min)	People with different clothing cross the field of view.	Task group A: Find the person with a hooded sweater. Task group B: Find the person with a red shirt and a headgear.	Switching focus on new persons. After identification, search targets become less important than new persons.

- **Switching focus:** The participants have to attend to various AOIs simultaneously. Since the task requires to distribute the gaze, the participants continuously switch their focus between AOIs. Although tracking of multiple objects is possible [73], retaining visual features is limited in this case and for identification tasks, the participants still have to focus on single objects.
- **Attentional synchrony:** The stimulus contains time spans with one AOI that attracts the attention of all participants, even if their eyes have been tracked separately. Attentional synchrony has been investigated for static and dynamic stimuli [268]; due to the high saliency of movement, it can frequently appear in dynamic scenes.
- **Smooth pursuit:** A moving AOI in the stimulus attracts the attention of the participants, causing them to follow its movement. In these time spans, smooth pursuit eye movement [149] can be present.

The above viewing patterns are canonical patterns that may occur in gaze data for dynamic stimuli. Since the induced patterns are often guided by the tasks given to the participants of the experiment, the effect of task dependency is also included:

- **Task groups:** The participants can be separated into groups of similar viewing behavior by assigning them different tasks. This type of pattern is of special interest for new methods that compare scanpaths to identify clusters of participants. Differences between groups may also be based on the participants' background or condition (not included in the dataset); for example, one could investigate differences in the scanpaths of healthy and mentally disordered persons.

To induce these viewing patterns, video scenarios with according viewing tasks were designed. Table 4.1 summarizes the stimuli settings and tasks, as well as the patterns produced. The stimuli from this dataset are applied for showcasing the techniques presented in Chapter 5. For the visualizations designed for active stimulus content, additional datasets are investigated that fit the requirements.

Analyzing a Single Video and Multiple Participants

This chapter covers contributed techniques for mainly dynamic stimuli without the active intervention of the participants. This means that people sat in front of a monitor and watched a video. They could not intervene with the stimulus. After recording, the data was synchronized with the presented video to provide the data for analysis.

For the support of the analysis tasks discussed in Chapter 4.2, a set of visualization techniques was developed. According to the taxonomy for eye-tracking visualizations, the techniques are either point-based or AOI-based. For a visual analytics approach on eye-tracking analysis, the techniques are combined in the *ISeeCube* framework to provide a comprehensive view of the data from different perspectives.

- **The point-based techniques** (Chapter 5.1) presented in this chapter focus on the overview of commonalities in eye-tracking datasets. With respect to analysis tasks considering *where* and *when*, the visualizations help identify attentional synchrony and interpret the overall gaze distributions.
- **The AOI-based techniques** (Chapter 5.2) complement these visualizations by views that provide details about individual AOIs and participants. For visual analytics support, all visualizations are linked and provide automatic processing methods to emphasize specific aspects of the data.

Although not included in *ISeeCube*, the third category of techniques provides a complementary view on the data that is worthwhile investigating:

- **The image-based techniques** (Chapter 5.3) contribute to a specific type of point-based visualizations that take the image content of single video frames into account. The techniques aim to provide more contextual information than other point-based methods.

This chapter is partly based on the following publications:

- K. Kurzhals and D. Weiskopf. “Space-Time Visual Analytics of Eye-Tracking Data for Dynamic Stimuli”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2129–2138 [16]
- K. Kurzhals, F. Heimerl, and D. Weiskopf. “ISeeCube: Visual Analysis of Gaze Data for Video”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2014, pp. 43–50 [15]
- K. Kurzhals and D. Weiskopf. “AOI Transition Trees”. In: *Proceedings of the Graphics Interface Conference*. 2015, pp. 41–48 [13]
- K. Kurzhals, M. Hlawatsch, F. Heimerl, M. Burch, and D. Weiskopf. “Gaze Stripes: Image-Based Visualization of Eye Tracking Data”. In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 1005–1014 [24]
- K. Kurzhals, M. Hlawatsch, M. Burch, and D. Weiskopf. “Fixation-Image Charts”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2016, pp. 11–18 [23] 🏆
- K. Kurzhals and D. Weiskopf. “Visualizing Eye Tracking Data with Gaze-Guided Slit-Scans”. In: *Proceedings of the IEEE Second Workshop on Eye Tracking and Visualization (ETVIS)*. 2016, pp. 45–49 [18]
- M. Koch, K. Kurzhals, and D. Weiskopf. “Image-Based Scanpath Comparison with Slit-Scan Visualization”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2018, 55:1–55:5 [10]

5.1 Point-Based Visualization of Gaze Distributions

For a first overview of the data, point-based techniques have the advantage to be directly applicable, without annotations necessary. For this purpose, two developed techniques summarize gaze data of dynamic stimulus content: (1) *Motion-compensated heat maps* summarize gaze points adjusted to the motion of an object the gaze was following. (2) The *space-time cube* provides an overview of the data that facilitates understanding the spatio-temporal gaze distribution. While motion-compensated heat maps are meant to summarize short time spans, the space-time cube can be applied to long durations, as shown for trajectories [297] and video analysis [245].

5.1.1 Motion-Compensated Heat Map

Motion-compensated heat maps introduce a new approach to summarize eye-tracking data of dynamic stimuli. A motion-compensated heat map shows high values for observed objects in motion. For example, imagine an object moving through the video from the left to the right side. Assuming all viewers would always observe the object, the resulting heat map of this time span would show a uniform distribution along the movement trail of the object. This is helpful to visualize trajectories, but it conceals the fact that all gaze points were on the object while it was moving. In contrast, the motion-compensated heat map would show high values only on the object that was observed, indicating the high amount of attention spent on it. The creation of a motion-compensated heat map can be described by particle tracing in a time-dependent vector field [305] as follows:

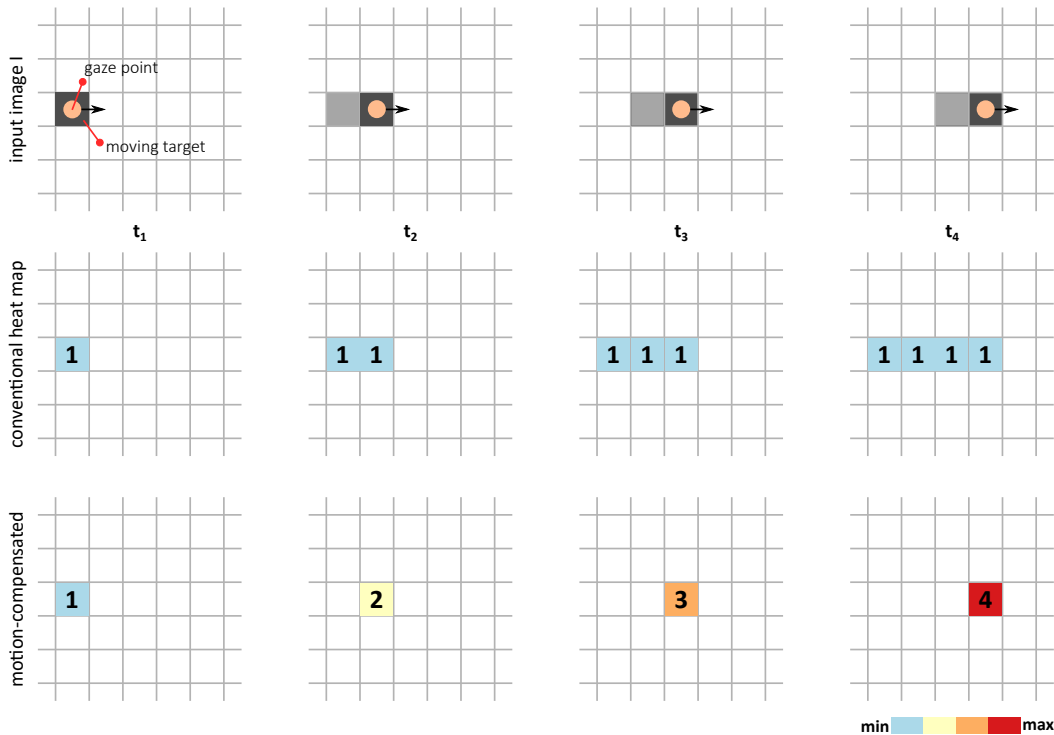


Figure 5.1: Schematic example of a moving target followed by a gaze point. In a conventional heat map, all gaze points are aggregated at the positions they appear. With a motion-compensated approach, the gaze points move with the target, creating a hotspot on the target.

1. The optical flow between consecutive frames in the video is calculated. It is described by a time-dependent vector field.
2. The analyst defines a time span to summarize.
3. A keyframe within this time span is picked. It defines the end for the particle tracing and serves as a representative for the sequence.
4. Each gaze point within the time span is traced along the flow until the keyframe position is reached. If the keyframe is not the last frame of the selected time span, the tracing is performed backward until the keyframe is reached.
5. The traced end positions are used to create a heat map that is blended together with the representative keyframe.

Figure 5.1 depicts an example of a sequence of four time steps. A target object moves from the left side of the screen to the right and a gaze point follows this object with a smooth pursuit. For simplicity, the last time step t_4 is also assumed to be the representative keyframe, so no backward tracing is necessary. In a conventional heat map, gaze points are distributed along the trajectory the eyes move. As a result, the heat

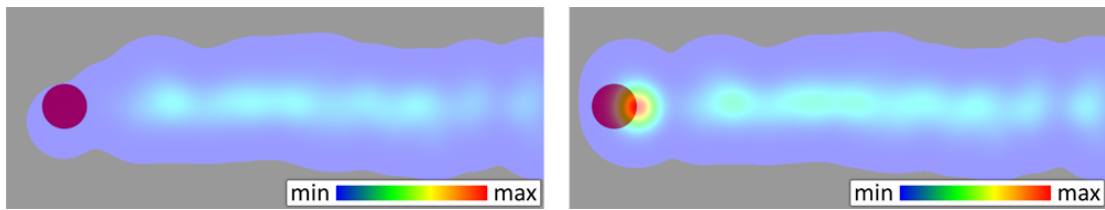


Figure 5.2: A conventional heat map (left) and a motion-compensated heat map (right) of a red circle that moves from right to left.

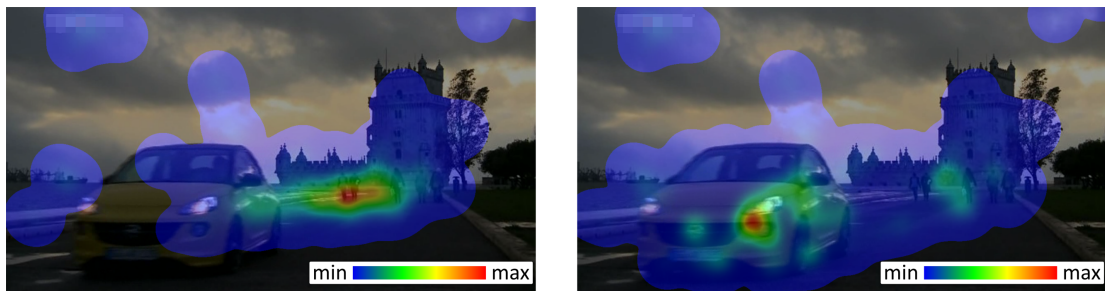


Figure 5.3: A car drives from the tower to the left side of the screen. The conventional heat map (left) provides only little useful information about the dynamic AOIs and could lead to misinterpretations because the hot spot lies on two persons. The motion-compensated heat map (right) conveys the information on which object (the car) most of the attention was spent.

map shows the gaze distribution but does not emphasize the fact that all gaze points focus on the object. In the motion-compensated heat map, this effect is emphasized by moving the gaze points with the motion of the object.

For a video example, Figure 5.2 shows a comparison between a conventional and a motion-compensated heat map. In the video, a red circle moved from right to left. The participants were asked to follow the circle during its movement. The measured data is distributed along the motion path and heat map values on the circle are low, showing no hotspots. The motion-compensated heat map transports the majority of the data points along the optical flow, showing the hotspot with the highest value on the circle itself. The motion path can still be recognized, providing summarizing information about the movement and which object was attended to.

Figure 5.3 shows a real-world example: Both heat maps represent a short sequence (about 7 sec) with a driving car and five persons in the background. In this sequence, the car receives most of the attention. Due to the dynamic changes in the scene, the conventional heat map is hard to interpret and the existing hotspot seems to lie on the background, which would be a misinterpretation. The motion-compensated heat map adjusts the data points along with the object movement, the hotspot lies on the car.

5.1.2 Space-Time Cube

A *space-time cube (STC)* describes a three-dimensional space, consisting of two data dimensions and time as the third dimension. In most scenarios, the data dimensions comprise a 2D spatial context. In the case of eye tracking, this is the coordinate system of a stimulus. The STC is used in various fields of research. Gatalsky et al. [122] describe its application to event data in a geographical context. Chen et al. [80] and Botchen et al. [58] represent video content in 3D to depict individual motion events. This thesis adopts the representation of videos in a STC and adds visualizations for gaze data.

In the context of eye tracking, Li et al. [188] describe the use of the STC to visualize eye trajectories. The authors focus on the analysis of static stimuli. For the application to dynamic stimuli, Duchowski and McCormick [100] describe a space-time representation of *volumes of interest* for aggregated gaze trajectories. The presented approach in this thesis extends the concept for dynamic stimuli and provides different data representations in addition to the mentioned eye trajectories.

The approach applies the clustering of gaze points and a 3D representation of cluster hulls in the STC. This helps find important AOIs and interpret dynamic changes in the distribution of gaze points. Clustering of eye-tracking data is already used when fixations are identified in raw data. Salvucci and Goldberg [253] describe a taxonomy for different fixation identification algorithms. For the clustering of multiple user gaze data, Sawahata et al. [256] and Mital et al. [212] use a Gaussian Mixture Model. The presented approach uses the mean shift algorithm for the clustering of gaze data [255] because it is robust to noise and does not require a preset number of clusters. If available, the approach respects shot boundaries from a shot detection algorithm.

The key contribution of this work is a unified analysis approach for gaze data in the spatio-temporal context of the stimulus. The presented approach includes means to ease the analysis of eye-tracking data in the STC, i.e., gaze point representation with color-coded highlighting of attentional synchrony; shot-based navigation through the STC for edited video content; and cluster analysis for the efficient identification of potential AOIs.

STC Visualization

Gaze data is depicted with different representations. Since the focus of this work is on the overview of the data, suitable summarizing visualizations are necessary. Figure 5.4 shows three techniques that are implemented to represent scanpath trajectories, raw gaze data, and clustered gaze points.

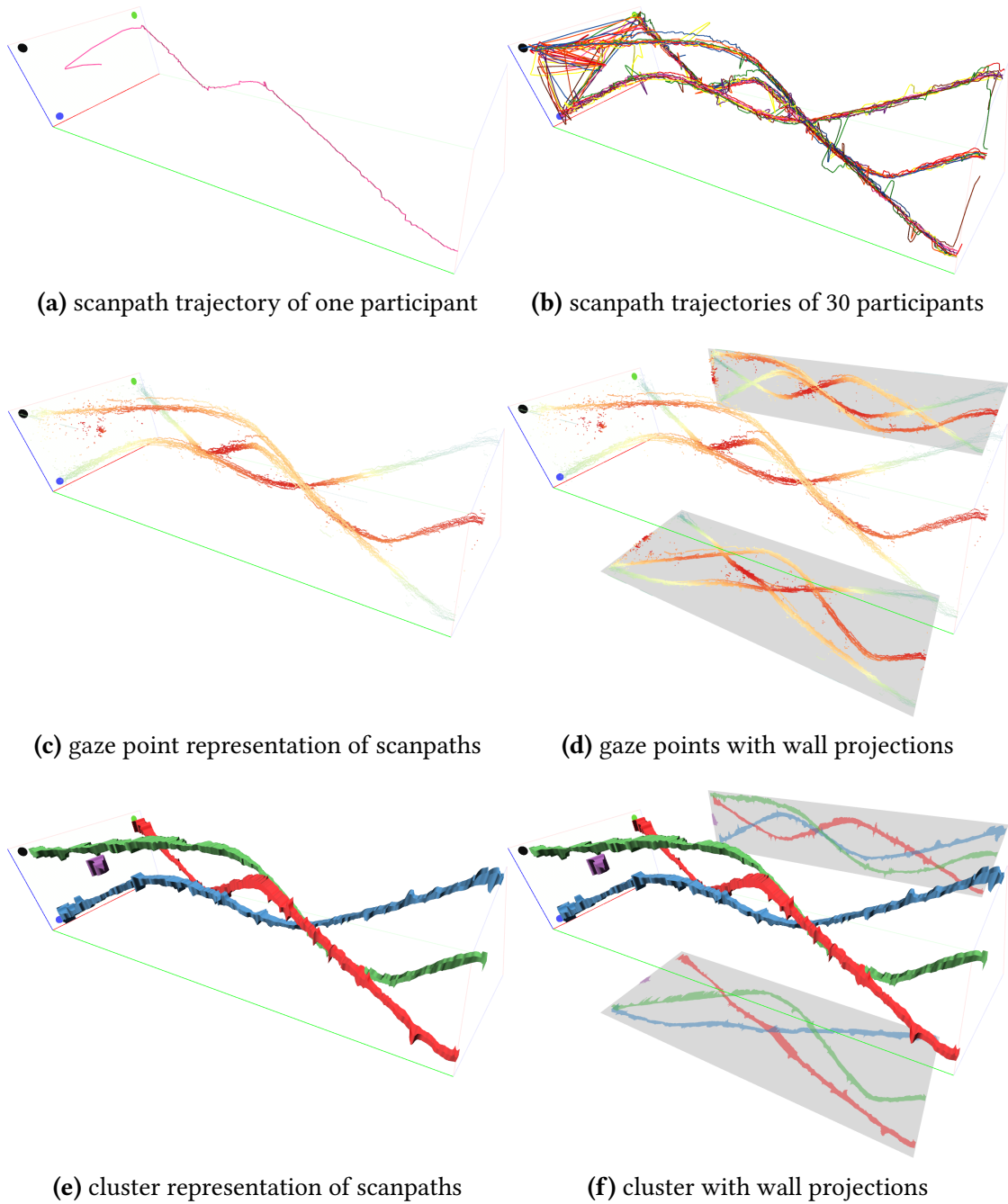


Figure 5.4: Depiction of gaze data from a video with three moving dots in the space-time cube.

Scanpath Trajectory

The scanpath of an individual participant is depicted by a 3D trajectory in the STC (Figure 5.4a). The comparison of a few participants can be performed this way, but with an increasing number of scanpaths, this technique tends to create visual clutter (Figure 5.4b). For the investigation of individual participants, there are more efficient visualizations that will be discussed in the following chapters. For an overview, a representation of raw gaze points with additional filter options is more convenient.

Gaze Points

In 2D, the simplest representation of gaze data on the stimulus is the bee swarm. It shows individual points superimposed on the video. Translating this visualization in the STC results in a 3D point cloud (Figure 5.4c) that gives an impression of the data distribution including the attentional synchrony between participants. To further highlight such time spans, the distance d of each point to the center of mass per frame is determined to calculate the value ν :

$$\nu(d) = e^{-0.5\left(\frac{d}{\sigma}\right)^2} \in [0, 1]$$

The value ν defines the transparency and the color of a data point. By reducing the kernel size σ in the parameter controls, sparse data points in the space-time visualization fade out, facilitating the identification of dense regions. Data points with a red color indicate a distance close to their frame's center of mass. When many viewers looked simultaneously at a small area, a large number of data points appear red and remain even when the kernel size is reduced. This representation can also reveal motion patterns of objects tracked by several viewers, for example, in cases where participants follow moving dots as depicted in Figure 5.4.

3D visualizations can be afflicted with perceptual issues resulting from occlusion, distortion, and inaccurate depth perception. To address these problems, the scene camera is adjustable in order to resolve possible occlusions in the STC. Further, the idea of 2D wall projections (Figure 5.4d) was adapted from ExoVis, introduced by Tory and Swindells [289]. With an adjustable scale and distance to the STC, the walls represent 2D overviews of the data without being occluded by the main visualization.

Clustering

In contrast to clustering algorithms for the detection of fixations from a single participant, clustering data from multiple participants helps identify interesting regions (i.e., AOIs) in a dataset. A clustering algorithm should fulfill the following requirements for the detection of potential AOIs:

- ▶ **Unknown number of clusters:** The number of data points to cluster can vary, depending on two factors: the number of participants for whom data was recorded; and the length of the stimulus presentation. Defining a proper number of clusters is not intuitive, even if these factors are known.
- ▶ **Parameterization:** A parameterizable clustering approach allows the user to define the granularity of the clusters. Hence, the adjustable parameters have to be intuitively understandable. The algorithm should depend on two controllable parameters that determine the spatial and temporal extents of the clusters.

The mean shift algorithm performs without a preset number of clusters and can be parametrized in space and time independently. Therefore, it fits the requirements and is suitable for clustering the data. Mean shift clustering is widely used for feature space analysis in the field of computer vision [86]. Santella and DeCarlo [255] introduced its application to eye-tracking data.

The algorithm by Santella and DeCarlo is adopted and extended to take into account shot boundaries. The viewers' gaze direction can be influenced by abrupt cuts [71, 295]. Hence, the detection algorithm described before in Chapter 2.1.2 is applied to identify these boundaries and clustering is handled separately for each shot. In its basic implementation, the mean shift algorithm moves points towards local centers of mass until convergence. If applied to spatio-temporal data, this can interrupt the temporal coherence of sequences such as smooth pursuits. Hence, moving points is only applied in the spatial dimension, resulting in better separable patterns that are clustered with *Density-Based Spatial Clustering of Applications with Noise (DBSCAN)* [107], with a fixed parameter set for distance and neighborhood values.

Cluster hulls (Figure 5.4e, 5.4f) depict the extracted clusters. Axis-aligned boxes around all data points of a cluster for every time step represent the most common convention for AOI representation. The boxes are connected after applying an exponentially weighted moving average [158] to their size, in order to provide a smooth transition between time steps. The boxes create a hull around the data points in a cluster. The spatial extent provides information about changes in the spatial distribution of points over time: “thick” cluster hulls correspond to a wide spread of points and, thus, low density—and vice versa. By projecting the cluster hull of a time step to the corresponding video frame, dynamic AOIs provide information about the distribution of gaze on different regions or objects. The cluster size is measured by the number of data points it contains.

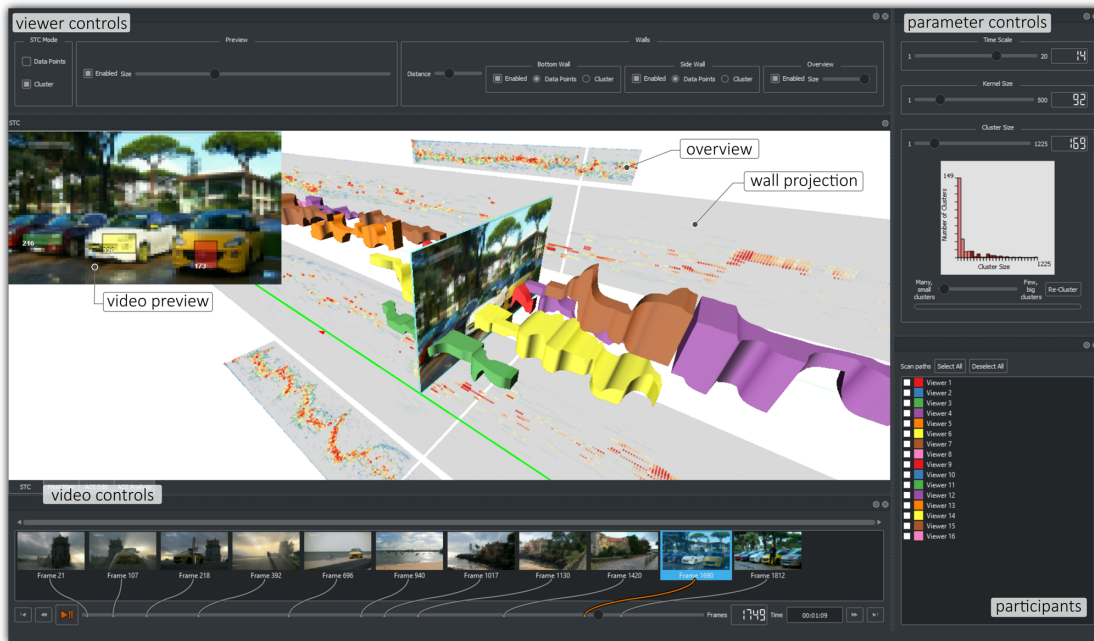


Figure 5.5: *ISeeCube*: The *viewer controls* adjust the size and content of the *video preview* and the *wall projections*. *Video controls* support the navigation. The *parameter controls* allow filtering of data points, cluster results, and individual scanpaths of *participants*.

Visual Analytics Framework: *ISeeCube*

The designed approach comprises a main view showing the STC and a video preview (Figure 5.5) in the framework *ISeeCube*. Additional control panels for the visualization, video navigation, and parameters are freely arrangeable around the main view. A video plane along the spatial dimensions inside the STC represents the current video frame. It is freely rotatable and movable to investigate the data around it. With the video controls, the analyst can navigate through the video with the time slider, frame-wise navigation, shot-boundary frames, or the playback function. Changing the frame position translates the STC relative to the video plane along the time axis, providing an easy method to analyze selected time spans. In the context of video analysis, the time axis typically shows the highest visual expansion. Therefore, scaling the time axis enables the user to explore the data as an overview as well as in detail. Shot boundaries are depicted by red arrowheads on the time axis of the STC. In the video controls, a keyframe represents the boundary. By picking one of the keyframes, the space-time visualization jumps to the corresponding position on the time axis, providing an efficient method to examine shot changes. This design supports multiple coordinated views [244] to show the different aspects of spatio-temporal eye-tracking and stimulus data. AOI-based implementations to extend *ISeeCube* will be discussed in Chapter 5.2.

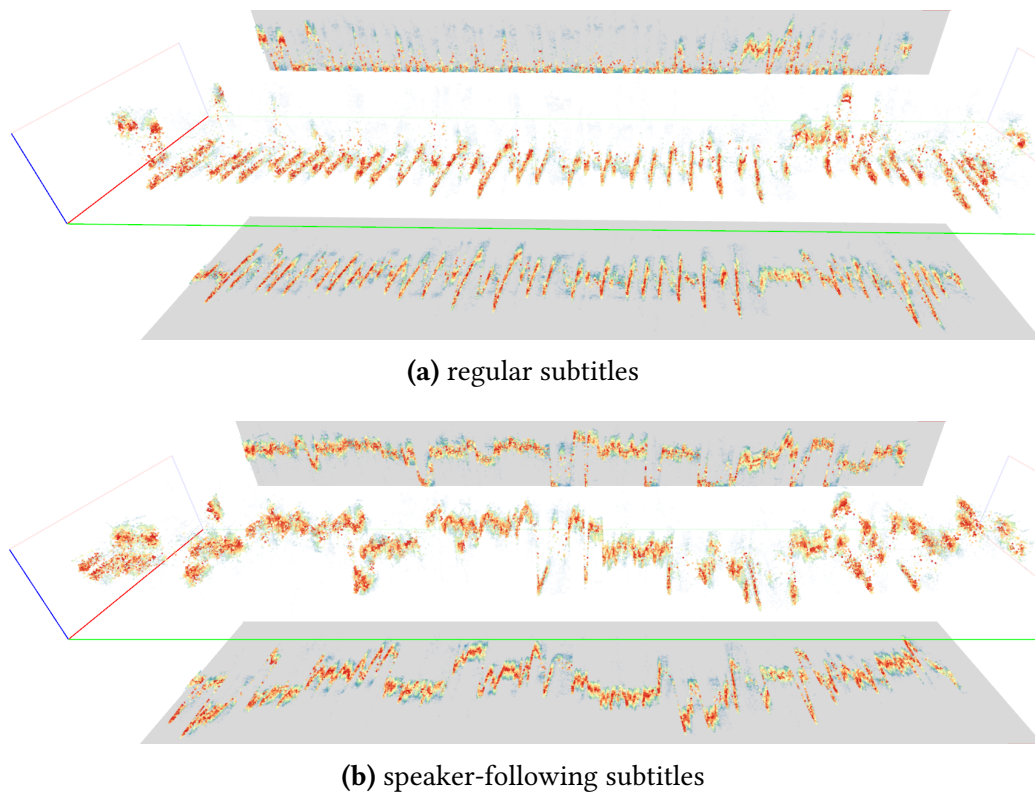


Figure 5.6: STC comparison of two subtitle layouts and their influence on gaze distribution.

5.1.3 Example: Subtitle Layouts

Coming back to the eye-tracking study described in Chapter 3.4.3, it was shown that speaker-following subtitles significantly influence the gaze distribution on text and faces in comparison to regular subtitles. With heat maps, some exemplary shots can be compared for the subtitle layouts, but the general overview of the data is missing.

Figure 5.6 depicts both layouts in the corresponding STC visualization, showing the gaze data from 20 participants each. Regular subtitles (Figure 5.6a) evoke a specific pattern that is clearly visible over the whole time. Since participants have to look at the bottom of the screen and read one or multiple lines of text, horizontal patterns arise. These patterns remain if the data is filtered for attentional synchrony of all participants. The upper parts of the video with the actual image content is less visited and gaze points show not much synchrony. In contrast, the speaker-following subtitles (Figure 5.6b) lead to a shift of gaze points to the upper part of the image. Due to the proximity of speech bubbles and faces, participants can switch more efficiently between both AOIs, which leads to significant changes in the statistical measures. Occasional appearances of gaze points at the bottom result from subtitles that could not be presented by speech bubbles due to an off-screen speaker.

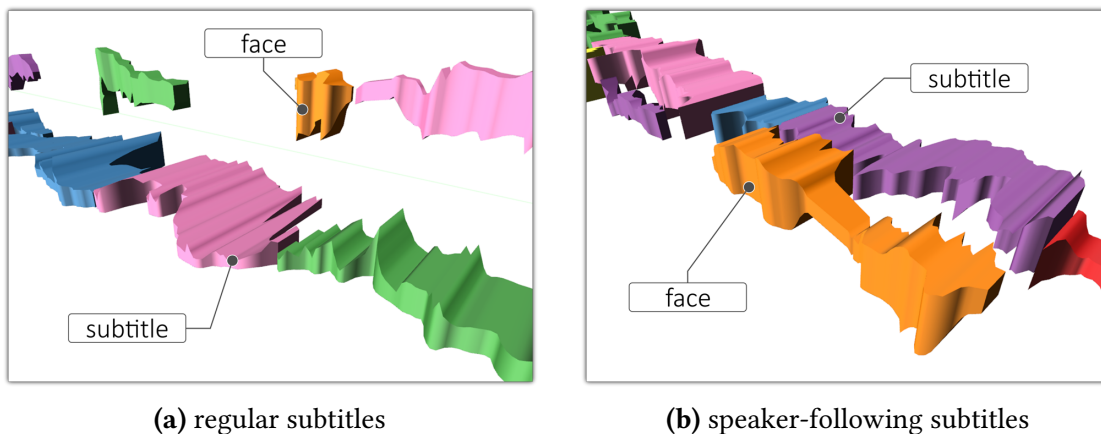


Figure 5.7: Regular subtitles lead to flat clusters at the bottom, with few clusters on faces. For the alternative layout, clusters on faces and subtitles are closer and evened.

Regarding the clusters derived from both layouts, the differences also become obvious (Figure 5.7). The horizontal pattern in the regular subtitles leads to long and flat cluster hulls at the bottom of the scene (Figure 5.7a). The small number of gaze points on the faces results in scattered clusters with short temporal coherence. With speaker-following subtitles, the clusters on faces and subtitles are close to each other (Figure 5.7b). Due to the increased number of gaze points on face regions, clusters show a larger extent along the temporal axis. These changes in the cluster structure help identify face AOIs by one cluster, while regular subtitles often lead to several clusters for one AOI.

For this example, the STC supports statistical results by showing that the measured differences are also apparent in the resulting visual patterns. Especially for the analysis of pilot studies, exploration in this way helps formulate hypotheses for a user study.

5.1.4 Discussion

Applying the gaze point and cluster visualization, important analysis tasks can be covered, mainly considering *where* and *when* questions (Chapter 4.2.1). The motion-compensated heat map focuses on the aspect *where* participants looked and aims to compensate temporal changes for a static result picture. Except for the compensation of temporal changes, the visualization has the same shortcomings as regular heat maps. In particular, the heat map shows a strongly aggregated view on the dataset that limits the application to other analysis tasks. The STC improves on the data overview by explicitly showing the temporal dimension in a static visualization. Spatio-temporal patterns can be identified and investigated directly, without the need to watch the whole video again. The temporal scalability of the approach is sufficient even for long-term eye-tracking data. Since the publication of this work, the approach was also extended to another

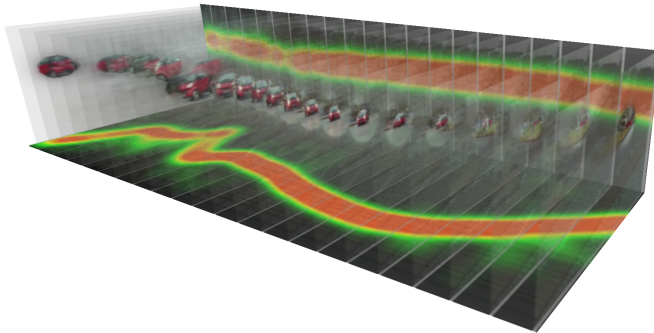


Figure 5.8: Volume visualization of the *Car Pursuit* video. The wall projections show the calculated density field and slices of the stimulus are displayed in the space-time volume. Such a representation provides a spatio-temporal overview and helps identify *what* happened faster than the point-based representation.

data domain with a larger dataset than the presented eye-tracking experiments. The STC was included as a complementary view for the analysis of visitor behavior for indoor event management, handling geo-located data from multiple days [11].

Volume Visualization for Eye Tracking One shortcoming of the presented space-time cube is the missing context of the stimulus. Although interesting time spans are easy to spot, the analyst has to adjust the slider to see the respective frame from the video. This can be compensated by combining techniques from volume visualization with the presented image-based techniques (Figure 5.8). A dynamic heat map can be interpreted as a spatio-temporal volume that consists of calculated gaze point densities. Representing this volume directly would provide information similar to the point representation. Mapping the density to alpha values for slices of the stimulus results in a visualization that shows when many participants looked at the same region and what they were looking at. This work could be further extended by reviewing techniques for volume visualization (e.g., transfer functions, segmentation algorithms) and their applicability to videos with eye-tracking data.

For questions considering *who* showed interesting behavior and for comparison tasks, AOIs are necessary. Although clusters could be used as AOIs, a semantically rich analysis requires more annotation of the data.

5.2 AOI-Based Scanpath Analysis

Based on the visual analytics approach of *ISeeCube*, AOI annotations expand the analysis framework with a multitude of new possibilities to answer questions that are hard or impossible to investigate with point-based techniques only. In particular, with the advantage of multiple coordinated views, timelines with information about AOIs (Chapter 5.2.2) and participants' scarf plots (Chapter 5.2.2) can be synchronized with other visualizations such as the STC. Furthermore, AOI transition trees (Chapter 5.2.3)

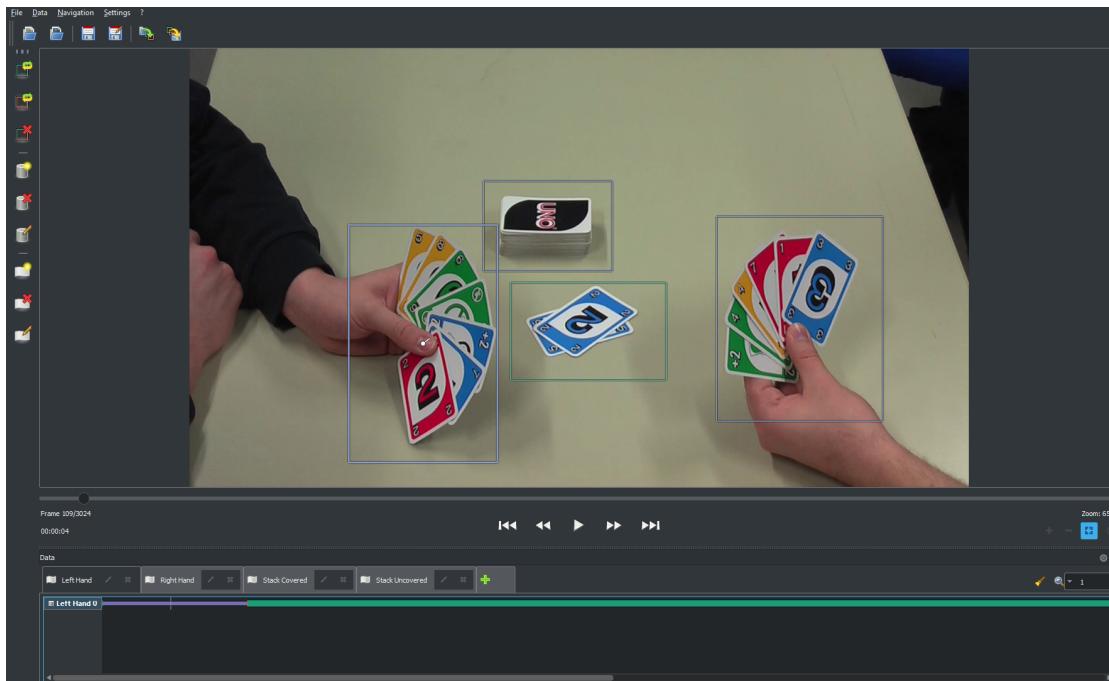


Figure 5.9: Editor for the definition of dynamic, axis-aligned AOIs. Individual categories allow for a semantic differentiation of annotated objects. In the presented example, the AOIs are categorized as *left hand*, *right hand*, *covered stack*, and *uncovered stack*.

provide an overview of AOI transition patterns coupled with scarf plots to highlight selected patterns in the scanpaths. For AOI annotations in the video, *ISeeCube* contains an editor to define rectangular bounding boxes for important objects in a video.

5.2.1 AOI Editor

For dynamic stimuli, the definition of AOIs that adjust to the changes of moving objects becomes an important step in the analysis process. For the presented examples in this thesis, an editor (Figure 5.9) was developed that allows for the definition of dynamic, axis-aligned bounding boxes to mark AOIs and define categories to specify the analysis. With the information provided by cluster analysis, the analyst can identify the most important objects and areas in a video and annotate them with the editor. The analyst creates new AOIs by drawing bounding boxes in the video at key positions during playback. Between key positions, the bounding boxes are interpolated linearly. Successive IDs are used for new AOIs, independent from their category.

The STC can also show selected AOIs (Figure 5.10). In contrast to the depiction of clusters as solid hulls, the changes of an AOI are depicted by space-time trajectories at the four corner points of the bounding box. In combination with the data points or

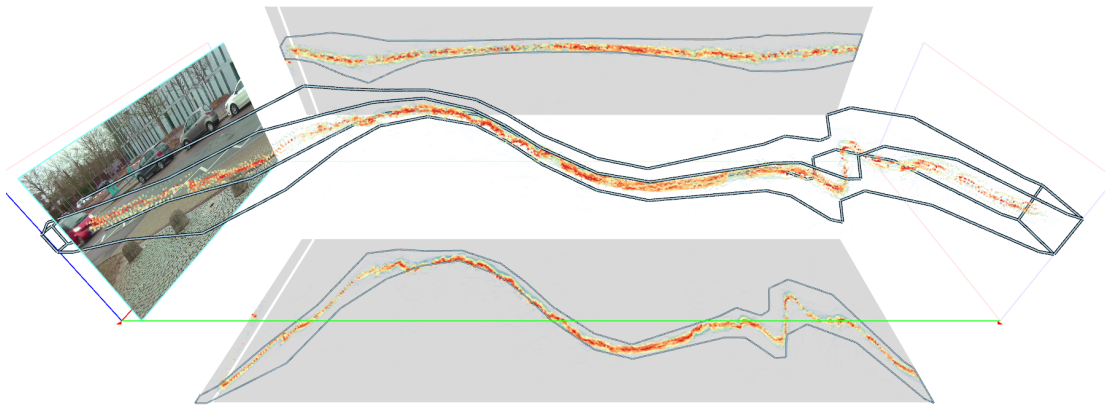


Figure 5.10: The AOI of a driving car in the STC. The spatio-temporal changes in position and size are visible by the shape that is formed by the outlines. With the data points, it is easy to spot when participants looked at the AOI.

clusters, one can see when participants looked at the AOI by investigating data that lies inside the spatio-temporal shape. Vice versa, position offsets due to calibration issues are easy to spot because they would result in data points close to the AOI shape, but not inside of it. If a data point lies inside an AOI, a label is assigned for further processing. Such labeled data will be applied for the following visualizations.

5.2.2 AOI Timelines and Scarf Plots

The representation of AOIs in the STC provides valuable information about the spatio-temporal extent of individual objects. However, due to occasional overlaps, displaying all AOI representations simultaneously leads to visual clutter. With AOIs, quantitative research on the data can be supported by alternative visualizations that abstract from spatio-temporal information and represent gaze data with semantic meaning.

Related Work

With timelines, the clutter problem is reduced by abstracting the visualization to the temporal component of the data and providing the spatial information only on demand in the STC. Timeline representations are a common method to visualize the temporal progression of events. André et al. [39] designed detailed timelines for hierarchies, relationships, and scale. This principle is adapted to provide additional AOI information on demand and adjust the presented information to the special requirements of gaze data recorded from videos. In the field of eye tracking, Andrienko et al. [40] use horizontal segmented bars in a temporal view to visualize the distance of eye trajectories to selected AOIs of static graphs. Ristovski et al. [243] show fixation time series with a highlighting function for fixations on the same AOI. These papers focus on static stimuli.

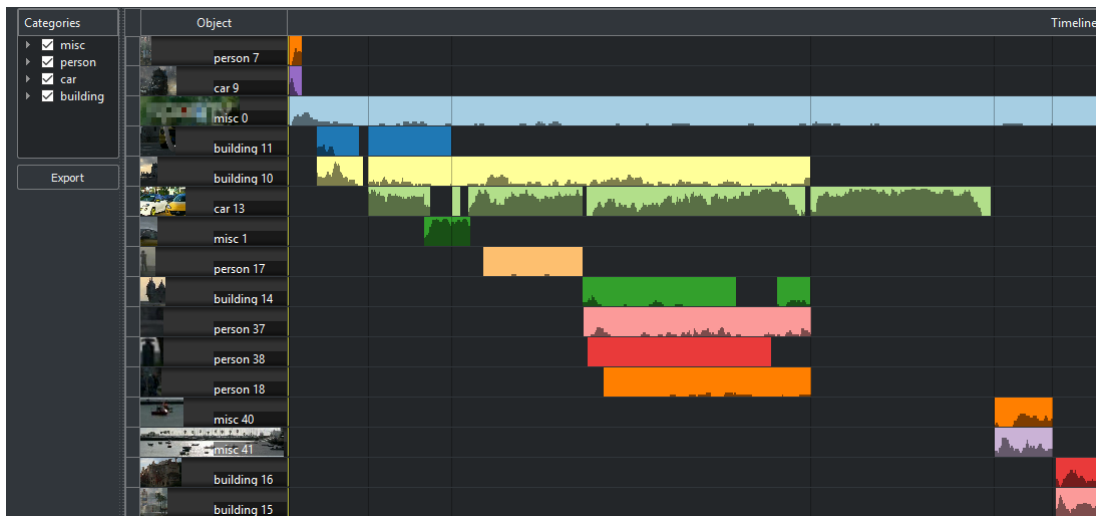


Figure 5.11: With the category tree (left), individual AOIs can be removed from the visualization. AOIs are presented on separate timelines (right) ordered by their first appearance. Colored bars with histograms indicate when an AOI was visible and how many participants looked at it.

For the analysis of dynamic stimuli, Richardson et al. [242] used scarf plots to visualize the recurrence of eye movements between two persons. Weibel et al. [303] integrate mobile eye-tracking data in ChronoViz, a tool to visualize multiple streams of time series data simultaneously. They use separate scarf plots for individual AOIs and concentrate their analysis on individual viewers; a similar approach is presented by Lessing and Linge [186]. Stellmach et al. [273] introduce a models-of-interest timeline that shows a viewer’s gaze distribution between various 3D objects in a virtual environment with individually selectable colors for each object. In their work, the main focus lies on the visualization of a single viewer’s gaze data over time with a constant set of objects.

In the presented approach, two types of timelines are derived from the data, one that focuses on the AOIs themselves, and scarf plots that focus on individual participants. AOI timelines convey the information of the chronological appearance of AOIs and the temporal distribution of attention.

AOI Timeline Visualization

Similar to the STC visualization, the AOI timeline provides an overview of the complete dataset, but without the spatial information (Figure 5.11). All AOIs are represented by rows, ordered by their first appearance in the video. The first column shows the name and a representative image of each AOI. The second column shows a colored bar for each time span in which the AOI exists. A histogram in the colored bar shows the gaze distribution of all participants. To distinguish between different AOIs, a qualitative

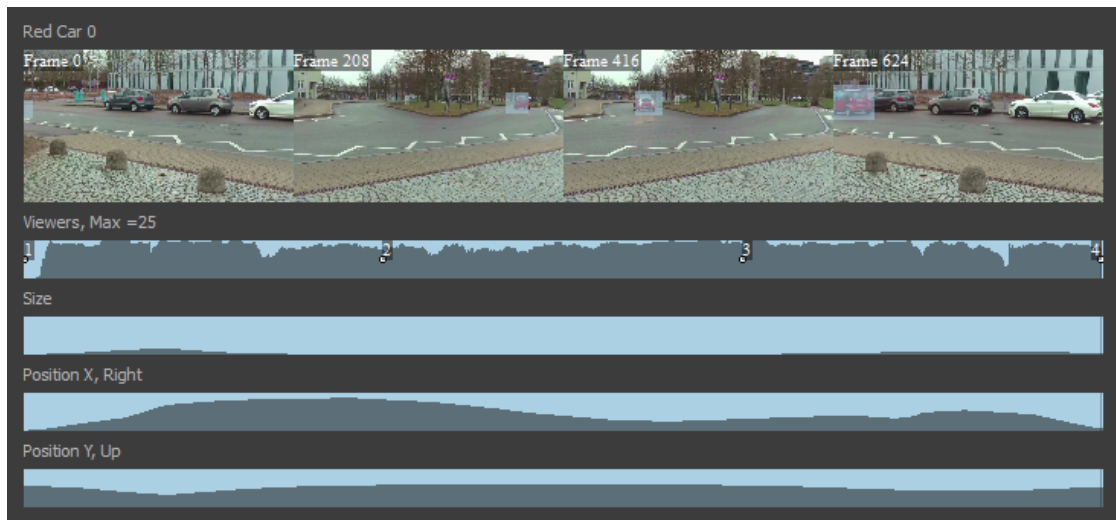


Figure 5.12: The overview provides a film strip of the selected AOI and histograms for viewers' gaze distribution, size, and X-/Y-position.

color scheme of 11 colors [137] was used. A color is locked to an AOI as long as the AOI exists and can be mapped to another one as soon as the respective AOI disappears. This strategy ensures an unambiguous mapping from AOIs to colors, as long as there are fewer than 12 AOIs with overlapping life spans. For additional AOIs, the color scheme is repeated and possible ambiguities have to be solved by looking at the histograms and the video preview.

Ordering the AOIs by their first appearance results in a timeline where early appearing objects are placed in upper rows and late-appearing objects in lower rows. This leads to problems when objects appear early and reappear several times in the video. In this case, a gap of empty rows occurs between the late appearing objects at the lower rows and the reappearing object in the upper row. Comparing the histograms of the involved AOIs becomes more difficult the farther the rows are apart. To solve this problem, a tree view left to the AOI timeline (Figure 5.11) shows all objects ordered by their category. Either the complete category or individual objects can be disabled to hide them in the timeline. With this approach, users can exclude all objects that are not present in the currently investigated time span to concentrate on the relevant information.

To obtain additional AOI information on demand, each row can be selected individually to show an overview (Figure 5.12). Each overview is presented in a separate window and can be activated as needed. The information provided by the overview consists of a filmstrip and four histograms. The filmstrip shows representative frames from the time span of the marked AOI. The frames are chosen by dividing the time span into four equal parts. If an AOI does not exist in one of the inner frame positions, the

parts are further divided until a valid frame is reached. The histograms show the gaze distribution, AOI size, and AOI position:

- **Gaze distribution:** The histogram is the same as in the timeline. The numbers mark the position of the frames from the filmstrip.
- **Size:** The size of an AOI is measured as area size relative to the video resolution. In Figure 5.12, the size histogram shows several time spans where the size of the AOI increases at the end of a shot. In combination with the information provided by the position histograms, one can interpret that the car was moving close to the camera in these situations.
- **Positions:** X- and Y-coordinates are measured at the bottom-center point of the AOI. In the histograms, high values represent a position in the right part of the scene and in the upper part, respectively.

This visualization provides an overview of the temporal distribution of gaze points on the AOIs. It is an aggregated representation of all participants' viewing behavior. For the comparison of individual participants, additional information is required. Hence, the scarf plot visualization is incorporated as an additional view, synchronized and linked with the AOI timelines.

Scarf Plots

The scarf plot of a participant is created by investigating the individual scanpath. If a gaze point in a frame is considered to be in an AOI, the corresponding color of the AOI is used to mark this frame in the plot. If either no gaze point is available, or cannot be assigned to an AOI, the frame is marked black (Figure 5.13). A gaze point is assigned to an AOI when it lies inside the bounding box of the AOI in the respective frame. Due to the dynamic content of video stimuli, overlaps between AOIs are often inevitable. Two common methods to handle this problem are either to distribute the value between the overlapping AOIs, or to calculate the distances between the gaze point and the involved AOI centers and assign the point to the AOI with the shortest distance [149]. In the presented work, the latter method is applied, since ambiguities are not supported by the common string comparison algorithms that are used for scanpath analysis.

The AOI timeline and the scarf plots are synchronized and the user can see directly which AOIs are involved in the current time span and what object they represent. As performed for string representations of scanpaths, each participant in the dataset is represented by a mapping of the gaze data to the annotated AOIs. These strings can be used for interactive scanpath analysis, whereas the scarf plot visualization allows for an easy interpretation of the results.

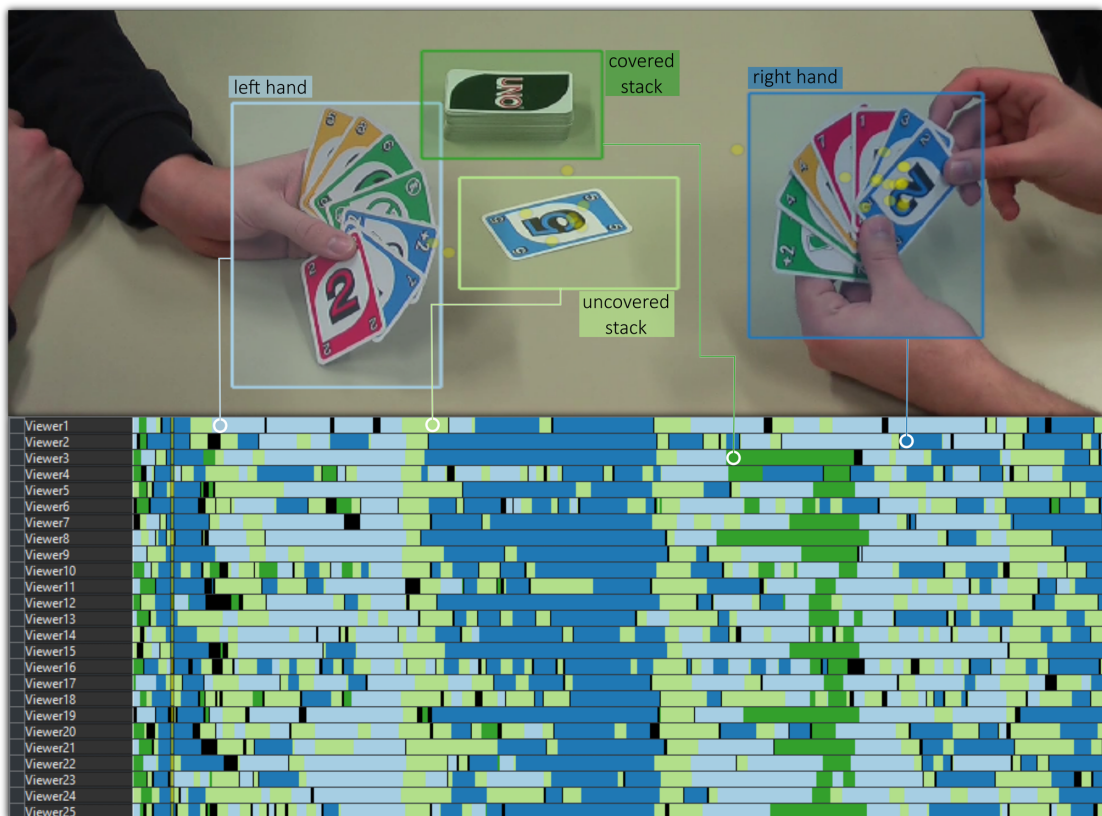


Figure 5.13: Scarf plots of 25 participants watching the video of a *UNO* card game. The individual viewing behavior of each participant is visualized by colored timelines that indicate which of the four AOIs was investigated at the current point in time.

Scanpath Comparison and Cluster Analysis

ISecCube integrates automatic processing of eye-tracking data to assess the similarity of scanpaths. Three different similarity functions can be applied with agglomerative hierarchical clustering to identify groups of similar behavior. Users can thus explore different facets of the dataset and select the distance function that fits their objectives and analytical goals best. The similarity functions available are the Levenshtein distance, a function based on gaze distribution, and one that is based on transition matrices (Chapter 3.4.1). They are comparable to applied measures in other works [243].

One challenge of scanpath comparison is that viewing behavior individually changes with increasing scanpath duration. Hence, comparisons become more meaningful for shorter time spans. In videos, this could be for example a specific shot, or the time span an important AOI was visible. To support such variable time span analysis, the user is free to set the beginning and end of a time interval. Based on this selected time

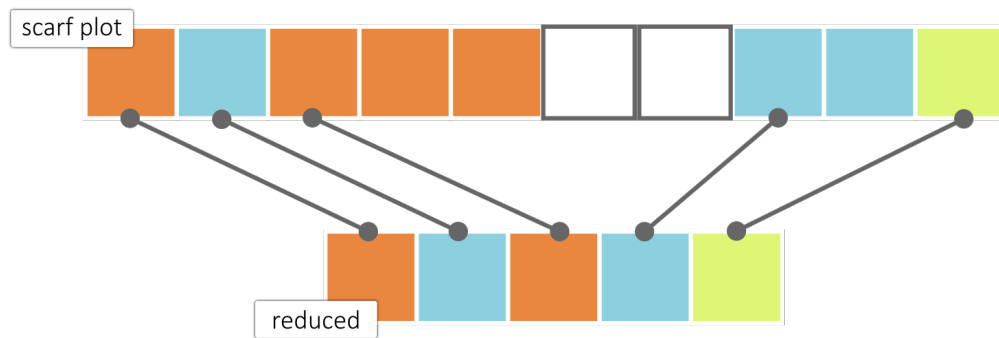


Figure 5.14: The original scanpath, represented by a scarf plot, is reduced by removing consecutive symbols of the same AOI. Blank regions mark time spans with no AOI-relevant data. The resulting sequence is used for the analysis of transition patterns.

span, a hierarchical clustering algorithm [140] calculates groups of similar scanpaths according to the chosen similarity measure. Hierarchical clustering has the advantage that it avoids the decision about the optimal number of clusters. The clustering is visualized as a dendrogram to the left of the scarf plots and provides an overview of all similarities. Hierarchical clustering starts out with a maximal number of clusters, with each scanpath forming its own cluster. The clusters are then merged consecutively, with the pair of most similar clusters being merged at each iteration of the algorithm. Average linkage is used to measure cluster similarity, i.e., the arithmetic mean of all pairs of instances in two different clusters. An example is discussed in Chapter 5.2.4.

To investigate changes in gaze behavior between AOIs, an overview of common transition patterns would be helpful. To achieve this, the AOI transition trees were developed.

5.2.3 AOI Transition Trees

The transition between AOIs marks an important step in the analysis of viewing behavior. It is often an indicator of attention shifts between different stimulus regions. To emphasize transitions, the earlier presented scarf plots can be abstracted even more by reducing the represented string to single AOI visits and the transitions between them. Figure 5.14 exemplifies this idea. While the scarf plot is based on data samples, e.g., one sample per frame, the reduced string is compressed to single dwells on AOIs where transitions become more obvious.

With such a reduced representation, the identification of common transition patterns is still cumbersome, especially when comparing sequences from many participants. Hence, the main goal of this visualization is to provide an overview of transition patterns of variable length with respect to the most common patterns but also including outliers. In the context of edited video content, the visualization emphasizes transition sequences within shots and the transitions between shots. An icicle plot helps achieve this goal.

Related Work

Considering the hierarchical structure of sequential visits on AOIs, generic visualization approaches for this type of data could be applied. An overview of such techniques is provided by Graham and Kennedy [127], Herman et al. [141], and Schulz et al. [260]. According to the design choices of this approach related work focuses on publications applying icicle plots to hierarchical/sequential data. Wongsuphasawat et al. [309] present an interactive icicle plot to display event sequences in hospital departments and millions of user action sequences from twitter. Trümper et al. [292] apply an icicle plot visualization similar to an AOI transition tree for the visualization of execution traces in software development. Telea and Auber [283] use a cushioned icicle plot to visualize the evolution of source code. Two linked icicle plots can be applied to compare hierarchical structures such as folders in file systems [150]. A sunburst visualization [272] uses a circular layout of icicle plots and can be used to visualize hierarchical data in general. However, all of these approaches do not include multiple linked trees and are not designed to fit the changing information of AOIs in a video.

Tsang et al. [294] visualize fixation sequences with a Word Tree [302], using AOI text labels for sequences with a maximal length of 5 for dynamic stimuli. The presented approach shares the same principal idea: sequences are represented by trees; branching into different AOIs along the timeline of the sequence corresponds to branching in the tree. However, there are several important differences as well. First, other than Tsang et al. [294], transitions between AOIs are visualized and not fixation sequences, to achieve a higher degree of data summarization. Second, the Word Tree is replaced by an extended version of a space-filling icicle plot [179] that allows the integration of thumbnails for an intuitive mental linking between visualization and stimulus. With the icicles, quantitative assessment of transition frequencies is better supported than by the text font size in the Word Tree. Third, an overview representation of multiple transition trees is introduced based on shot boundaries, leading to better scalability with stimulus length and number of AOIs in the full stimulus. Additionally, Tsang et al. [294] and West et al. [307] focus on the analysis of scenarios that contain a static set of few AOIs. With the presented visualization approach, changing AOI constellations are handled with thumbnails and transition sequences of arbitrary length can be displayed.

Visualization Requirements

Depending on the analysis task, stimulus, and recorded eye-tracking data, different requirements need to be met for an appropriate visualization of the data. In this case, the analysis task is to identify common transition patterns in gaze data from multiple participants watching video. Such patterns can reveal potential solution strategies for a given task, for example, how people examine a metro map to find the way from a start to a target location [30]. The following requirements and characteristics are relevant for the visualization and analysis:

Analysis of transition sequences and transition frequencies The visualization needs to display AOI transition sequences, not just transitions between pairs of AOIs. The visual salience of important transitions and their frequencies should become accessible by the visualization.

Subsequences of linear, ordinal scale time The temporal aspect of the data in the case of transition analysis focuses on the ordinal time scale of visited AOIs, arranged along linear time [33]. Furthermore, identical patterns of linear transition subsequences in the data should become visible, regardless of their exact temporal position; i.e., subsequences of patterns should be identified anywhere along the timeline.

Temporal division of the stimulus In contrast to unedited videos (e.g., from head-mounted eye tracking), edited material often contains intentional cuts that divide a video into scenes and shots that lead to abruptly changing AOI constellations over time. With these shot boundaries, a divide-and-conquer approach that splits the recorded data into semantic coherent sections can be applied. The advantage is that by dividing the data, consecutive transition sequences become shorter and therefore easier to interpret. In general, even unedited material can often be divided into parts of semantic coherence, e.g., by different events that happen.

Scalability Scalability with respect to the length of a video is not a critical aspect: video shots can be seen as individual units, and from the vast number of AOIs in a video, only those that exist in the current and the directly adjacent shots are important for the visualization of the transition tree. Scalability, in this case, concerns the number of recorded participants that have to be compared and the length of the transition sequences. Visualization techniques that display participants individually (e.g., scarf plots) tend to become harder to interpret with an increasing number of participants. Therefore, the visualization needs an aggregated representation of the participants, independent from their number. Although the frequency of transition patterns decreases with increasing sequential length, the visualization should show transition patterns of variable length, until the patterns become unique.

Semantic interpretation of AOIs Video stimuli can contain a vast number of AOIs that appear during different time spans in the video. Color mapping of AOIs is a common approach to make AOIs distinguishable (e.g., [66, 294]). For a semantic interpretation of an AOI, additional labels are necessary. A visualization with text labels can be the best choice if only a few AOIs exist and unambiguous labels can be given. In the case of edited video stimuli, however, a large number of AOIs can appear, making it tedious to find an appropriate label name for every AOI.

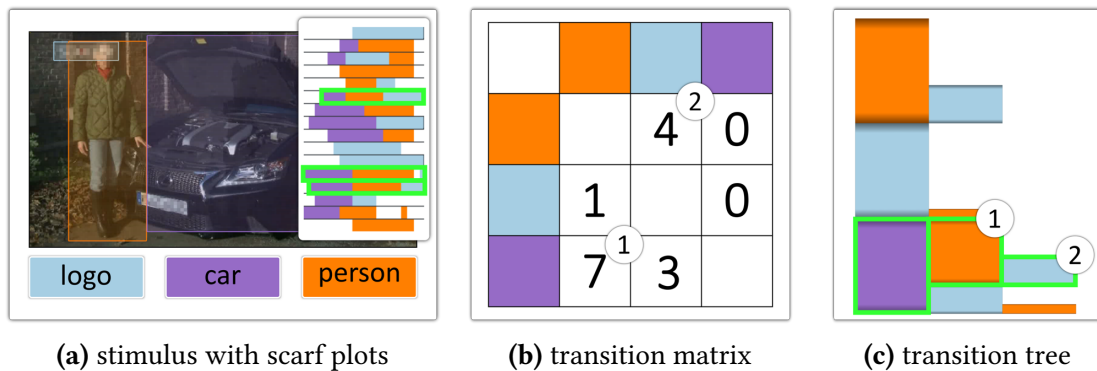


Figure 5.15: Creation of a transition tree. (a) In the example, three AOIs are visible: a logo, a car, and a person. The scarf plots show the scanpaths of 16 participants. (b) Pairwise transitions from one AOI to another can be visualized with a transition matrix. (c) The resulting transition tree shows the same information as the matrix on the first level (1), but can be further extended to depict how the sequence continued (2). The transition tree can then be used to highlight sequences in the scarf plots (green).

To meet these requirements, an enhanced icicle plot visualization—the *AOI transition tree*—was developed that displays the hierarchical structure of transition sequences. The area of individual nodes can be utilized for the color coding and labeling with thumbnails. The scalability of this visualization approach is improved by shot-based division of the stimulus and aggregation of sequence frequencies between participants.

Visualizing Transition Sequences by Extended Icicle Plots

The visualization is based on the reduced strings of all included scanpaths. Adopting Tsang et al. [294], the reduced strings are interpreted as a tree. In contrast to their approach, all subsequences are placed into the tree representation: regardless of when the subsequence occurs in the string, it is placed, beginning at the root of the tree. In this way, the requirement for subsequence analysis is met. In detail, transition sequences are represented by a multi-rooted tree, single nodes represent AOIs in a transition subsequence, the levels of the tree correspond to the length of a subsequence. In addition, nodes are assigned a numerical attribute that represents the frequency of visits to the corresponding AOI. With this interpretation of the scanpath data, the task is designing a visualization for a tree of varying depth and with one numerical attribute; the attribute has the property that the sum of the children's attributes is equal or less than the value of the attribute of the node itself because there cannot be more visits to subsequent AOIs than to the current AOI of a sequence. With this abstraction in mind, there are many potential visualization techniques (see Ward et al. [301] for a recent textbook presentation). The icicle plot [179] was chosen from this list of candidate techniques because it best meets the requirements.

Figure 5.15c shows an example of a transition tree derived from an example (Figure 5.15a). The transition sequences are represented by an icicle plot with horizontal orientation, i.e., the time axis is along the standard left-to-right reading direction in English. Single nodes of AOIs are displayed by rectangular boxes in the icicle plot. The height of the box indicates the frequency of AOI visits. Data from several participants is easily aggregated by adding up transition frequencies for the respective icicle boxes. The boxes are sorted according to descending height. Finally, the boxes need to be visually associated with respective AOIs. The color coding approach from *ISeeCube* is applied to color a box in the icicle plot.

The interpretation of the transition tree in Figure 5.15c can be explained by traversing the icicle plot from left to right:

- The *first level* of the tree (leftmost column) shows the dwell distribution to AOIs, aggregated from the full sequence. In the example, 10 participants started by looking at the car, the logo was visited 10 times, and the person was looked at 11 times. This level can be interpreted as a vertically stacked histogram.
- On the *second level*, transitions between two AOIs are displayed. The second level of a transition tree shows the same information as a transition matrix (see Figure 5.15b), representing the frequency of transitions between two AOIs. In the transition tree, the frequency can also be read off from the height of the box in the second level.
- Starting with the *third level*, the advantage of the transition tree representation becomes clear: sequences are interpreted identical to the second level, by traversing the transition tree from left to right. Other than with the transition matrix, sequences of arbitrary length can be identified efficiently. Since all appearing subsequences are displayed, patterns are possible to appear in other branches of the tree, showing which AOIs were visited before the sequence started.

Sequence of AOI Transition Trees

So far, the transition trees are applied to a defined time span in a video. If the time span includes several shots, the first level of the transition tree will include more AOIs that appeared in the video and individual scanpaths become visible with increasing length of transition subsequences. The temporal division of the stimulus allows for an approach that creates a sequence of smaller transition trees, instead of just a single, very large tree for the complete video. Figure 5.16 shows an example of a sequence of transition trees. For an individual AOI transition tree, absolute time is not considered. Still, additional information about the duration of a shot is important to find out if long transition sequences in a tree result from a long shot, or from diverging viewing

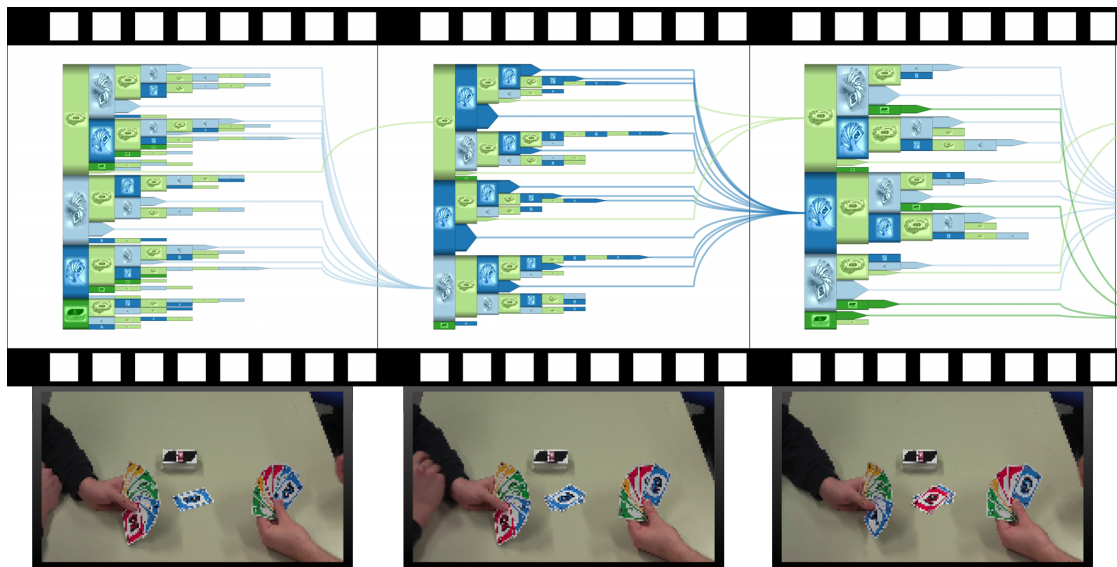


Figure 5.16: Shot sequence with three AOI transition trees depicted by a film strip metaphor. AOIs that continue a sequence in the following shot are connected by lines.

behavior of the participants. Therefore, a film strip metaphor is applied to facilitate a qualitative assessment of the length of a shot. Film strips that represent the video shots are concatenated horizontally, forming a horizontal timeline summarization of the complete video stimulus. Logarithmic scaling is applied to ensure that transition trees fit even in short shots. The transition trees are then positioned on the film strip in the corresponding shot.

To connect the AOIs of two consecutive trees, all transition sequences are extended by an additional level of AOIs after the end of a sequence. These additional AOIs are the next elements in the transition subsequences that continue in the consecutive shot. Here, an arrow shape is applied to emphasize the transition to the next shot. If a sequence is shortened due to filtering, no additional AOI is added since the sequence is not continued in the next shot. Finally, lines connect corresponding AOI boxes.

AOI Thumbnails

To this point, the transition trees consist of boxes with individual colors that represent the AOIs but the semantic interpretation of the AOIs has to be facilitated. Labels are a good way of building the link between the icicle box and the corresponding AOI. Text labels provide information about an AOI, but assigning meaningful labels becomes tedious with an increasing number of AOIs and often depends on subjective interpretations. Furthermore, text labels require relatively wide (horizontal) space.

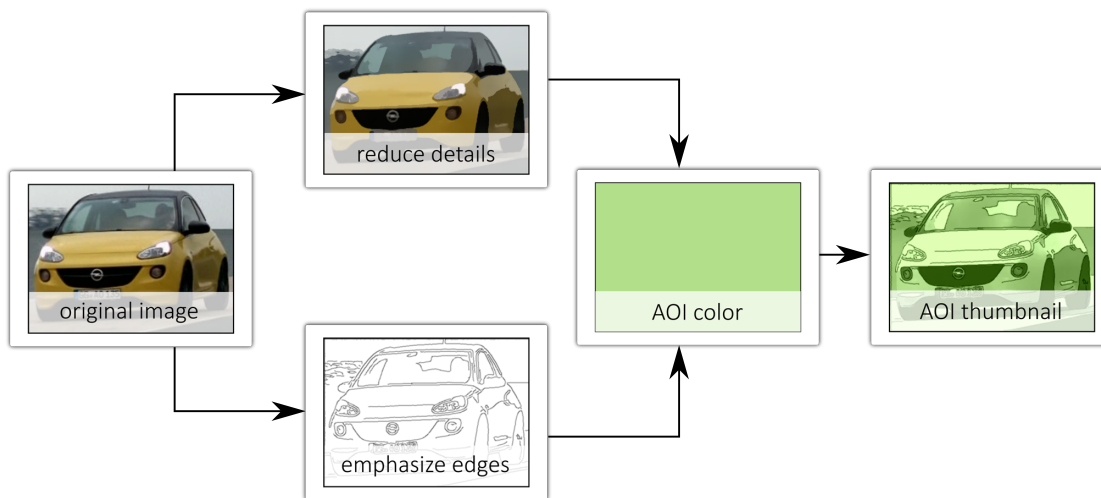


Figure 5.17: AOI thumbnail creation. The image is processed by multiple filtering steps, resulting in an abstracted representation of the AOI.

As an alternative, pictorial labels can be applied. To this end, AOI thumbnails were used: small images that show the object of the AOI in an abstracted representation, and that preserve the color assigned to the icicle box of the AOI. The AOI thumbnail is placed inside the icicle box to illustrate the AOI's object. A schematic representation with enhanced feature lines, adjusted lightness contrast, less image detail, and color modification is created for each AOI. The color is changed so that it matches the hue of the icicle box to maintain the color patterns of the AOI transition tree. This abstracted, non-photorealistic representation was chosen because it can be made readable even when shown as small picture. With this approach, labels are less dependent on subjective annotations of the labeling person, and interpretations of the AOIs that are involved in a time span become simpler, even without knowing the stimulus.

Figure 5.17 illustrates the image processing steps required to create an AOI thumbnail. First, a valid frame from the life span of an AOI is chosen. By mean shift filtering, image details are reduced and compositing with the AOI color can be performed on areas with a consistent lightness. Important edges are emphasized in the resulting image to provide the analyst with enough structural information of an object for its recognition. To this end, Canny edge detection is performed on the gray-scale version of the original image. For final compositing, the resulting images from mean shift filtering and edge detection are combined, taking into account the color of the AOI. Image compositing is performed in the perceptually linear CIE $L^*a^*b^*$ color space. The final image is obtained by compositing the lightness of the images and the AOI color. The thumbnail is finally inserted into the corresponding boxes of an AOI in the transition tree.

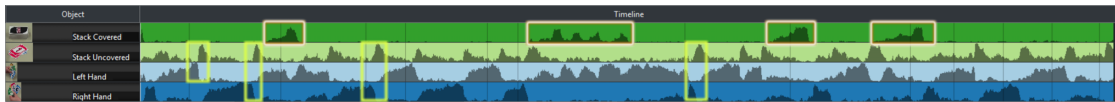


Figure 5.18: AOI timelines for the *UNO* dataset. High peaks on the left or the right hand are often followed by a peak on the uncovered stack (yellow). The covered stack is investigated only four times in the video (white).

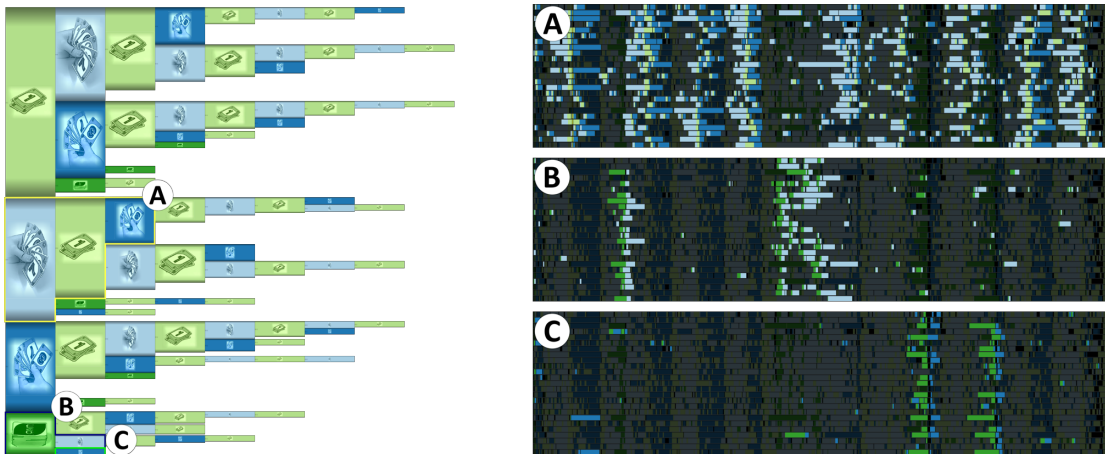


Figure 5.19: The AOI transition tree shows frequent sequences in the data. (A) One common pattern is from one player over the uncovered stack to the other player. (B)(C) Selecting the transition from the covered stack to one of the players reveals when they had to draw a card.

5.2.4 Example: UNO Card Game

All of the presented techniques were integrated into *ISeeCube*, providing a system of linked views for visual analytics on eye-tracking data. By the combination of different, interactive visualizations and automatic processing, a wide range of analysis tasks can be covered. This chapter aims to provide an example of how the combination of the presented techniques helps interpret the data.

For this example, the *UNO* video from the benchmark dataset (Chapter 4.4) is examined more closely. Figure 5.18 depicts the AOI timelines of the four AOIs in this video. Looking at the gaze distribution over time, it is clearly visible when the majority of the participants looked at the left and the right players. Attentional synchrony on one of the players is typically followed by a look at the uncovered stack and then on the opposite player. This represents the main pattern in this dataset: one player puts a card on the uncovered stack, the participants follow this card and then look at the other player to consider the next move. Investigating the timeline for the covered stack, only four time spans can be found that seem to draw attention. All of them are related to the event when one of the players has to draw new cards.

A general overview of these patterns is provided by the AOI transition tree (Figure 5.19). Selecting one of the aforementioned two patterns, e.g., *left hand* → *uncovered stack* → *right hand* (A), shows that the selected transition sequence appears multiple times in the scarf plots, as expected from the AOI timelines. As mentioned, the four events when all participants looked at the covered stack marks the time when one of the players had to draw a new card. Using the AOI transition tree, it is easy to determine which player had to draw. (B) Selecting all sequences from *covered stack* → *left hand* shows a clear majority of this sequence at the first two draw events. The second event in the middle shows this pattern multiple times since the left player had to draw four new cards at once. This is also indicated by the four peaks in the second event (Figure 5.18). (C) Selecting the sequence *covered stack* → *right hand* shows that the last two draw events are performed by the right player.

For a detailed look at a draw event and how it is attended by the participants, Figure 5.20 depicts the first of the four events. The corresponding time span is selected and scanpath clustering is performed. From the time span shown in the AOI timelines and the scarf plots, three example frames were extracted to explain what happened in the video:

- ① The player on the right places the red card with the number 1 on the uncovered stack. The player on the left has no valid card to play and is forced to draw a new card from the covered stack. Participants watching this video have different reaction times to realize that a new card will be drawn.
- ② The participants 8, 3, and 4 are the fastest to look at the covered stack in anticipation that the next action will take place there. This indicates that they followed the game attentively and anticipated the next move correctly. However, the gaze of participant 4 moved back to the hand of the right player, leaving the possibility that the look at the covered stack was unconscious.
- ③ The player on the left draws the new card from the covered stack. When the hand moves towards the covered stack, the majority of the participants move their gaze on this AOI. Such behavior can be just a reaction to the motion in the video, indicating that some of the participants did not follow the game attentively, either because they were bored, or did not fully understand the rules of the game.

The dendrogram in Figure 5.20 shows the clustering based on the Levenshtein distance. The first big cluster (Viewer 10 – Viewer 23) consists of participants who mainly looked at the covered stack after the hand of the left player was moving towards it. Within this cluster, many participants were looking at the uncovered stack when the red 1 was played. After that, their gaze moved back to the right player, although this was not necessary to anticipate the next move. In this example, the right player could also have played a red 7, which could be the reason for some of the participants to look back.



Figure 5.20: Scanpath comparison based on Levenshtein distance and clustering incorporated in the scarf plots by a dendrogram on the side. The participants in the lower cluster (Viewer 8–6) recognized earlier that the left player has to draw a card from the covered stack.

Viewer 1, 17, 22, and 23 moved their gaze immediately to the cards on the left, staying there until the new card was drawn. The second cluster contains all participants that looked at the uncovered stack before the hand was reaching for the card. Viewer 4 and 6 show some kind of outlier behavior. As mentioned before, Viewer 4 looks early at the covered stack and switches between the right hand and the stack. Viewer 6 shows more inconsistent viewing behavior, constantly switching between all AOIs.

5.2.5 Discussion

The AOI-based methods, i.e., the *AOI timelines*, *scarf plots*, and *AOI transition trees* extend the range of solvable analysis tasks. Each visualization by itself is valuable, but their combination helps investigate data more efficiently. Participant-related questions and comparisons can be investigated with the first two visualizations while the transition trees provide an overview for relations between AOIs in form of transition patterns.

The example shows how AOI-based visual analytics can be applied with a drill-down strategy. First, the overview by the AOI timelines and transition trees is investigated. After identifying important events, i.e., drawing of cards, the user can investigate individual events in detail in the scarf plots and the video. Vice versa, it is also possible to explore the data for events and then look at the overview to find similar patterns. To further improve this aspect, searching for similar patterns could be improved by automatically calculated suggestions. Future work could incorporate a query interface similar to the one presented for movie analysis (Chapter 2.3) to achieve this.

The presented techniques require AOIs that have to be tediously annotated first. Hence, this thesis further contributes some work on alternative, image-based approaches for gaze visualization. By incorporating stimulus content in a point-based visualization, the definition of AOIs becomes easier, or even unnecessary for some scenarios.

5.3 Image-Based Eye-Tracking Visualization

To this point, gaze data was interpreted based on spatio-temporal patterns or visited AOIs. The investigated visual stimulus was often only visible by representative screenshots or video playback. The idea of image-based eye-tracking visualization is that the visual content of a stimulus is represented directly in the context of the gaze data. For video, this provides faster interpretations of what happened and image processing can be incorporated in visual analytics approaches. As with the other presented techniques, the main goal of the visualization is to provide an abstract overview of the dynamic data that reduces video skimming and supports an efficient interpretation of gaze behavior. To achieve this goal, this chapter discusses two approaches that cut out image content from the respective video at gaze positions:

- **Gaze stripes:** This approach takes the current gaze position of a participant and creates a thumbnail of the foveated video content. The images are placed on a horizontal timeline, representing a scanpath as a sequence of thumbnails. Multiple participants' sequences are stacked vertically for comparison. Different annotations on the *gaze stripes* [24] provide means to create a visual protocol for communicating study results. With *fixation-image charts* [23], this concept is further extended by a glyph-based representation and visual analytics for annotation purposes.
- **Gaze-guided slit-scans:** Slit-scans are static representations of video content created by placing vertical slices from the video next to each other over time. By adjusting the position of a slice to the current gaze position, an individual slit-scan is created, representing a visual fingerprint of a scanpath [18]. Image-based metrics are applied to perform scanpath comparison without AOIs [10].

Both techniques provide effective analysis methods without AOIs. By incorporating the stimulus content in the visualization, synchronized participant data becomes easy to compare in the search for commonalities and outliers.

5.3.1 Gaze Stripes

Gaze stripes are a visualization technique for passive stimulus content. This type of data has the advantage that it can be synchronized between participants and patterns over time become visible in the visualization.

Related Work

The presented technique displays the gaze data from multiple participants by stacking individual timelines on top of each other. This approach is visually similar to the work by Andrienko et al. [40] who visualize the distances to selected points of interest with color coding, similar to the scarf plots discussed earlier. Although visually similar, their technique depends on annotated eye-tracking data. The main advantage of the presented approach is that the definition of AOIs is not required.

Fixation data can be mapped individually to AOI labels without actually defining boundary shapes. A semi-automatic approach for such annotation can be found in *SemantiCode* [236], which uses image thumbnails based on fixation positions from a video stimulus to let the user define to which AOI they belong. This information is then applied to create a classification scheme for the remaining fixations in the data. Ishiguro and Rekimoto [162] also extract gaze data this way to represent video life-log recordings. These approaches are similar to the presented in terms of interpreting gaze data by image thumbnails. However, *SemantiCode* applies this principle for annotation only and further analysis by statistical or visual techniques is still required to interpret the data. Also, in dynamic stimuli, the changing conditions require the analyst to perform manual annotations. With gaze stripes, the gaze data can be interpreted directly by the analyst, while automatic processing of the image data can be applied on demand to further support the interpretation of selected time spans.

Manovich [202] discusses thumbnail-based visualizations to summarize video data. The author describes an approach to stack key frames of video gameplay of a user, as well as of multiple video sequences (see also Takahashi et al. [279] and Christel and Martin [85]). In these publications, only complete frames are visualized to summarize the content of a video without the spatio-temporal eye-gaze information of multiple participants. With the work in this thesis, this principle is extended by a gaze data-driven selection of the sub-scene context to create thumbnails for an arbitrary number of participants that can then be compared with each other. By applying a sequence of thumbnails with adjustable crop size, occlusion-free stripes display the participants' eye gazes with stimulus information.

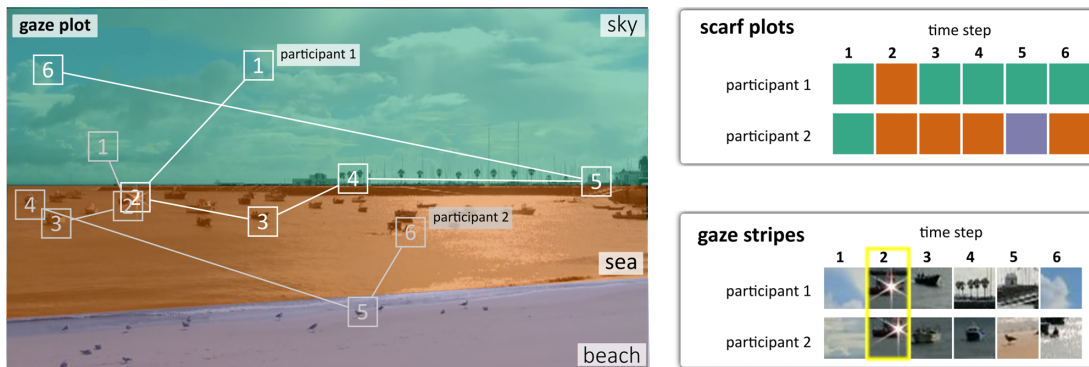


Figure 5.21: Gaze stripes in comparison with scarf plots. With stimulus information in the gaze stripes, it is easy to identify why participants investigated a region simultaneously (yellow).

Gaze Stripes Visualization

The visualization of data from multiple participants with gaze stripes is based on two data sources: the visual stimulus that was investigated and the spatio-temporal point-based gaze information that was recorded by an eye tracker. This means that a coupled data analysis problem has to be solved and the resulting visualization should help answer questions about time, space, context, and individual participants.

Figure 5.21 describes in detail how gaze stripes are created. The example shows a scene divided into three parts: the beach, the sea, and the sky. Two exemplary scanpaths are shown with a gaze plot. For each gaze point, a thumbnail image with the local context of the stimulus is cut out and stacked along the timeline. Invalid sample points (e.g., due to missing eye detections) are not drawn, leaving an empty field for this time step to keep the data synchronized. This approach maps participants and time in a similar way as scarf plots, i.e., time along the x-axis and participants along the y-axis. In comparison, the amount of details visible in the scarf plots strongly depends on the defined AOIs. On a coarse scale with 3 AOIs as in Figure 5.21, the scarf plots provide only the information that the participants first looked at the sky (time step 1), then at the sea (time step 2). In the gaze stripes, one can directly see that both participants did not just look at the sea, but at the same boat in time step 2. To acquire this information with scarf plots, either a definition of more AOIs or additional visualizations are required.

Raw gaze data coordinates, as well as filtered data (i.e., fixations), can be analyzed with gaze stripes. For fixation data, microsaccadic eye motions are filtered and identical thumbnails tend to recur more often. However, it could be more difficult to detect certain viewing behavior with filtered data, for example, smooth pursuits when the participant follows objects. If fixation filtering is applied, the thumbnails can be summarized into one image per fixation, but this would impair the comparability of synchronized time steps. This idea was reconsidered for the *fixation-image charts* (Chapter 5.3.2). Since

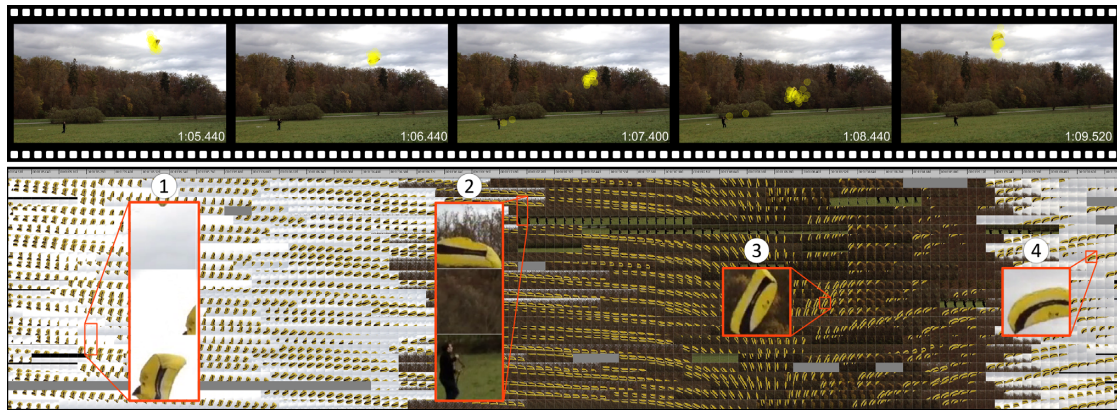


Figure 5.22: Gaze stripes of the *Kite* video from the benchmark dataset.

the investigated video stimuli contain sequences of smooth pursuit eye movements that are not fully covered by current fixation algorithms, the visualization based on raw gaze data is preferred.

Figure 5.22 displays an example of the technique applied to the *Kite* video (Chapter 4.4). The gaze stripes for the participants are displayed in a stacked manner. On the overview level, patterns and outliers can be detected and a general impression of the scene is provided. For instance, all participants followed the motion of the yellow kite. Zooming-in allows a more detailed analysis of the eye-tracking data, demonstrated with the close-up images: ① different participants focused on different parts of the kite; ② while most of the participants focused on the kite, one participant looked at the person controlling the kite; ③④ the changing orientation of the kite is clearly visible.

For a first impression of the data, this visualization is already helpful. To further communicate findings with the visualization, different complementary views on the data are available that can be used by the analyst to annotate important time spans. The approach allows a detailed analysis of the data without the need to define AOIs or apply complex algorithms. The visualization is occlusion-free and easy to understand, even for non-experts.

Complementary Views

Besides zooming and panning the gaze stripes, the analyst can select time steps and time spans of single or multiple participants' data by simple mouse dragging. A zoom lens (Figure 5.23 ③) can be activated on demand to enlarge the local neighborhood of the currently hovered thumbnail. For every selection, a new view can be created, showing specific aspects of the data. These views are attached to the gaze stripes as annotation items. The global context is displayed in a video player with a bee swarm.

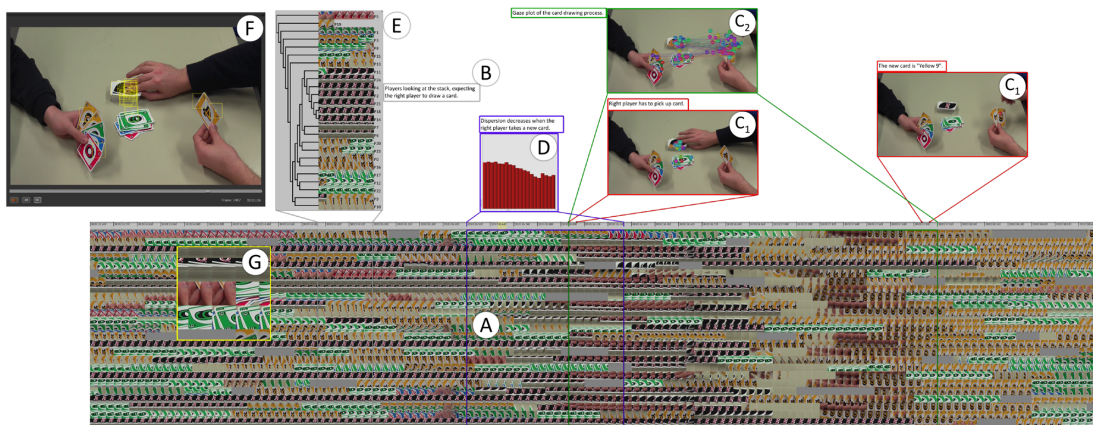


Figure 5.23: Gaze stripes can be enriched by several complementary views that provide the global context of the stimulus. Time spans can be annotated with (A) colored area markers, (B) annotation notes, (C) gaze plots, (D) dispersion histograms, and (E) a hierarchical clustering of the participants. (F) A linked video player plays back the stimulus with gaze information. (G) An interactive zoom lens facilitates a detailed analysis without leaving the overview.

The different annotation items are *area marker*, *note*, *gaze plot*, *dispersion histogram*, and *hierarchical clustering*. Each item is freely scalable, movable, and individual colors can be applied to support visual grouping of items. All items that refer to single time steps or time spans set markers on the timeline in their corresponding color.

Figure 5.23 displays the *UNO* dataset in a time span that comprises the event when the right player has to pick up a new card from the uncovered stack of cards. The figure shows a screenshot created completely with the implementation (except for the enumeration symbols (A)–(G)). All applied items describe the viewing behavior of the participants during this event:

- (A) **Area marker:** Selected regions of the gaze stripes can be highlighted by a colored frame around the involved thumbnails. Individual participants or groups of participants can be marked for annotation with the other items.
- (B) **Annotation note:** Note items provide the analyst a free-text field to annotate events of special interest or comment on other items. This allows for a detailed description of the data to communicate the visualization.
- (C) **Gaze plot:** Depending on the analyst's selection, gaze plots can be created for single time steps or over longer time spans. If only one time step is selected, all participants' gaze positions included in the selection are rendered into the video frame (C₁). If a time span is selected, the last video frame is used to provide the scene context and the spatio-temporal development of the participants' scanpaths is depicted by the gaze plots (C₂).

- Ⓓ **Dispersion histogram:** The visual similarity of regions in the stimulus can lead to similar thumbnails in the gaze stripes. Although this is an advantage for some cases, other situations require the analyst to know if the participants were looking synchronously at a particular region, or if their gaze was distributed between similar looking objects. Therefore, the intersubject dispersion metric D_t [222] for N participants is included:

$$D_t = \frac{1}{N} \sum_{i=1}^N \frac{g_{t,max}^{i'} - g_t^{i'}(x_i, y_i)}{g_{t,max}^{i'} - g_{t,avg}^{i'}}$$

For frame dimensions $m \times n$ with $x \in \{1, \dots, m\}$ and $y \in \{1, \dots, n\}$, the gaze density function $g_t(x, y)$ is defined as:

$$g_t(x, y) = \frac{1}{N} \sum_{i=1}^N \varphi_i(x, y)$$

The gaze positions (x_i, y_i) from N participants at time t , are replaced by the Gaussian function:

$$\varphi_i(x, y) = e^{-\left(\frac{(x-x_i)^2}{2\sigma^2} + \frac{(y-y_i)^2}{2\sigma^2}\right)}$$

In the equations, $g_{t,max}$ and $g_{t,avg}$ denote the maximum and average of $g_t(x, y)$ with i' denoting that the i -th gaze position is excluded from $g_t^{i'}(x, y)$. The value $\sigma = 40$ pixels is chosen according to the visual angle at a distance of 64 cm, the standard setting for the recorded data. Higher values of D_t indicate that the participants' gaze data was distributed more widely over the scene, and lower values indicate time spans of attentional synchrony [268].

- Ⓔ **Hierarchical clustering:** Since similar viewing behavior can occur between arbitrary participants, a new ordering of the gaze stripes is required to obtain better visual coherence between neighboring stripes. Therefore, the selected time span can be duplicated and clustered based on the scanpath comparison with a modified Levenshtein distance for image histograms. The result of the hierarchical clustering is then displayed as an item attached to the timeline.
- Ⓕ **Video player:** The video player shows a gaze replay of the recorded eye movements. For video stimuli, the animated content is displayed, for static stimuli, only the gaze replay is presented on the image. For the gaze replay, the borders of the thumbnails for each participant are shown in the stimulus.
- Ⓖ **Zoom lens:** With the zoom lens, single thumbnails and their neighborhood can be investigated without leaving the overview.

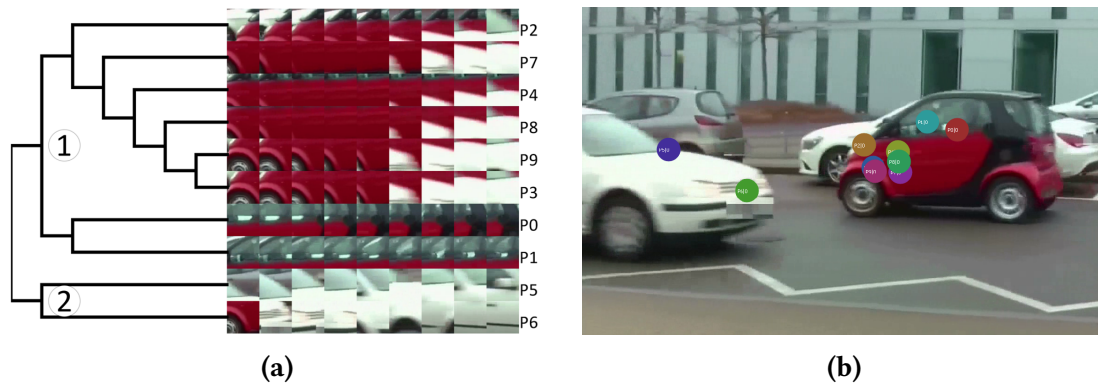


Figure 5.24: (a) Example clustering of a short sequence of gaze stripes from the *Car Pursuit* dataset. Two major clusters are visible: ① participants focusing on the moving red car, and ② participants shifting their gaze to the appearing white car. (b) Gaze plot from the video of the scene in which the white car suddenly appears from the left.

Scanpath Clustering

To allow analysts to identify structures of selected gaze sequences based on their similarity, hierarchical clustering similar to the approach described in Chapter 5.2.2 is included. Since no AOIs are used, the comparison is performed with a modified Levenshtein distance based on the image histograms. Such an image-based comparison has the advantage that it is not depending on a common coordinate system for all participants, as it is the case for trajectory-based comparisons. For thumbnail sequences, rather than counting the number of exchange operations, the costs of each of these operations are quantified by the distance between both thumbnails. The thumbnail distance is measured by the correlation of hue and saturation histograms of two images. Only the hue and saturation channels are used to reduce problems with shadows in the scene [315]. The Pearson correlation coefficient ρ_{H_1, H_2} is applied to measure the similarity between two histograms (Chapter 2.1). The modified implementation of Levenshtein's string distance measure uses $d = (1 - \rho_{H_1, H_2})$ as the distance for two images. This method for calculating image sequence distance is akin to the one presented by Tan et al. [281], which was developed to compare long video sequences. Histogram correlation works well as an image distance measure for the tested datasets, but using the Levenshtein algorithm as a sequence distance measure is flexible enough to accommodate any other image comparison method in case they are better suited for particular datasets.

The clustering item shows the result as a dendrogram allowing for an in-depth analysis of scanpath similarities. The selected sequences form the leaf nodes of the dendrogram, including the IDs of the respective participants. An example is depicted in Figure 5.24. The clustering shows a sequence from the *Car Pursuit* data in which the camera follows

a red car moving from right to left. Suddenly, a white car appears from the left, and some participants shift their gaze to it. A clustering of the gaze stripes from the point at which the white car appears is depicted in Figure 5.24a, while Figure 5.24b contains a gaze plot of this sequence. The clustering shows the different reactions to the sudden appearance of the white car. While the six participants in the top cluster kept their eyes on the red car (P2, P3, P4, P7, P8, P9), the two participants in the lower cluster immediately shifted their gaze to the white car (P5, P6). In the upper cluster, the white car only appears in the thumbnails when it starts occluding the red one. Two of the participants in the center cluster (P0, P1) can be considered as outliers, as they get merged late in the clustering process. This is due to the fact that both participants were keeping their eyes on the side window of the red car rather than on its body, which differs in terms of the color palette.

The gaze stripes provide an easy-to-interpret visualization for the first look on recorded data. With the annotation items, visual protocols can be created for dissemination purposes. Since the horizontal scalability depends on the number of samples, an approach that shows only fixations would drastically reduce the horizontal extent of the visualization. For this purpose, *fixation-image charts* were developed.

5.3.2 Fixation-Image Charts

The advantage and also the main issue of gaze stripes is the fact that they rely on the synchronous comparison of data samples. It is easy to spot similar thumbnails and outliers over time. However, for an average frame rate of 30 Hz, less than one second of recorded data can be displayed in original resolution on a regular screen without zooming out. For the gaze stripes, this issue is handled by uniform skipping of samples to reduce the number of thumbnails. If fixations are available, the according time span can be summarized with a single thumbnail. This idea is incorporated into a glyph-based visualization approach.

Visualization Components

Fixation-image charts consist of (1) representative fixation images to display the context of the underlying stimulus, (2) distance bars as an indicator for saccade lengths between consecutive fixations, and (3) time streams to maintain the temporal synchronization between participants. Figure 5.25a shows a single glyph and Figure 5.25b shows how the glyphs are displayed in sequence.

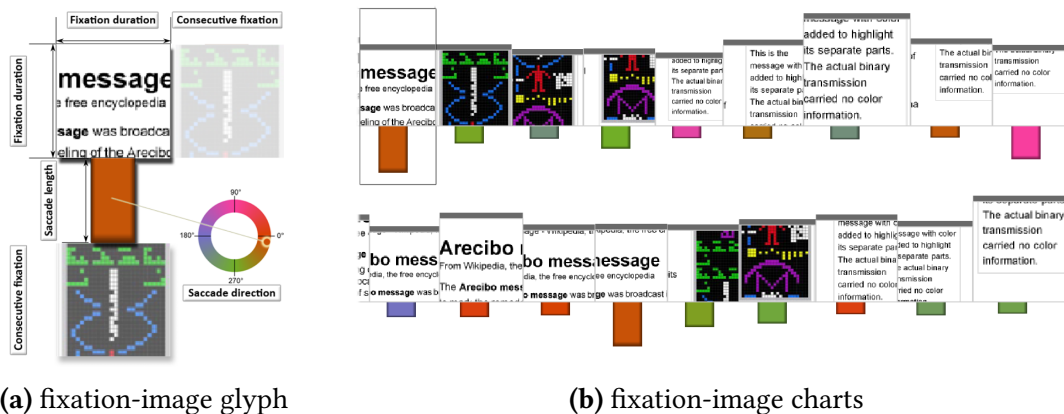


Figure 5.25: (a) Multiple fixation metrics can be encoded in one glyph. (b) The fixation-image charts represent the scanpaths of participants and can be compared with each other. (Stimulus: Wikipedia Arcibo Message¹)

Fixation Images

The fixation image is extracted as described for the gaze stripes. This image provides information where the participant was looking. To encode the duration of a fixation, the height and width of the image are adjusted accordingly (Figure 5.25a). The fixation duration is normalized by the longest fixation of all participants. This approach has the benefit that longer fixations also receive more space in the visualization. To emphasize the differences between low values and assure the linear growth of areas, the square root of the values is used. A scanpath is visualized by placing the corresponding images next to each other (Figure 5.25b). If single fixations are selected, a compact representation of the results is shown under the timelines. In this case, the consecutive fixation is placed under the distance bar, indicating the target position of a saccade.

Distance Bars

Below each fixation image, a distance bar indicates the distance between consecutive fixations (Figure 5.25a). Although not calculated explicitly by algorithm, this distance is often applied as an implicit indicator for saccade length. The height of the bar describes the Euclidean distance between two consecutive fixations, normalized by the maximum distance within all participants' fixations. This representation is similar to a time plot of this metric that is common in eye-tracking research and is easy to interpret by experts.

The distance bars are color-coded by saccade direction to show even more information about the spatial relation between two fixations. The angle between the horizontal and the connection line between two fixations is used to obtain the color from the

¹ http://en.wikipedia.org/wiki/Arcibo_message, last checked: October 13, 2018

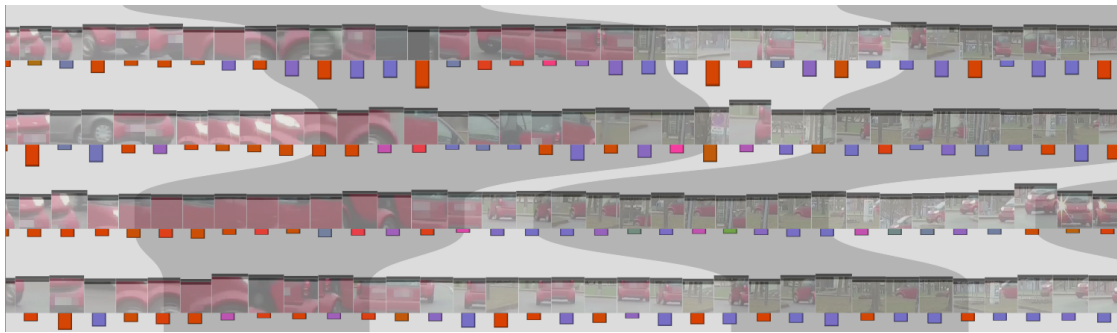


Figure 5.26: Time streams starting from fixed timeline intervals and comprising all fixation images that lie within the interval.

color-legend wheel (Figure 5.25a). By this approach, it is easy to detect bars of similar length and color, which can indicate repetitive behavior (e.g., reading from left to right). However, the direction-to-color encoding requires some practice for interpretation. Therefore, the color legend was integrated directly to a corresponding filter dial to help with the interpretation of directions.

Time Streams

Since the number and length of individual fixations vary for each participant, the depicted image sequences have different element counts and consequently varying width. For displaying such sequences on a timeline, there are two options: showing the images on an absolute time scale or showing the images stacked next to each other. The first approach preserves the synchronization between participants, but the resulting timeline would create gaps between fixations and the horizontal scalability would be impaired. Therefore, the second approach is applied, neglecting absolute temporal position of fixations. This approach creates a dense representation of all fixations in their sequential order, requiring less space on the horizontal axis. However, it results in asynchronous timelines due to the varying number of fixations, impairing an efficient comparison between participants.

To compensate the asynchronicity between participants, time streams for equidistant time intervals (see Figure 5.26) were included. For each time stream interval, the stream passes the fixation images with the respective time stamps. The resulting segments are finally combined to a set of Bézier curves that mark the corresponding time spans for all participants. The time streams are rendered in the background of the fixation images with an alternating color scheme that can be adjusted individually. With these time streams, the asynchronicity can be compensated and the timelines remain comparable. The user can adjust the selected time span interval.



Figure 5.27: Visualization overview: (A) Fixation-image charts, (B) filter query interface, (C) query results, (D) label editor, (E) stimulus view. (Stimulus: UAD Infographic²)

Interaction

Like the gaze stripes, the fixation-image chart displays an overview of consecutive fixations without overlap. It would be possible to incorporate annotation items here as well. However, under the aspect of visual analytics, the focus of this work is to improve the visual search for important events in eye-tracking data with additional processing and interaction techniques. Therefore, a stimulus view that displays the data with established scanpath visualizations, filter options to fade out fixation images unimportant for the current analysis, and a labeling function for selected fixations were implemented (Figure 5.27). The fixation-image charts (A) and currently selected elements (C) are in the center view.

Stimulus View

The additional view shows the complete stimulus (Figure 5.27 (E)). A gaze replay of selected participants is displayed during the playback of the recorded data. The linking between the stimulus view and the visualization is bidirectional. Changing the time in the stimulus view selects the respective fixation images in the visualization. Selecting a fixation image in the visualization sets the stimulus view to the according time stamp. The video player shows a gaze plot of selected time spans. As it is common practice, the fixation duration is encoded by the radius of individual fixations. Additionally, labels are rendered into the visualization during playback, allowing for the verification of the labeling process. By selecting consecutive fixations of a participant, the respective part of the scanpath is also highlighted in the stimulus view.

² <https://www.stopalcoholabuse.gov/resources/Infographics/share.aspx?info=6>, last checked: October 13, 2018

Filtering

From the experience with the gaze stripes, providing just an overview of the data is not sufficient for some tasks. Automatic highlighting and selection based on different properties can support the visual analysis. With such support, it becomes easy to investigate fixations and label them. Therefore, an interactive query interface (Figure 5.27 (B)) allows one to filter the data, according to the following categories:

- **Filter by fixation data:** The properties *fixation duration*, *fixation distance*, and *saccade direction* can be applied as filter criteria for selecting fixation glyphs. Hence, knowledge about certain eye movements can be applied to highlight corresponding fixations.
- **Filter by image similarity:** The analyst can select a reference image from the visualization and retrieve similar images. To this end, two similarity measures are included. The first is based on the detection and matching of SIFT features [197] between the reference and the fixation images. To normalize the similarity measure, the number of matching features between the reference image with itself is used. The second similarity measure is based on a histogram comparison with the Bhattacharyya distance [51] between the images. These two measures are chosen because of their applicability to different analysis tasks. Regions with a similar structure can be identified with SIFT features, whereas regions with similar color can be identified with histograms.

All filters can be enabled and adjusted individually by separate dials (Figure 5.27 (B)). Enabled filters are concatenated by a logical AND connection. Fixation images outside of the selected ranges will be faded out in the visualization, highlighting only the currently relevant images.

Labeling

As support for dissemination and automatic processing (e.g., classifier training) the analyst can specify labels that can be assigned to fixations (Figure 5.27 (D)). In contrast to intersection tests with AOIs, labels can describe more than just regions or objects, e.g., task-specific events. Labels are visible in the visualization through their assigned color and in the stimulus view through their name.

For example, Figure 5.28 shows the fixation-image charts for the *Memory* dataset (Chapter 4.4). Since the participants should watch the game attentively and anticipate where the matching card for each turn is, one can assume that they fixate a covered card longer when they think it matches to the current one. Hence, the filter for fixation duration is activated to show only fixations longer than 3.5 seconds. With this threshold,

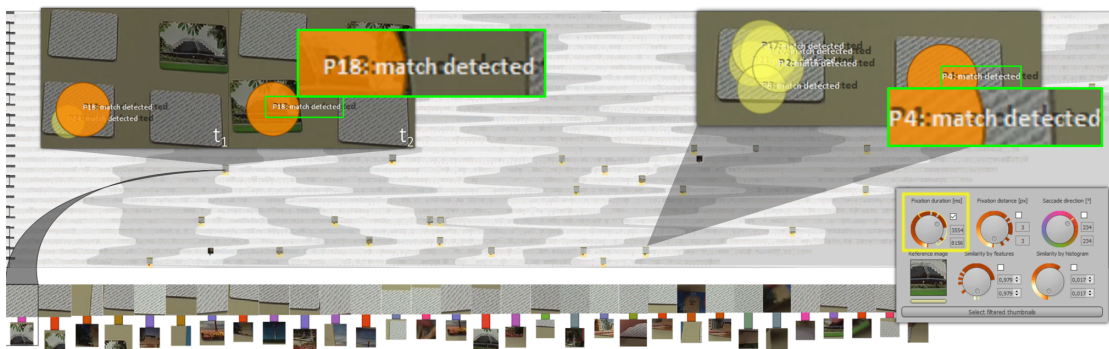


Figure 5.28: Labeling example for the *Memory* dataset. Only fixations with the longest durations are highlighted by the selected filter. Except for two of them, all images (bottom) show an event where participants fixate a covered card because they assumed the matching card there. All fixations can be labeled at once and individually validated in the video.

the filtered data shows 29 fixation images that display the backside of the memory cards and two on uncovered cards. One can select all backside images and label them, here with the label *match detected*. From the large number of fixations in the dataset, this already reduces the search time for events when participants thought they identified the correct match. Investigating the labels individually helps identify which participants are correct (Figure 5.28, P18 left) and who fixated the wrong card (Figure 5.28, P4 right). That way, top-down analysis with existing knowledge about the data is supported.

Fixation-image charts can also be used to label static stimuli. The respective publication [23] contains an additional example that demonstrates how prior knowledge about reading behavior can be applied to label a dataset of participants looking at a website.

With the step from thumbnails of raw data (as performed with gaze stripes) to the extraction of images based on fixations, the required screen space for the visualization is significantly reduced. *Gaze-guided slit-scans* further reduce the required screen space down to one pixel per depicted time step.

5.3.3 Gaze-Guided Slit-Scans

One important aspect of image-based eye-tracking visualization is the question: *How much image content is necessary for an effective analysis of the data?* For gaze stripes and fixation-image charts, the choice of thumbnails is motivated by the foveated area on the screen. With slit-scans, the image content per time step is reduced to a scanline. In the following, it is analyzed how well the visualization is suited for scanpath comparison.

Related Work

The slit-scan technique is popular for artworks depicting video motion either in a static picture or in a new abstracted video sequence³. Early examples can be found in Stanley Kubrick's *2001: A Space Odyssey* in the star-gate sequence, and in the adaption of the technique for computer graphics [228].

In research, slit-scans are an effective method to summarize long video sequences for visual inspection. For example, Martinho and Chambel [203] and Schoeffmann et al. [258] apply slit-scans as timeline visualization for fast video browsing. For automatic video analysis, slit-scans are often referred to with the term *visual rhythm* [82, 173] and used for various purposes: Motiongrams [166] focus only on the motion in the video, reducing the visual complexity of the resulting visualization. Bezerra and Lima [49] extract descriptors for soccer analysis tasks and for shot detection [50] based on slit-scans from recorded videos. Based on these experiences from other research fields, there is much potential for slit-scan analysis for eye-tracking videos.

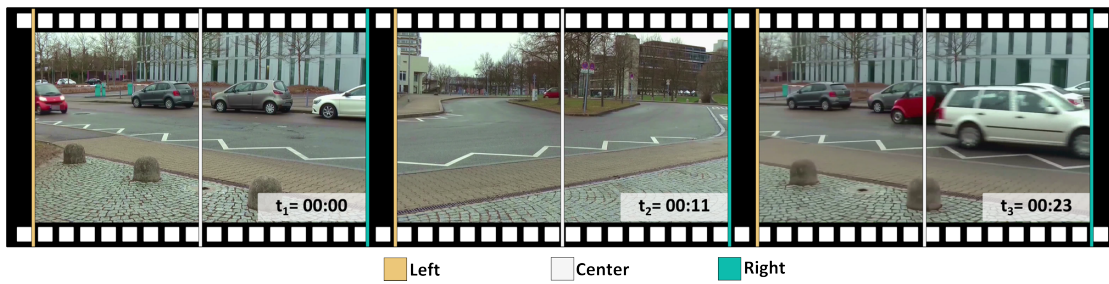
Tang et al. [282] extend the concept by allowing the user to define arbitrary scanline orientations to investigate specific regions in video sources. Based on the idea of flexible scanlines, this chapter introduces the concept of gaze-guided slit-scans [25]. By including the technique in a visual analytics approach [10], an alternative method for scanpath comparison is provided and shows that an image-based metric for slit-scans provides promising results that are more similar to measures from annotated data, than trajectory-based alternatives.

The presented approach is a new scanpath representation based on the slit-scan technique, creating an individual visual fingerprint for each participant's scanpath. The resulting images can be compared with established methods for image comparison and based on scanpath metrics. By integrating the slit-scans in a visual analytics system that supports different metrics, interactive support for multiple metric comparisons is achieved. The proposed image-based metrics for the slit-scan provide results that correlate stronger with AOI-based measures than trajectory-based metrics.

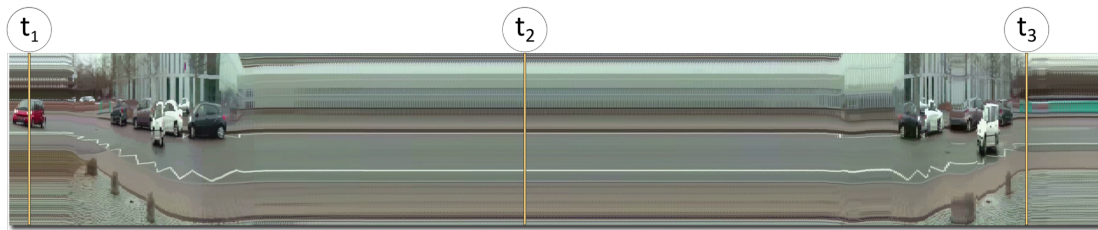
Visualization Components

Figure 5.29 shows how the slit-scan technique works. In the example, a vertical scanline is extracted from each frame of the video (a) and placed next to the previous time step, one from the left (b), the center (c), and the right part of the image (d). Each scanline results in a different image. The video contains two camera panning motions, the first from left to right and the second from right to left. This camera motion visible in all three images (b–d) independent from the position of the scanline. When the camera

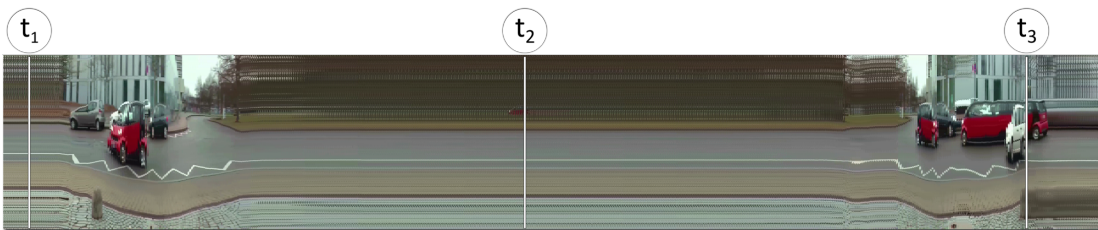
³ Levin, Golan. An Informal Catalogue of Slit-Scan Video Artworks, 2005-2015, http://www.flong.com/texts/lists/slit_scan, last checked: October 13, 2018



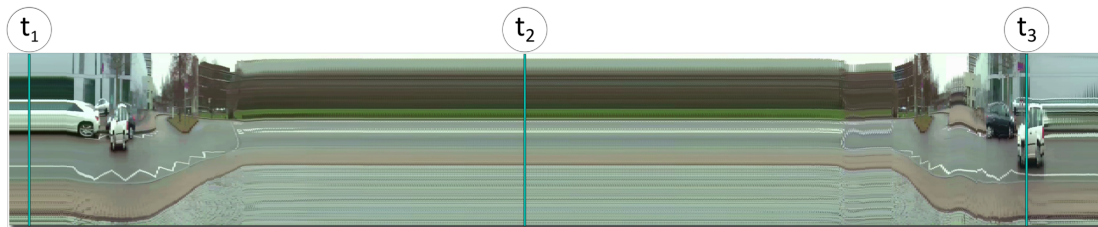
(a) Car pursuit video with three different scanlines (left, center, right).



(b) left



(c) center



(d) right



(e) gaze-guided slit-scan

Figure 5.29: The slit-scan technique demonstrated on the *car pursuit* video. Depending on the position of the scanline, the resulting image differs (b)–(d). Moving objects appear mirrored in the visualization. Adjusting the scanline to the gaze position results in a slit-scan (e) that contains information about all attended content.

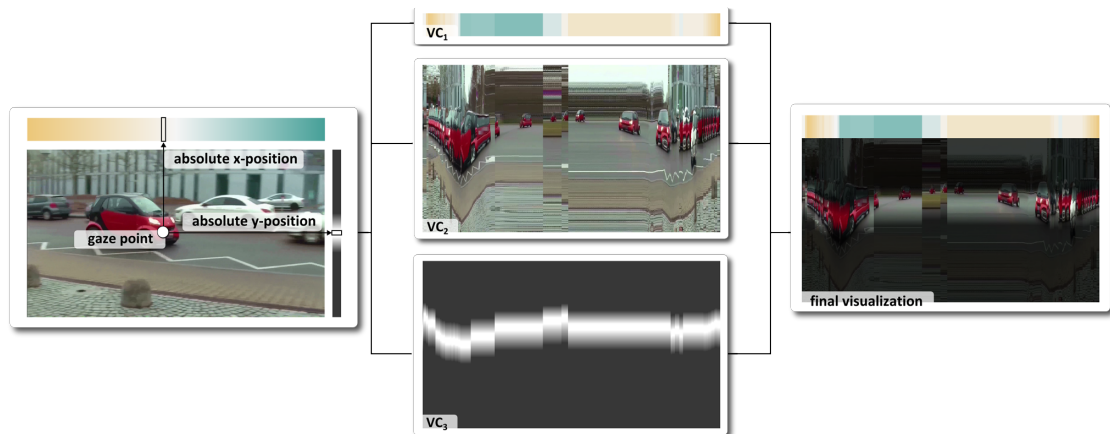


Figure 5.30: The slit-scan (VC_2) is enriched with information about the absolute gaze position. The horizontal position is mapped to a color map (VC_1). The vertical position is depicted by an alpha map (VC_3).

moves, the image content is reconstructed. If the camera remains static, a repeating pattern of the background becomes visible that is only disturbed by objects moving through the scanline. If the scanline is placed at the wrong position, some important objects might be missed in the visualization summary, e.g., the red car is not visible in (d). Hence, this idea is extended by incorporating the gaze position to dynamically change the position of the scanline, according to the current point of interest. That way, the image content always includes information about important AOIs. For example, in Figure 5.29 the red car is always visible in the *gaze-guided slit-scan*.

To this point, the resulting image of a gaze-guided slit-scan contains no information about the absolute horizontal and vertical position of a gaze point. However, this information is necessary to interpret the content of the visualization in the context of the whole stimulus. To compensate for this shortcoming, two additional visualization components are derived from the gaze position (Figure 5.30):

- **Horizontal position map:** To avoid visual clutter, the x-coordinate is represented as a separate timeline with a corresponding color-mapping. This timeline is attached on top of the slit-scan, providing an effective visual comparison between the scanpaths of multiple participants.
- **Vertical position map:** The y-coordinate of the gaze point is used to create an alpha map with full opacity at the gaze point and a gradual fade to transparency. The resulting visualization is superimposed with the slit-scan and depicts the stimulus and the vertical gaze distribution.

Table 5.1: Comparison of implemented scanpath similarity metrics. The criteria are rated as (○) not supported and (●) supported for each method accordingly.

	Measure	Abbr.	Temporal Order	Annotation-free	Semantics
AOI	Levenshtein Distance [187]	LD	●	○	●
	Needleman-Wunsch [216]	NW	●	○	●
Trajectory	Dynamic Time Warping [47]	DTW	●	●	○
	Fréchet Distance [35]	FD	●	●	○
Image	Bhattacharyya [51]	BD	○	●	●
	Chi-Square [95]	CD	○	●	●

In the presented examples, slit-scans depict raw data to support the depiction of possible smooth pursuits. Analogous to the fixation-image charts, the approach can also be applied to fixation data, resulting in longer time spans with a consistent visual pattern.

Visual Analytics Framework

Participants who looked at the same objects will create slit-scans that visually resemble each other. Hence, an automatic comparison based on the resulting images seems reasonable. Furthermore, traditional metrics as discussed before in Chapter 3.4.2 (e.g., Levenshtein distance) can be interpreted with the slit-scans. To foster the advantages of visual analytics, the visualization and the metrics are combined in an interactive framework. A set of established metrics (Table 5.1) is included to provide support for scanpath comparison. The metrics are chosen as representatives for either AOI-, trajectory-, or image-based similarity measures. Depending on the metric, some support the temporal order in a scanpath and some require no annotations on the data. Assuming that both aspects are desirable, the trajectory-based metrics are promising. However, the AOI-based methods are more reliable in the context of semantic interpretation. Hence, an annotation-free metric with results similar to the AOI-based metrics would be best. This assessment will be discussed in the next section.

The metrics provide a single value indicating the degree of similarity between two scanpaths. Combined with gaze-guided slit-scans, the presented visual analytics approach helps interpret the discussed measures and compare the results between metrics. Figure 5.31 depicts an overview of the implemented visual analytics framework.

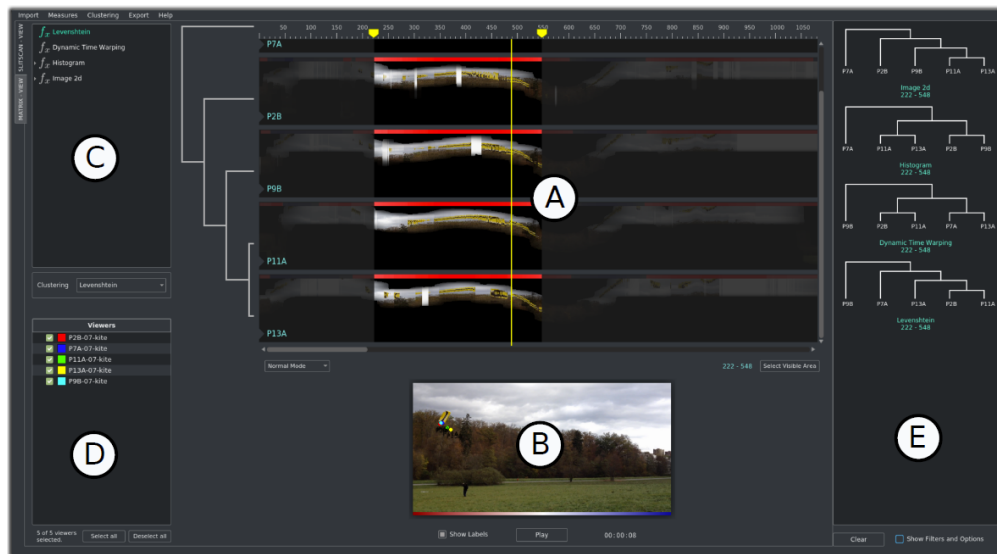


Figure 5.31: Overview of the approach for interpretation and comparison of eye-tracking data based on scanpath metrics. The investigated data shows four participants of the *Kite* dataset. (A) Scanpath representation based on slit-scans; (B) gaze replay; (C) metric selection; (D) participant selection; (E) history of comparison results.

- (A) **Slit-scans:** The slit-scans for the different participant are vertically stacked, facilitating the comparison of scanpaths over time. Selected scanpaths are ordered according to the results of agglomerative hierarchical clustering, based on the currently selected metric. This representation resembles the dendrograms applied to scarf plots and gaze stripes.
- (B) **Gaze replay:** The vertical scanlines forming a slit-scan can partially convey the context of the stimulus. For a detailed view on specific time spans, the slit-scans are linked with a video player that shows a bee-swarm visualization.
- (C)(D) **Filter selections:** An editable list of metrics is displayed to select the measure for performing the clustering. Data from participants can be de-/selected for the analysis. The result is displayed in the slit-scan view.
- (E) **History:** An important part of the presented approach is the possibility to compare the impact of the applied metric on the results of the clustering. Results of a clustering step are saved as small dendrograms in the history view. Depending on the applied metric, clusters might change for a selected time span. If a dendrogram in the history view is selected, its similarity to the other dendrograms is displayed based on cophenetic correlation [269]. This idea was not included in the former approaches and provides quantitative and qualitative information about the applied metrics.

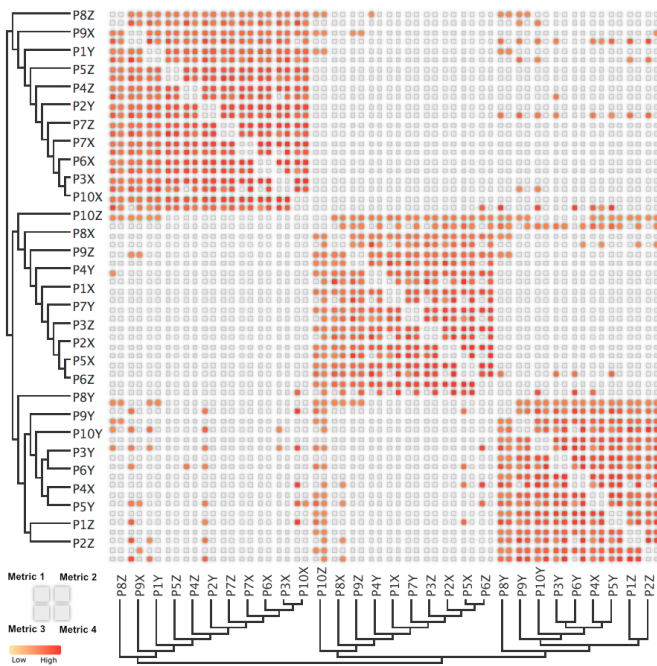


Figure 5.32: Matrix overview of multiple metrics. An individual cell shows the pairwise similarity based on selected metrics. The symmetrical pattern in the matrix is slightly impaired because the order of metrics in a cell is not rotated accordingly to keep the interpretation consistent. Cells can be selected to investigate the corresponding slit-scans in detail.

In addition, an overview of all selected metrics is available in a separate matrix view (Figure 5.32). Pairwise similarity values are color-coded in the cells of the matrix. Selecting a cell displays the corresponding slit-scans of the pair for direct comparison. With this view, correlations between measures can be investigated in detail.

Assessment of Image-Based Measures

To compare the image-based similarity of slit-scans with established metrics, three synthetic video stimuli are assessed with recorded eye tracking data. The dataset contains smooth pursuit patterns that define different groups of viewing behavior.

Scanpath Patterns

Following the approach presented by Haass et al. [132], three artificial stimuli were created to evoke different smooth pursuit patterns. Each stimulus video contains three colored dots (blue, black, and green) that follow different motion paths, as illustrated in Figure 5.33. Ten participants were recorded with the instruction to follow a specific dot color with their eyes. The task was repeated with all colors for each participant (nine tasks per participant), resulting in 30 scanpaths per stimulus. The order of colored dots to follow was counter-balanced between participants using a Latin Square design. As a consequence, the resulting scanpaths of the same task should be more similar to each other than the scanpaths from another task. An appropriate metric would result in clearly separable clusters for the different patterns in a stimulus.

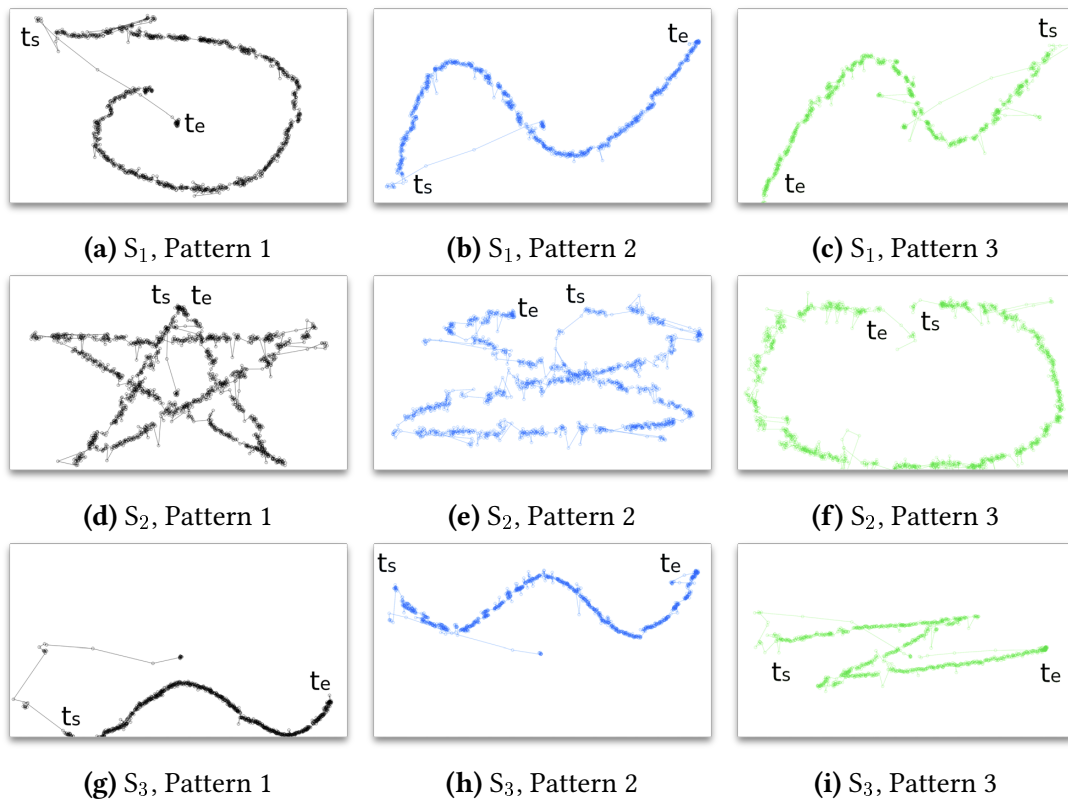


Figure 5.33: Recorded gaze point examples from smooth pursuit patterns for the stimuli S_1 (a–c), S_2 (d–f), and S_3 (g–i).

Experimental Setting

All three stimuli were presented with a resolution of 1920×1080 and 25 frames per second. Their respective lengths were 23, 30, and 23 seconds. A centered black cross was presented before each task to start all participants from the center of the screen. All similarity values between participants were computed based on the raw gaze data. String encodings and trajectories are based on raw gaze points instead of fixations, due to smooth pursuits. The image-based measures use the distribution of hue and saturation values within the slit-scans. The range of the hue and saturation values is divided into 30 equally sized bins.

Results

For each stimulus, hierarchical agglomerative clustering (average linkage) is applied to the recorded scanpaths. This is done for all listed measures in Table 5.1. In order to retrieve the three largest clusters from the hierarchy, the resulting trees are flattened

in a top-down manner. Then the F_1 -scores are calculated for each of the measures, according to the task category the scanpath should belong to:

$$F_1 = 2 \cdot \frac{\textit{Precision} \cdot \textit{Recall}}{\textit{Precision} + \textit{Recall}}$$

Table 5.2 shows the averaged F_1 -scores over the stimuli S_1 , S_2 , and S_3 . The AOI-based approaches result in a correct clustering of all scanpaths for each task category. From the trajectory-based approaches, the dynamic time warp distance (**DTW**) can also clearly separate the clusters, the Fréchet distance (**FD**) results in numerous misclassifications. The image-based measures provide correct results for Bhattacharyya (**BD**), Chi-Square (**CD**) results in some misclassifications.

The F_1 -scores show the overall accuracy of the used measures, but do not provide information on how the calculated metrics relate to each other. Hence, the rank correlations between the similarity values are calculated (Table 5.3). The AOI-based approaches are the reference for the other measures that do not rely on annotation. Consequently, a high correlation to Levenshtein (**LD**) and Needleman-Wunsch (**NW**) indicates a better correspondence between the similarity metric and semantically interesting stimulus regions. The Bhattacharyya distance (**BD**) shows the highest correlation values in comparison with the other metrics, indicating that by including stimulus content into the metric, results comparable to algorithms with annotated data can be achieved.

Table 5.2: Averaged F_1 -scores over the stimuli S_1 , S_2 , and S_3 .

Measure	Category	F_1 -score
(LD) Levenshtein Distance	String-based	1.00
(NW) Needleman-Wunsch	String-based	1.00
(DTW) Dynamic Time Warping	Trajectory	1.00
(FD) Fréchet Distance	Trajectory	0.50
(BD) Bhattacharyya Distance	Image-based	1.00
(CD) Chi-Square Distance	Image-based	0.96

Table 5.3: Averaged Spearman correlations of similarity values over the three stimuli.

	LD	NW	DTW	FD	BD	CD
LD	1.00	0.99	0.67	0.23	0.76	0.6
NW	0.99	1.00	0.67	0.23	0.76	0.6
DTW	0.67	0.67	1.00	0.47	0.72	0.58
FD	0.23	0.23	0.47	1.00	0.33	0.19
BD	0.76	0.76	0.72	0.33	1.00	0.6
CD	0.6	0.6	0.58	0.19	0.6	1.00

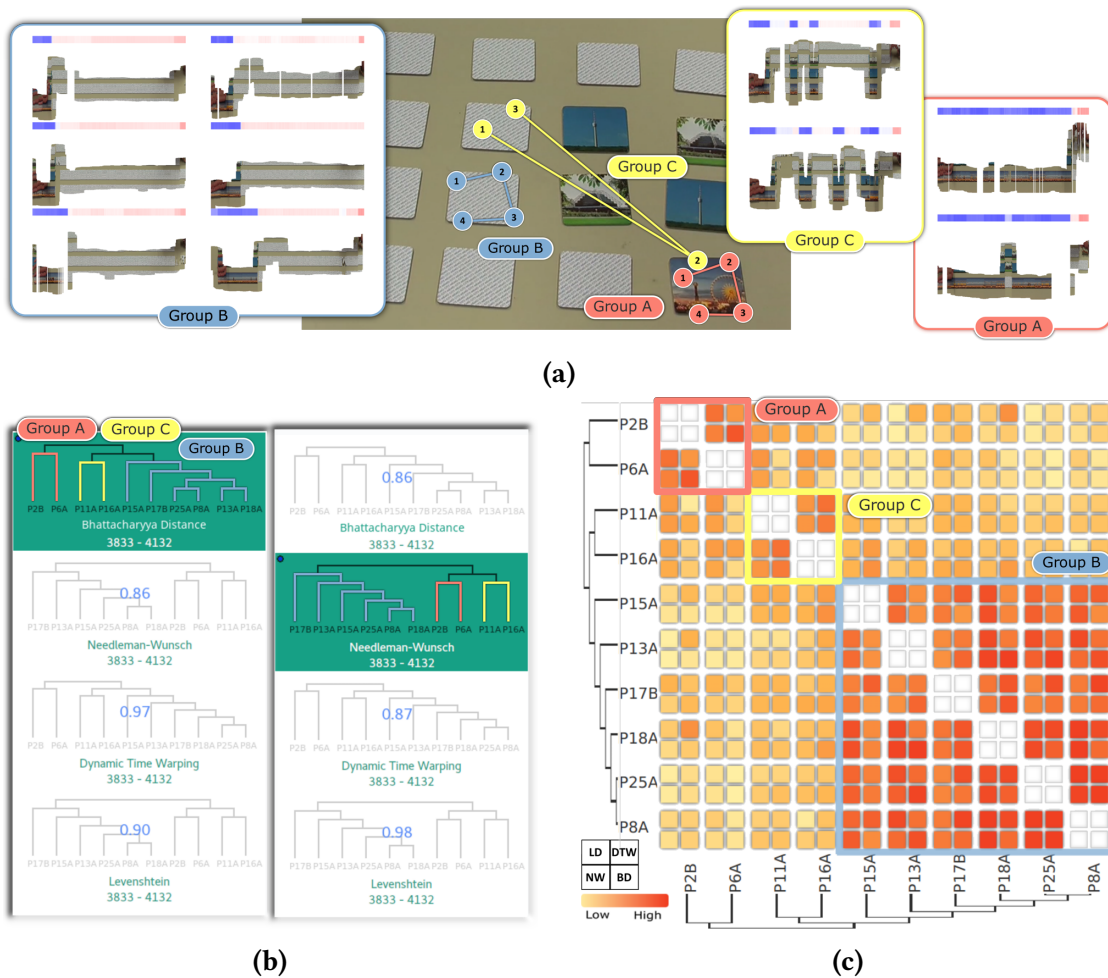


Figure 5.34: (a) Slit-scans of participants watching the *Memory* video clustered in three groups. Group A fixates the uncovered card at the bottom-right corner. Group B fixates the correct covered card at the center. Group C keeps on switching between two different cards. (b) Dendrograms show the clustering results with different metrics. (c) Metric results can be investigated in detail with the comparison matrix.

Example: Memory Game

The approach is demonstrated on the *Memory* dataset. For this video, it is shown how the comparison between metrics can be performed interactively. Figure 5.34 shows a scene from the video. In the presented time span, the card in the bottom-right corner (a Ferris wheel) is the last uncovered card and the corresponding card is located at the center. As illustrated in Figure 5.34, the participants in Group B correctly fixate the covered counterpart card, whereas the participants of Group A fixate the last turned card, and Group C fixates the wrong covered card. The identified patterns for Group A and C indicate difficulties in remembering the correct card.

Figure 5.34 (top) also shows the slit-scans of ten participants assigned to the respective groups according to their viewing behavior, as identified by clustering with the metrics **LD**, **NW**, **DTW**, and **BD**. Especially high correlations can be observed between **BD** and **DTW**, as well as between **LD** and **NW** (Figure 5.34b). All four metrics could correctly separate between the Groups A, B, and C. The main difference between the metrics appears in the later merging steps of these groups. **BD** and **DTW** merge B and C first, whereas the AOI-based metrics merge A and C first. This can be explained by the nature of AOI-based metrics. Participants of Group C partially fixate a covered card that is different to the covered card fixated by Group B, leading to small similarity (due to different AOIs). In contrast, **BD** and **DTW** find similarities between the covered cards due to image content and spatial proximity.

For a full overview of the results, the comparison matrix view can be investigated (Figure 5.34c). Interactive filtering helps explore correlations between metrics indicated by similar colors within the blocks of the matrix. High similarity values for all applied metrics are apparent in the cluster of Group B, where participants could identify the position of the correct card. Selecting a specific cell in the matrix displays the corresponding slit-scans of the participants to help interpret the metric results.

5.3.4 Discussion

Image-based representations of scanpaths provide much potential for comparison tasks and the detection of deviations from regular patterns. All presented techniques do not require AOIs and are applied directly to the recorded data. *Gaze stripes* and *fixation-image charts* show an overview of the data for sequences shorter than one minute. Although adjusting the sampling rate and zooming and panning provides leeway for longer sequences, the temporal scalability of both approaches is limited. In contrast, *gaze-guided slit-scans* reduce the width of one time step down to one pixel, which allows one to display much longer sequences (e.g., *Last Clock* [87]). However, the interpretation of details in the stimulus is often harder to achieve than with the other two techniques.

For future work, the extension to long-term recordings such as feature-length movies should be considered with a combination of slit-scan and thumbnail-based techniques. The slit-scans may provide an overview of the dataset and thumbnails of selected time spans summarize the gaze data to provide more details without frame-wise skimming. The application of image-based visualization to mobile eye tracking is another research topic that provides much potential for future research. The first step in this direction is discussed in Chapter 6.2.

To this point, all examples were based on a single stimulus that was watched by multiple participants. In the following chapter, the analysis of data from mobile eye tracking is discussed. In this case, multiple participants create their individual stimulus videos, which further complicates data analysis for more than one person.

Visual Analytics for Mobile Eye Tracking

This chapter covers research on the analysis of eye-tracking data with dynamic stimuli and participants who actively influence the stimulus. In general, this category is split into two types of experiments:

- **Desktop scenarios:** Examples that comprise experiments that utilize a remote eye-tracking setup. Participants are typically asked to interact with an application on a desktop computer (e.g., a visual analytics tool [3]). Each participant performs individual interactions, needs different time to solve the task, and finally records an individual video with gaze data.
- **Mobile eye tracking:** The second category comprises experiments conducted with a head-mounted eye tracker. A world-view camera records a stimulus video onto which gaze data is mapped. This setup allows the participant to perform tasks requiring mobility that cannot be achieved with a remote setup. With this high degree of freedom for in-the-wild studies, the difficulty for the analysis of the data increases.

Applying point-based techniques to such data is limited to scenarios where gaze coordinates can be transformed into a joint coordinate system. For desktop scenarios, this can be achieved by accessing the screen coordinates of individual components. For mobile eye-tracking, optical markers are often used [232]. However, markers restrict the number of applications for mobile eye tracking. In highly dynamic scenarios, for example, in pervasive eye tracking over long time spans, it is simply not possible to prepare the environment with artificial markers. As an alternative, approaches based on *Simultaneous Localization And Mapping (SLAM)* map gaze data directly to reconstructed 3D surfaces [226]. Such approaches seem promising for future research, especially in combination with mixed reality applications incorporating eye tracking.

This chapter is partly based on the following publications:

- K. Kurzhals and D. Weiskopf. “Eye Tracking for Personal Visual Analytics”. In: *IEEE Computer Graphics and Applications* 35.4 (2015), pp. 64–72 [17]
- K. Kurzhals, M. Hlawatsch, C. Seeger, and D. Weiskopf. “Visual Analytics for Mobile Eye Tracking”. In: *IEEE Transactions on Visualization and Computer Graphics* 23.1 (2017), pp. 301–310 [29]

For the majority of scenarios, semantic interpretation and therefore the annotation of AOIs is often necessary to make existing techniques applicable. Since numerous AOI-based techniques can be applied once the annotation is done, this chapter focuses on two other important aspects of mobile eye tracking:

- *How can eye tracking be applied in the context of personal visual analytics?*
This scenario is particularly interesting because it includes long time spans of data from an individual person that have to be presented in a casual way.
- *How can the annotation of gaze data be improved with image-based techniques?*
Since AOIs have to be identified in each individual video source, the annotation effort increases significantly in comparison to the scenarios described in the previous chapter. The concept of the presented image-based approaches is extended to improve the annotation of gaze data and provide an in-situ analysis during the annotation phase.

The investigation of data from mobile eye tracking is one of the most challenging scenarios for gaze behavior analysis. A wide range of experiments can be covered due to the high degree of freedom with this setup. Hence, it is important to develop methods to handle the resulting data efficiently.

6.1 Personal Visual Analytics

Eye tracking is becoming more affordable for consumers. As an example, games can be interacted with by gaze using low-cost remote eye tracking¹. It is reasonable to assume that soon, the hardware development will advance to provide consumer glasses that allow for pervasive eye tracking [64]. This provides numerous applications for gaze-based interaction with the surrounding world. In general, the application of eye tracking for human-computer interaction is categorized by four groups: *explicit eye input*, *attentive user interfaces*, *gaze-based user modeling*, and *passive eye monitoring*. The categories reach from overt/intentional to covert/unintentional systems [201]:

¹ <https://tobiigaming.com/>, last checked: October 13, 2018

- **Explicit eye input:** Gaze is used to interact consciously with a computer system for controlling purposes. For example, by replacing the mouse with gaze positions.
- **Attentive user interfaces:** Such interfaces do not rely on gaze as an explicit input. The gaze information is rather used implicitly, e.g., for gaze contingent displays that adjust the render quality according to the user's eye movements.
- **Gaze-based user modeling:** In contrast to the previous groups, modeling approaches aim to understand and formalize human gaze behavior and cognitive processes. The detection and prediction of specific behavior (e.g., being attentive while driving) can be used as meta information for further processing.
- **Passive eye monitoring:** Monitoring comprises scenarios when gaze data is only recorded for later processing without a direct influence on the surroundings. This provides useful information for diagnostic purposes or other scenarios such as life logging.

Mainly scenarios in which data has to be investigated retrospectively benefit from visualization and visual analytics. Hence, this chapter focuses on the passive monitoring of gaze data. In particular, the potential of this technique in the context of personal visual analytics and personal eye tracking is discussed:

How can users of eye-tracking glasses recapitulate on their viewing behavior, understand interactions with others and the environment, or just have fun with their personal data?

The possible application scenarios for personal eye tracking cover diverse fields. With the additional information about the user's gaze, important events in the video database can be extracted to facilitate re-experiencing these events. Possible scenarios comprise applications to support self-reflection and self-insight [143] by video analysis with gaze information. This could be the analysis of interaction logs for personal relations with others, vigilance optimization during driving situations, or cognitive activity recognition that can be applied for quantified-self scenarios [180]. For example, users could monitor their reading behavior and time spent on reading texts; a goal might be to read at least 10.000 words a day. Also, catalogs of interest could be generated, depending on objects that attracted the user's attention to serve as recommender systems. For example, the viewing behavior could be analyzed to present similar suggestions for future media consumption. The time spent on a personal visual analytics application strongly depends on the scenario. For example, users who benefit from the analysis for health or social reasons will be more motivated to spend time with the application than users who browse recorded data just for fun.

This chapter further discusses how personal eye tracking can be categorized in the general context of personal visual analytics and what special requirements and challenges

have to be considered for applications. As one example of the visualization of personal eye-tracking data, a new approach, the *AOI cloud*, is presented to display information about the gaze distribution across multiple videos. With this technique, annotated AOIs such as persons can be displayed in an overview by a representation similar to a tag cloud. Additional rings on the AOIs allow for easy navigation through several videos to examine time spans that received the user's attention.

6.1.1 Eye Tracking in the Context of Personal Visual Analytics

Personal visualization and personal visual analytics concern the application of existing and the development of new techniques for data representations and interactions in a personal context. The main question in this context is:

“ How can the power of visualization and visual analytics be made appropriate for use in personal contexts—including for people who have little experience with data, visualization, or statistical reasoning? ”

Huang et al. [154]

The design dimensions of personal visual analytics are investigated first and it is discussed how an application for personal eye tracking fits in. In particular, an example case of *personal encounter analysis* is classified according to these specifications. To this end, the classification introduced by Huang et al. [154] is examined, which consists of four categories with dimensions considering the *data*, *context*, *interaction*, and *insight*:

Data The scope of the recorded data is a combination of data about oneself and data about other people. Data about oneself is recorded by gaze information and by the video camera of the eye-tracking device that captures data about the environment. This data is very personal and has to be handled with care. Under the assumption that eye-tracking devices will become more and more comfortable in the future and comparable to the regular experience of wearing glasses, the effort to record data will be reduced to sensor recording only. Current eye-tracking devices still require elaborate calibration procedures that increase the effort to record data. Regarding the controllability of the data acquisition, the user has partial control whether to record the surrounding.

Context The influence context of mobile eye-tracking analysis is mainly personal to inform the user wearing the device. However, since other people will often be involved in the recorded data, the user could communicate extracted events through social media to involved persons, for example, to recapitulate parts of a conversation. The design context of an application depends on the scenario. In the example case for browsing encounters with other people (Chapter 6.1.3), the application to examine the recorded

data is designed by the researcher. However, the components of the visualization are freely organizable, allowing the user to arrange groups of persons and extract and summarize important personal events in an easily accessible visual representation. For scenarios with automatic data analysis (e.g., recommender systems), predefined representations of the results should be sufficient.

Interaction The degree of attentional demand for interaction also depends on the scenario. In case the analysis is performed automatically and the user has to choose between different results (e.g., recommended media), the attentional demand will be low. For the analysis of personal encounters, the user has to focus on the visualization to investigate interesting events. Hence, high attentional demand is required. High explorability of the data in the application allows users to investigate multiple video streams simultaneously for interesting events that received much attention.

Insight Apart from technical issues, fully automatic analysis of the data can only be applied in a subset of scenarios and for pre-processing. An analysis of subjective events cannot be automated and requires the user to make conclusions of the data. Also, the degree varies to which extracted insight from the application can influence future actions. In the best case, the examination of the recorded data leads to an identification of self-defined misbehavior that can be avoided in future actions. For example, a person who is considered a close friend received less attention than the user would consider appropriate. Being aware of this situation, the user can then spend more time with this person to strengthen their friendship.

In addition to these general design dimensions, some specific requirements for mobile eye tracking have to be considered. These requirements relate to common issues with this technique and to the specific personal context.

6.1.2 Special Requirements

For the personal analysis of eye-tracking data, certain aspects that differentiate personal from professional visual analytics have to be considered. The following characteristics and requirements of personal eye tracking are identified as most relevant.

Accuracy In professional eye tracking, high accuracy of the analysis is critical because research results, product design, security-relevant decisions, or others rely on the quality of the analysis. Fortunately, personal eye tracking is less critical in terms of analysis accuracy. Therefore, there is some leeway in designing personal visual analytics.

Time spans and reasoning artifacts Personal eye tracking will cover much longer time spans than traditional eye-tracking experiments, requiring more time-compressed visual representations. Similarly, different reasoning artifacts are relevant [285]. For example, patterns in the transitions between fixations are of lesser interest than *events* or *objects* extracted from the data (such as people with whom the person interacted). Specific aspects of tasks for personal eye tracking will be complemented by general observations for casual visualization [271].

Semantic information and combination with other information Since personal eye tracking focuses on identifying relevant events or objects, it benefits from linking those to semantic information and embedding them into the context of *outside* information. For example, people identified as being important could be associated with information from their web profile.

Visual interface and application scenario Like any personal visual analytics application, the design of the visual interface has to be easy to use for non-expert users and avoid a steep learning curve. The automatic processing for the analysis should be robust so that there is little or no need for the user to interfere and fine-tune the underlying data mining or computer vision techniques. Similar to many of the apps in mobile personal use on smartphones, visual analytics software for personal eye-tracking will most likely be application-specific. In contrast, professional tools tend to be generic so that they can work with any study setup.

Privacy Personal visual analytics has to incorporate mechanisms to protect privacy because potentially sensitive information is recorded from the environment. Therefore, the analysis needs to be designed to work with the principle of data minimization (e.g., to work with video recordings in which faces of persons or license plates of cars are modified to make them unrecognizable). Also, high data security of the personal gaze data of the user is required [192].

The above aspects will be critical in (1) the design of appropriate visual interfaces, and (2) the development of automatic analysis techniques to be integrated within visual analytics. In summary, it is expected that personal eye tracking will come with many challenging research questions related to design, interaction techniques, visualization, computer vision, pattern recognition, and semantic modeling. While there is substantial research in these areas, the personal perspective will require researchers to devise new variants of existing techniques or develop completely new ones. To illustrate the possibilities of personal eye tracking, a prototype was implemented for a commonly representative scenario: the analysis of personal encounters of a user.

6.1.3 Example: Personal Encounters Analysis

The analysis of interactions between persons plays an important role in psychological and cognitive science (e.g., for joint attention [215]). For a private user, the analysis of personal encounters can also be interesting, be it a self-reflection of social behavior or just for re-experiencing situations that received much attention. In this example scenario, the user was wearing eye-tracking glasses during a recurring event over one week: *the coffee break*. During the coffee breaks, a group between 3–6 people, including the person wearing the eye-tracking glasses, gathered to discuss miscellaneous topics. The recordings during these breaks lasted between 3–9 minutes with a varying set of participants. All participants agreed to be recorded on video if their faces would be anonymized. Considering the privacy issues discussed in Section 6.1.2, this was an important prerequisite for all participants. Also, the recorded audio should not be included in any form of publication of the data. One participant (P1) of a coffee break did not agree to be recorded in any form, so P1 sat next to the person wearing the eye-tracking glasses, being not visible to the camera. This situation exemplifies the issues that occur when other people are recorded on video and have to be considered for the application of personal eye tracking.

Automatic pre-processing of this data requires an algorithm to detect faces in the videos, store them in a database, and recognize the faces when they reappear. In this scenario, the faces are AOIs. Compared to other tasks in computer vision, this can be performed without much user interaction, since there is no semantic gap that requires human interpretation of situations. The user might identify a person once, while the rest of the data is processed automatically. With the information what faces can be seen in the videos and where they appear, a gaze distribution can be calculated by the AOIs of faces and the eye-tracking data. Although computer vision approaches can nowadays be applied for automatic segmentation and classification of such events (e.g., Jasinschi et al. [165]), this example is showcased with manually annotated data since current automatic approaches often face difficulties with changing environmental conditions as in the presented case and ground truth data was preferred to show the visualization.

AOI Cloud Visualization

To show the gaze distribution on AOIs, common visualization principles such as an overview and interactive filtering of the data have to be available. For personal eye-tracking data, the overview of all AOIs and how much attention they received plays an important role. The interactive visualization has to meet the requirements for personal eye tracking and enable the user to browse the recorded video data for events and time spans when a specific object was looked at.

Figure 6.1 shows the implemented visualization approach. The annotated persons (more general: AOIs) are represented as circles consisting of a representative image

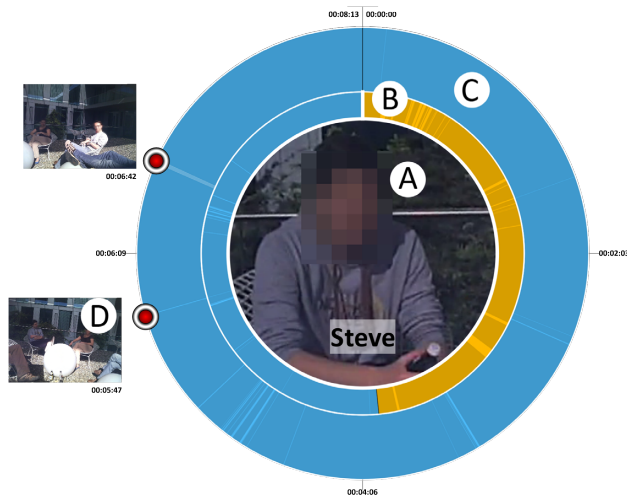


Figure 6.1: Visualization of one AOI: (A) representative image with a label, the radius indicates the attention spent on the person; (B) inner ring with segments for all videos the person appeared in; (C) the outer ring shows the currently selected video; (D) reference images with markers on the outer ring.

(A) and an inner (B) and outer (C) ring. Reference images (D) help browse the content of multiple videos. Radial visualization approaches are applied in cases where hierarchical structures, relationships among disparate entities, or as in this case time series data, have to be displayed in a dense representation [97]. The presented radial approach was chosen due to its compact representation of the temporal dimension on the rings that can be interpreted by using a clock metaphor, the accessibility for novices [98], and its possibilities for fast interactions. The radius of the circle can be determined by an appropriate eye-tracking metric. In this example, the total amount of gaze points on the person from all videos was calculated. Other metrics such as transition counts between AOIs or fixation durations could also be applied, depending on the analysis question. Hence, the visualization approach is independent of the applied metric.

Since some persons appear only in one video and others in three, the difference between the gaze points on the AOI with the lowest value and the AOI with the highest value can be high. This leads to extreme differences in the size of the circles, resulting in the problem that at least one of the AOIs is either too small or too big to be readable. Hence, logarithmic scaling of the metric was applied to adjust the visualization for a better representation of all AOIs. The representative image of a person is determined by the first appearance in the data. Alternative approaches could determine the representative image based on a special event in the data or a profile image from social networks.

The inner ring consists of segments that each represent a video containing the AOI. Hence, the inner circle contains all the videos where the AOI appeared, and the size of a segment is determined by the relative length of the corresponding video. Segments in the inner ring are connected to the outer ring by identical colors. To visualize when an AOI was looked at, an approach similar to AOI timeline visualizations was used. Time spans without gaze on the AOI are displayed darker, whereas time spans with gaze points are displayed with full brightness. This way, important events can be identified



Figure 6.2: The touch-friendly design of the AOI cloud allows for an analysis of the data on mobile devices such as tablets. The user can arrange the visualization individually and explore the data in everyday situations.

efficiently by directly selecting the emphasized time spans. Typical approaches with AOI timelines consider only one video. Here, multiple video stimuli are combined in one visualization to investigate the data more efficiently.

By selecting a segment of the inner ring, a second ring appears outside, representing the selected segment zoomed over the whole ring. Timescales for start and end of the video as well as for the quarters help the user to navigate clockwise through the video. Initially, one marker is available on the rim of the outer ring. It can be moved around the ring to navigate through the video. A thumbnail image next to the marker shows the currently selected frame as a reference to the video content. By clicking on the thumbnail, the corresponding video appears in a separate player window and can be played back directly at the selected position. The user can also create additional markers to select multiple events of potential interest to compare them, or just summarize the gist of important interactions with the person in this video. With this approach, interesting events can be assessed simply by clicking on the thumbnails that represent them.

The complete dataset can finally be visualized by items for each AOI that can be arranged in a layout similar to a tag cloud [298]. Important AOIs are placed in the center of the cloud while less important AOIs appear in the outer regions. With this analogy, the accessibility of the visualization is supported, since tag clouds are familiar to most users and already established in everyday life. From that point on, the user is free to rearrange all items to build groups or rankings of persons, based on subjective criteria. As an example, the user could rank the persons based on friendship relations and investigate if their received attention relates to this ranking (Figure 6.2). Time spans when a person received attention are easily accessible by the inner and outer rings. With this approach, the exploration of multiple video sources is simplified in an easy to understand interactive visualization. Due to the touch-friendly design of the visualization, users can examine their data on mobile devices. This design enables easier integration of the application into the everyday life of the user which is important for long-term use.

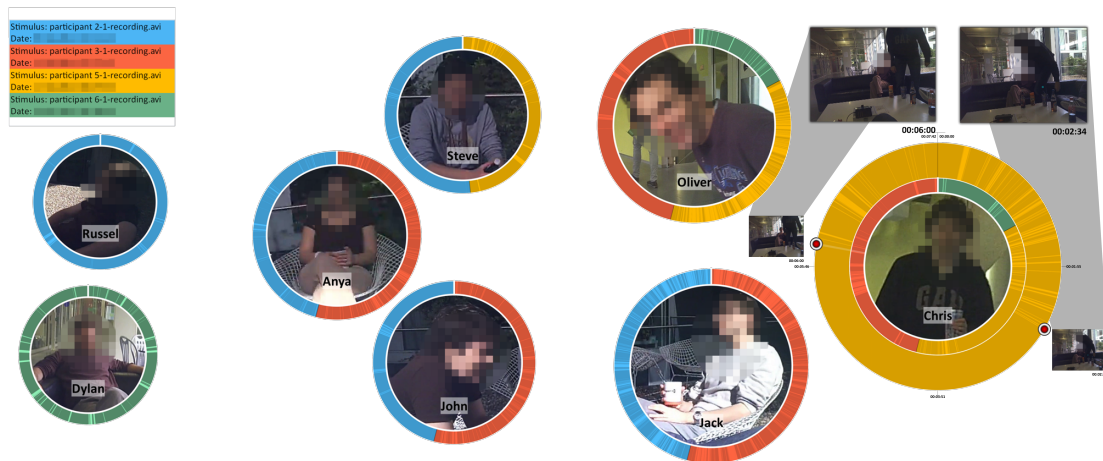


Figure 6.3: AOI cloud for eight persons over four videos. The items are freely arrangeable by the user. In this example, three groups are created: Group 1 (Dylan, Russel), who received the least gaze points; Group 2 (Anya, John, Steve) with medium amount; and Group 3 (Jack, Oliver, Chris) with the highest amount of gaze points.

Analyzing the Coffee Break

Figure 6.3 shows a summarization of four videos from the *coffee break* dataset. Two videos (yellow, green) are from the same session because the person constellation changed after the first record ended. Altogether, eight individual persons participated in the breaks and received different amounts of gaze from the user wearing the eye-tracking glasses. The user organized the participants in three groups, based on the number of gaze points they received:

Group 1 Dylan and Russel appeared just once in different videos. Both were looked at less than the others, especially Russel, who was sitting next to the user, was looked at only when he was talking, since the user had to turn the head to look at him. Dylan was also watched when he was not talking since he was sitting in front of the user. Both persons could have received a similar amount of attention as in Group 2, if they appeared in another video and Russel was seated in a better position.

Group 2 Anya, John, and Steve appeared in two videos and were watched occasionally by the user. Steve could also be shifted to Group 1 since he was looked at only a few times during his attendance in the coffee break.

Group 3 Jack, Oliver, and Chris received most of the gaze points, although the distribution highly depends on the constellation of persons. As an example, Oliver was looked at much in video 3 (yellow), where he, Chris, and Steve were present. During

this coffee break, Chris left the room for half of the time (see markers at 00:02:34 and 00:06:00) where the main focus was on Oliver. In video 2 (red), Oliver received fewer gazes. In this video, as well as in video 1 (blue), Jack was the attention catcher. Since he talked most of the time in both videos, the user looked often at Jack. Hence, he received most of the gaze points although he was only present in two videos.

How much attention a person received in this *coffee break* example strongly depends on the position of the person, their active participation in discussions, and who else was present. Persons that talked less and required the user to turn the head received fewer gazes, especially when an attention-catching person was present. In conclusion, if the user would like to spend more attention on some of the persons from Group 1 or Group 2, talking with these people outside the coffee breaks when Jack is not present to capture the attention might be an option.

6.1.4 Discussion

The AOI cloud provides an accessible approach to investigate the personal distribution of attention over several videos. The visualization approach is not restricted to persons and could be applied to an arbitrary set of objects, assumed that an annotation of the objects is possible. Although the most important AOIs will always be in the center of the initial cloud, a large number of AOIs and videos might reduce the readability of the visualization. Therefore, the scalability of this approach can be improved by additional filtering, concerning the number of AOIs and video segments. By thresholding gaze-related metrics, AOIs that received fewer gaze points could be removed from the visualization. The same approach could be applied to the video segments of an AOI.

The presented visualization focuses on the analysis of individual relations between a person and the user. For future extensions, an analysis of group interactions would be beneficial for a reflection on personal social activity. A comprising set of personal analysis interests could be covered by adding new possibilities to examine the switching focus of attention on different persons and how it correlates with their activities.

Perspectives

This chapter presented a glimpse into the future of mobile eye tracking for personal scenarios. With the presented visualization approach, personal encounters can be analyzed in an easy, accessible way. Mobile eye tracking comprises most scenarios that can be achieved with head-mounted cameras or head tracking. Its main advantage lies in the additional gaze information. In all cases where multiple objects are in the center area of the recorded image, detailed information about the current point of regard on an object can be derived. A typical example is a person looking at a picture collection where the identification of the currently focused picture is not possible

without determining the gaze position. Regarding the visual design of applications, the focus lies on personal scenarios. Hence, designing interfaces to combine mobile eye-tracking data with existing applications for personal visual analytics would be desirable. To extend the possibilities of personal eye tracking in the near future, the challenges linked to the requirements have to be addressed.

Data acquisition To increase the accuracy and accessibility, self-calibrating approaches need to be developed. Current techniques rely on calibration procedures not feasible for a personal application. Also, managing the influence of uncontrolled lighting conditions in the environment bears problems that require further research.

Automatic detection of objects of interest Defining areas or objects of interest solely relying on computer vision might be hard to achieve in the near future. Arbitrary user-defined queries (e.g., searching all cars in the videos of the database that the user looked at) are required to process the recorded data to its full extent. With the advances in computer vision research, this can be achieved already for low-level semantics (this is a car). For high-level semantics (this is my car), semi-automatic approaches and crowdsourcing could bridge the semantic gap that is apparent in automatic approaches. Hence, visual analytics fits well to support such semi-automatic analysis.

Cognitive activity recognition Additionally, the interpretation of the gaze data itself, regarding cognitive processes, has to be considered. Current approaches using cognitive modeling and machine learning to predict and classify gaze behavior (e.g., detecting arousal or vigilance) need further development to provide more information than just distributions of attention. In the example, this information could be applied to weight the AOI circles. Additional information from measured pupil dilation can be included since current eye-tracking devices already record this data and preliminary work to correlate pupil changes with emotional states already exists. Supplementary sensors (e.g., heart rate sensors) can also provide such information and are already combined with mobile eye tracking nowadays.

With currently existing methods, one of the biggest challenges is still the annotation of AOIs in the data. The following section will outline how image-based visual analytics can ease this task and increase the efficiency in comparison to polygon-based bounding shapes annotated by drawing in the video.

6.2 Image-Based Visual Analytics for Mobile Eye Tracking

For most analysis scenarios of mobile eye-tracking data, the annotation of AOIs is inevitable. This often proves to be the most cumbersome process of the analysis phase of an experiment. Every video has to be investigated and coherent AOIs have to be labeled in all of them. Once this troublesome step is taken, a wide range of AOI-based techniques [4] can be applied (Chapter 5.2). For that reason, the visual analytics approach presented in this section focuses on the efficient labeling of gaze data. It simplifies the complex annotation process by reducing the problem to an image-sorting task supported by automatic image analysis. The concept extends on the ideas and experiences with *gaze stripes* [24] and *fixation-image charts* [23] for single video scenarios (Chapter 5.3). To ease the labeling of thumbnails, unsupervised clustering of the data is performed in a pre-processing step. The resulting clusters can be explored with different strategies to identify and label AOI-relevant clusters. To support the search for misclassified elements, different image queries can be applied to retrieve the missing thumbnails and assign them to the correct label. This approach does not rely on the definition of bounding shapes, where the defined shapes might be very different between annotators. Reasons for a low inter-annotator agreement [44] can be investigated easily by looking at all thumbnails misclassified by the annotators.

The remainder of this chapter outlines how this approach relates to other work in this field, what the visualization requirements are, and how they are approached. The approach is evaluated in two ways:

- ▶ The labeled results from this approach were compared with the results obtained by a collaboration partner for the same real-world dataset (Chapter 6.2.5).
- ▶ Additionally, an expert user study was conducted at an eye-tracking conference to collect feedback about the usability of the visual analytics system and to identify the applied strategies during the use of the application prototype (Chapter 6.2.6).

6.2.1 Related Work

The annotation of AOIs can be performed either by annotating the stimulus content directly or by labeling the recorded gaze data.

For direct video annotation, manual and automatic approaches exist to extract objects as AOIs. In the best cases, semi-automatic tracking [48, 259] or automatic approaches with markers [233] and without markers [62, 291] facilitate the detection of AOIs. Tracking can improve the annotation speed, but initial definitions and corrections of bounding shapes are still required.

Fully automatic identification of AOIs without markers requires an algorithm to detect and recognize the corresponding objects. This is a common problem in computer vision that can usually be solved for specific scenarios (e.g., mobile text recognition [178]), but typically, a training phase with all involved AOIs is required. This prerequisite impairs the application of an automatic approach to solve annotation issues for arbitrary experiments. In contrast, the presented approach requires no initial training phase and can be applied to eye-tracking experiments in general.

Labeling the gaze data itself often provides more accurate information about AOIs, since gaze points that were not in an AOI but close to it, for example, due to calibration issues, can be identified and corrected. Therefore, each measured gaze point, or in the aggregated case each fixation, has to be investigated in the video to assign the correct label. This annotation approach is in most cases far more time-consuming than the definition of bounding shapes. For example, Tsang et al. [294] depict fixations labeled this way also by thumbnails. As the authors mention: “This process constitutes a significant amount of time and effort if the number of fixations is large [294]”. Netzel et al. [218] report an average annotation speed of 5 fixations per minute by a similar approach, leading to 140 hours spent on about 40.000 fixations for an experiment. There is some work that improves the annotation step by semi-automatic algorithms. Pontillo et al. [236] present an image-based approach to label fixations by showing images of fixated regions to the analyst for semi-automatic classification of fixated areas. However, the authors apply this approach only to assign fixations to labels, further analysis with statistical or visualization techniques is still required for this annotated data. Also, their approach requires step-wise labeling of the data, while here, automatic clustering of the images in a pre-processing step is applied, which reduces the number of images to investigate.

There are numerous methods to depict large collections of video data, e.g., Luo et al. [198] analyze and visualize news video collections according to an interestingness measurement. The cluster editor view of the approach is similar to storyboard visualizations that depict keyframes of videos in a grid (e.g., the work by Bailer and Thallinger [43], Furini et al. [119]). Fu et al. [118] use a similar concept to visualize multi-view videos that show the same scene from different views. In the presented technique, images can be assigned to AOI labels by dragging and dropping on their representative pictograms. This approach is similar to MediaTable by Rooij et al. [246, 247]. The authors use a bucket-based workflow to categorize videos. Buckets represent media categories and videos are displayed by representative images. However, their application was developed only for video content without any eye tracking information. The principles of their workflow are adapted to provide an efficient means of labeling AOIs with pre-processed data.

This section discusses a new visual analytics approach that allows the efficient comparison of data from multiple videos acquired during experiments with mobile eye tracking.

By including unsupervised clustering techniques in the pre-processing and interactive image queries in the labeling step of the analysis process, annotation results comparable to current state-of-the-art techniques can be achieved, but with far less human effort due to a more efficient annotation process.

6.2.2 Domain-Specific Analysis Process

In order to design a visual analytics approach that facilitates the analysis of the data, the requirements have to be identified first. Based on this, the changes in the analysis process in comparison to a traditional procedure are defined. Although this approach is designed with a specific application scenario in mind, the derived analysis questions apply to a multitude of possible mobile eye-tracking experiments.

Domain Problem Characterization and Design Process

The development of the technique was accompanied by discussions with a collaboration partner. He is an experienced eye-tracking researcher (8 years at the time this work was published) at the Stuttgart Media University in the field of print media. Following the principles of user-centered design [214], the domain problem characterization was addressed and the requirements of the collaboration partner were identified. In an iterative process, the visualization design was discussed, adjusted, and improved. For the domain problem characterization, the following points were identified as the main analysis questions to be answered:

- Q₁ What was the distribution of attention between AOIs?*
- Q₂ When was a specific AOI investigated for the first time?*
- Q₃ In which order did the participants look at the AOIs?*

These are three basic questions for eye-tracking analysis tasks (Chapter 4.2.1), which allow the application of established visualization techniques and descriptive statistics to present the extracted information in an appropriate way. To answer the first question, a gaze histogram is applied, showing the average gaze duration of all participants in relative time. This representation is consistent with the one used by the collaboration partner. The inclusion of additional metrics would be possible to address further research questions. For the other questions, scarf plots for all participants are included. Choosing suitable visualization techniques for interpreting the labeled data was a minor issue during the design process of the visual analytics approach because established methods were identified to be appropriate.

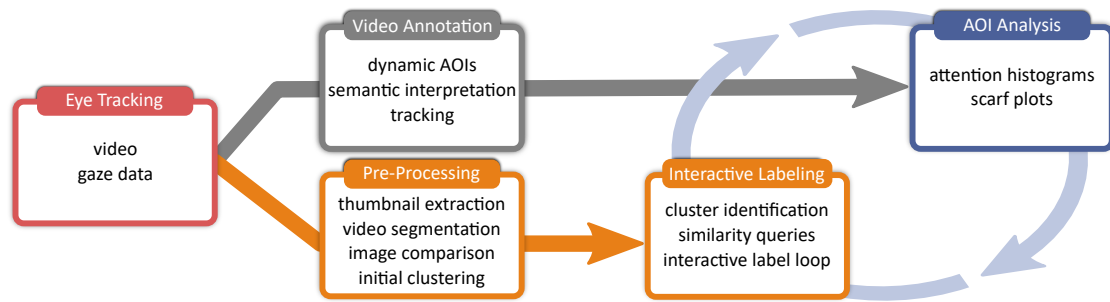


Figure 6.4: Analysis process for mobile eye-tracking data (red): AOIs can be defined by annotating the video (gray) or the gaze data (orange). The annotation based on gaze data is facilitated by automatic pre-processing and reduced to an interactive image labeling task. For the analysis of mapped gaze data on AOIs (blue), common visualization techniques such as histograms and scarf plots can be applied.

Analysis Process

With the domain problem characterization and requirements analysis, the typical workflow associated with mobile eye-tracking studies was investigated. Here, AOI labeling was identified as the most time-consuming step. Therefore, the main focus of this work is on making the labeling process more efficient.

For this purpose, the common process for annotating eye-tracking videos with AOIs is changed. In the traditional annotation process, the video is investigated and dynamic AOIs have to be defined on the video image (Figure 6.4, gray). This procedure has to be repeated for each video (i.e., each participant) recorded in the experiment. Furthermore, consistent labeling of AOIs is critical for the analysis. Gaze points are mapped to the AOIs by automatic hit detection; the analyst is usually not involved in this mapping process. Consequently, errors from imprecise bounding shapes or offsets in the calibration of the eye tracker might be missed.

By investigating the image content of fixated regions directly, the analyst has the possibility to decide whether a gaze point was on an AOI, or not. However, looking at the image content of each measured gaze sample individually would need much more time than the definition of dynamic AOIs. Therefore, the analysis process is split into two stages (Figure 6.4, orange): (1) a pre-processing step that can be performed automatically and clusters gaze data based on the investigated stimulus content, and (2) the subsequent analysis of these clusters itself. The analysis can be interpreted as a loop between the interactive labeling of the clusters and the coupled interpretation of the results with the provided visualization techniques: all changes in the labeling can be directly interpreted with the other visualizations. Then, the other clusters can be investigated based on the insights derived from the visualizations.

6.2.3 Pre-Processing

As illustrated in Figure 6.4, eye-tracking data has to be recorded and pre-processed before the interactive labeling step. For the examples, the SensoMotoric Instrument (SMI) head-mounted Eye Tracking Glasses 2.0 were used. However, the visual analytics method does not make use of any specific characteristics of the SMI glasses. Therefore, it works with any eye-tracking device that provides gaze coordinates of fixations and a video of the stimulus. The pre-processing phase is separated in four steps: thumbnail extraction, video segmentation, image comparison, and clustering. Figure 6.5 shows an overview of the pre-processing steps.

Thumbnail extraction Each gaze point provides an x- and y-coordinate mapped to the corresponding video recorded by the scene camera of the eye tracking glasses. Around the gaze position, a thumbnail is cut out of the video image, representing the currently watched region. This step is identical with the procedure for gaze stripes described in Chapter 5.3. In general, an increased crop area for the thumbnail is advantageous for the detection of image features, but impairs the interpretation of what was investigated by the participant during the experiment.

Video segmentation This step describes the temporal segmentation of the video. First, the number of relevant images is reduced by taking advantage of the temporal coherence of the underlying video and gaze data. Fixations on a specific area typically result in a sequence of images that are similar. Therefore, a comparison of thumbnails from subsequent video frames is performed and the images are aggregated until they drop below a similarity threshold. Depending on the applied similarity measure, the threshold can be adjusted to achieve longer or shorter segments. For the applied measure (see next paragraph), a threshold of 0.4 still provided good segmentation results without aggregating different stimulus regions (Figure 6.6). This aggregated sequence of thumbnails is referred to as *segment* in the following. A segment is represented by the first thumbnail of the sequence. The aggregation is stopped for a segment when more than two frames are missing in the gaze data between consecutive thumbnails. This can happen when the eyes are not recognized for a short time span by the eye-tracking device. For the investigated data, this segmentation step reduces the number of images for the subsequent clustering step to approximately 10% of the original thumbnails, removing redundant images from fixations on the same regions. Other experiments that involve smooth pursuit eye movements should aim for smaller segments, since the motion of the underlying stimulus content might be hard to interpret when it is represented only by a single image [23].

Image comparison As described above, thumbnail similarity is compared for two reasons: (1) segmentation of the image sequences and (2) clustering of the remaining

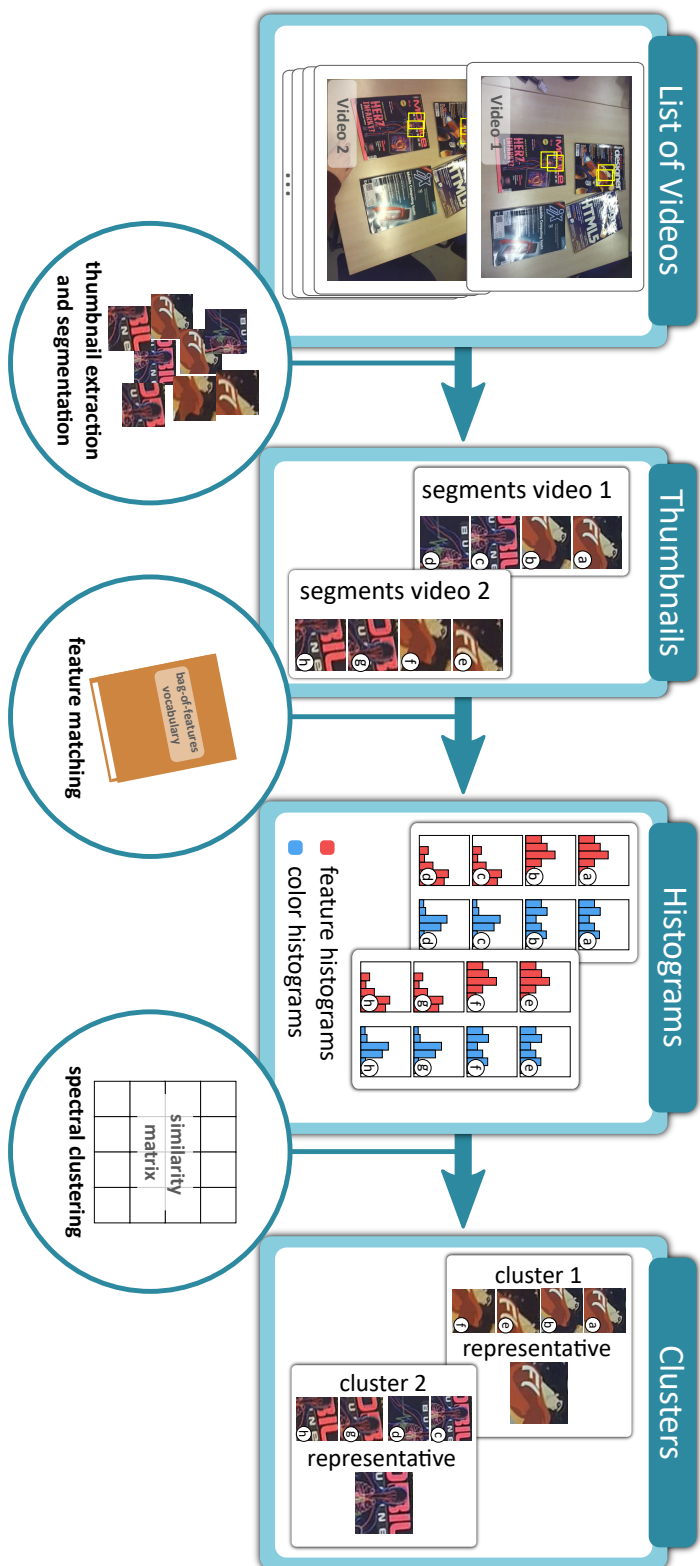


Figure 6.5: Overview of the video segmentation and image clustering process: Thumbnails are extracted from all videos and temporally aggregated. The segment representatives are compared using a bag-of-features approach. Clustering is performed on the resulting similarity matrix.

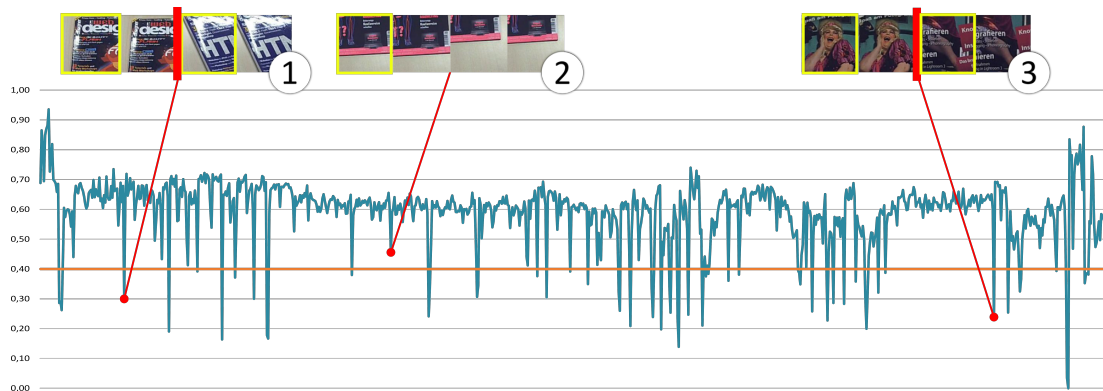


Figure 6.6: Segmentation of a thumbnail sequence: Changes in the image sequence lead to similarity values below the threshold (① ③) and start a new segment. Smaller changes due to short saccades or head movement are aggregated in the same segment (②). The first element of a segment is chosen as representative (yellow border).

segments. From the numerous possibilities to compare images, two approaches are combined that require only a few parameters and can be applied to arbitrary image sequences (Figure 6.5). The first similarity value is calculated from extracted SIFT features [197] using a bag-of-features approach [168]. The features are extracted from each thumbnail and create a feature vocabulary, using k-means clustering. For the tested examples, a set of maximal 200 features per image and a vocabulary size of $k = 500$ led to good results for the segmentation. In general, the size of the vocabulary depends on the number of regions to differentiate. It has to be considered that a large vocabulary size might cause overfitting. With the vocabulary, feature histograms can be derived for every image. At this point, this approach is typically applied to train a classifier for a specific image category. Since this would already require ground truth data for the AOIs, the similarity between the feature histograms of two images is calculated using the inverted Bhattacharyya distance [51]. The extraction of SIFT features depends on the quality of the investigated images; some of the analyzed thumbnails provide only a few or no features to extract. To compensate for that, a second image similarity measure is included, based on color histogram comparison. Both similarity values, feature-based and histogram-based, are aggregated equally. In the case that the feature recognition of an image fails due to a low number of recognized features, only the color histogram value is applied.

Clustering Unsupervised clustering of the thumbnails is performed on their similarity matrix, using self-tuning spectral clustering [316] that only needs a maximum number of clusters as a parameter. For the pre-processing, this should be at least the number of required AOIs. Since irrelevant regions and large AOIs will lead to sub-clusters, the maximum number of clusters should be adjusted accordingly. For

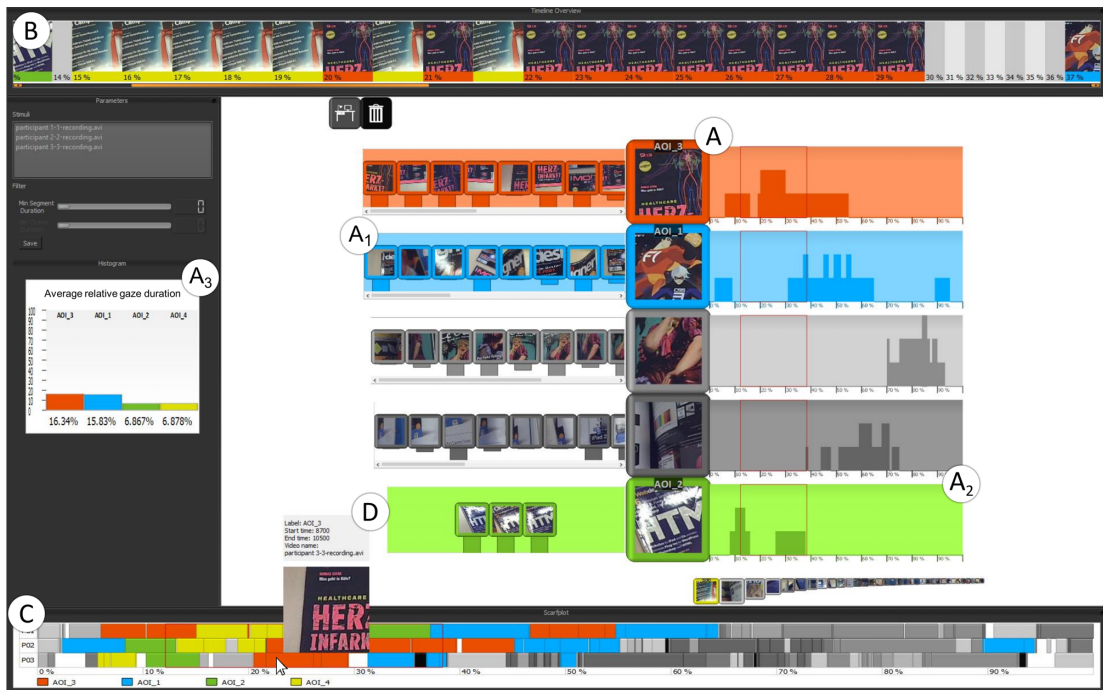


Figure 6.7: The main view consists of four different elements: (A) The cluster view lists all clusters sorted by their accumulated duration. (A₁) To the left of each cluster representative, the cluster elements are displayed. (A₂) To the right, histograms for the clusters are shown. (A₃) The total gaze duration on the labeled clusters is displayed in the histogram on the left. (B) The timeline overview at the top presents the clusters that are viewed by the majority of the participants. (C) The scarf plots display for each participant which clusters were investigated. (D) A tooltip shows additional information when hovering over scarf plot segments.

the experiments, 50 clusters separated the images sufficiently to initialize the labeling process. For each cluster, a representative is determined by calculating the thumbnail that is most similar to the other thumbnails in a cluster.

The similarity measures could be replaced by others that apply better to the specific requirements of the recorded data. Therefore, it is referred to Smeulders et al. [266] for an overview of other possible image-retrieval techniques. As a result of the pre-processing, a list of clustered thumbnails and the calculated similarity matrix are acquired.

6.2.4 Analytics Environment

The analytics environment was designed in a way that the analysis process is effectively supported (see Figure 6.4) and that the questions (Q₁–Q₃) of the collaboration partner can be efficiently answered. For this, a number of components (see Figure 6.7) were

implemented which allow an effective analysis of the data and provide important information related to these questions. The analysis is performed on the clustering results from the pre-processing step. So far, the clustered data does not contain any semantic interpretation of AOIs and misclassified segments can appear in the clusters. Therefore, the analytics environment supports an intuitive labeling process, the detection of falsely clustered elements, and the modification of clusters.

Main View

The *main view* allows performing all AOI analysis tasks and parts of the interactive labeling. To further improve the annotation, additional views are provided: the *cluster editor* to inspect, modify, create, or delete clusters; the *video player* to investigate the video stimuli and gaze behavior of individual participants and to search for segments by defining AOIs directly on the video. With these different views, it is possible to apply different strategies to label and analyze the data.

Cluster view The central component is the cluster view (Figure 6.7 ^(A)). It lists the clustered segments of the eye-tracking data. Each cluster is depicted by a cluster representative computed during the clustering process, which is shown enlarged on a vertical axis. To the left of each cluster representative, all segments inside the cluster are shown ^(A₁). They are sorted according to their similarity. The initial view shows the thumbnails with the lowest similarity on the left. In this way, the user gets an impression of the quality of the cluster and mismatching thumbnails might be found directly without additional exploration. To the right, a histogram shows the occurrence of the cluster accumulated over all participants ^(A₂), i.e., the histogram value is determined by the number of participants that looked at the respective segments contained in the cluster. Since the recorded videos have different durations, the timeline representations are calculated in relative time in order to make the data comparable between participants. To provide a quick overview of the potentially most important parts of the data, the cluster list is sorted according to the accumulated gaze duration of the clusters.

The cluster view serves as a starting point for the analysis by allowing the investigation and labeling of the pre-computed clusters; label colors and names can be assigned to the clusters in this view. It is also possible to modify the clusters in this view by dragging individual elements to other clusters. However, if the clustering quality is not satisfactory and larger modifications are required, this is better performed in the cluster editor, which can be opened by simply double-clicking on cluster representatives. Coupled with the interactive labeling process, the components for the AOI analysis are updated accordingly. The histograms ^(A₂) can help partially answer the question when an AOI was visited the first time (**Q₂**). However, this visualization is better suited to find out if there were time spans during the experiment when many participants

looked at the same AOI (e.g., reading the caption of an article only in the beginning). An even more aggregated histogram is shown on the left (A₃). It shows the average relative gaze duration of all labeled clusters. One can directly see which AOIs received most attention by the participants (Q₁). In general, additional descriptive statistics could be integrated into this view. These histograms do not allow solving all of the mentioned analysis questions efficiently. For this, two additional views are integrated into the framework, i.e., the *timeline overview* and the *scarf plots*.

Timeline overview The timeline overview is displayed on top of the framework (Figure 6.7 B). This view shows a cluster representative on the timeline if the number of participants looking at the cluster segments is above a user-defined threshold. In this way, the timeline overview presents a summary of what the majority of participants looked at, i.e., it can be seen as an accumulated histogram showing only the clusters with much attention over time. The part of the timeline that is currently shown is also marked in the other views with a red box so that the user can analyze how this summary correlates with the cluster histogram and the scarf plot on the bottom. Furthermore, label colors are also shown to ease the identification of clusters and the temporal position is displayed as a percentage of the relative video length. With the timeline overview, answers for questions Q₂ and Q₃ in terms of an *average scanpath* can be easily found by looking for the first appearance of a cluster or by investigating the order of the clusters on the timeline. While vertically stacking the cluster representatives for the same time would ease the interpretation, they are stacked horizontally to keep the view compact and avoid vertical scrolling. However, since the feedback from the user study (Chapter 6.2.6) showed that users had problems with this view, this component should be improved in the future with a better design. Furthermore, the timeline provides only information accumulated over all participants; a detailed analysis of individual participants is not possible with it. For such an analysis, scarf plots are integrated.

Scarf plots The scarf plots at the bottom show the data of individual participants (Figure 6.7 C). The length of a block corresponds to the duration of a segment, i.e., how long a participant looked at a specific region. Initially, different shades of gray are automatically assigned to the clusters. These are replaced by the label color when the user assigns a label to the cluster. By double-clicking on a specific segment in the scarf plot, the cluster editor opens the corresponding cluster the segment belongs to. The analyst can then label the cluster and its segments will appear in the color of this label in the scarf plots. Gray segments between segments of the same color can be an indicator of faults in the clustering results. The analyst can investigate these segments and label the respective clusters, coloring the scarf plots iteratively with every new cluster. Hence, the loop between the labeling and the direct analysis of the labeled data can be repeated until all relevant information is found or the complete dataset is labeled.

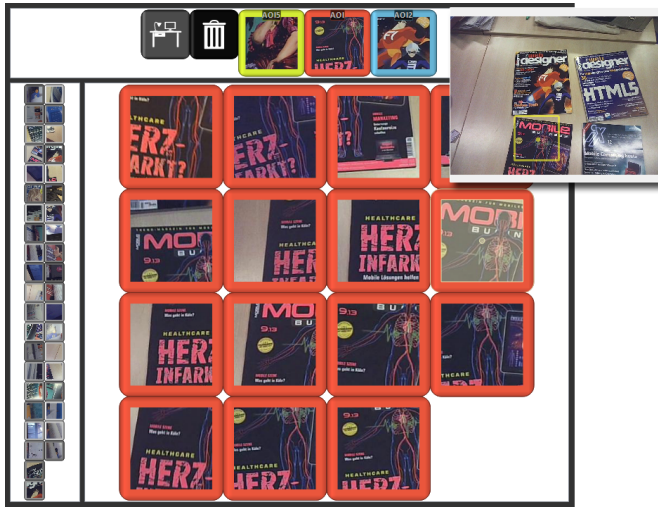


Figure 6.8: The cluster editor allows editing, merging, and deleting individual clusters. It shows a list of all clusters and in the center the elements of the selected clusters. By dragging and dropping cluster elements, they can be assigned to other clusters or deleted. Labeled clusters and their elements are shown with the label color. When hovering an element with the mouse, a tooltip shows the full video frame with the gaze point and thumbnail border marked.

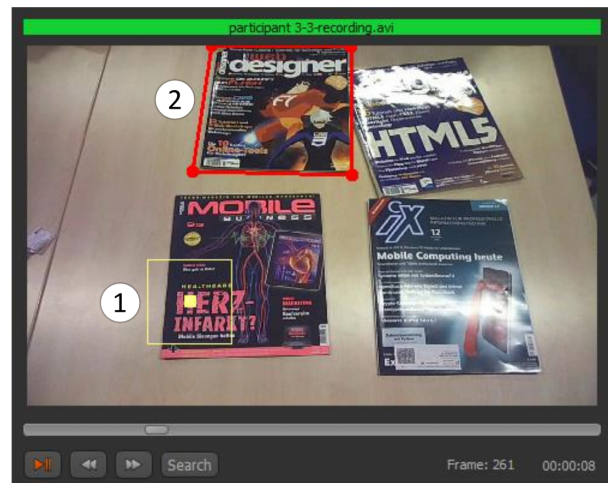
With the scarf plots, it is easy to see when and how long different participants looked at specific clusters, providing detailed information to answer the questions Q_2 and Q_3 . When hovering with the mouse over a block of the scarf plots, additional information are provided (Ⓧ): an image showing what the participant looked at, the duration of the block in milliseconds, the label, and the name of the corresponding video. With this view, it is possible to see how the gaze of individual participants moved between different AOIs. Viewing behavior can also be compared between participants to detect outliers or common patterns, comparable to the techniques described in Chapter 5.2.

Cluster Editor

The effectiveness of the analysis with the main view strongly depends on the quality of the clusters, i.e., how well they represent specific AOIs in the stimuli. The results are only meaningful if a cluster contains all relevant elements of an individual AOI. Since the clustering algorithms are not perfect, manual verification and modification of the clusters are necessary. For this, the cluster editor (Figure 6.8) was developed. The cluster editor is divided into three parts: a list of labeled clusters at the top, a list of non-labeled clusters on the left, and the segments of a selected cluster in the center. Clusters are selected by clicking on them; their elements are then shown. Unwanted elements of a cluster can be deleted (by dragging the element on the garbage can symbol) or moved to other clusters (by dragging the element on a cluster representative).

Dragging an element onto the desk symbol allows collecting different thumbnails of potential interest (e.g., ambiguous thumbnails) for further inspection later on. Integrated image search further supports editing the clusters. It is possible to select a thumbnail and let the system search for similar thumbnails in the same cluster, in the other clusters,

Figure 6.9: ① Gaze positions are marked with a yellow square and bounding box. ② A polygonal area can be marked (red) in the video frame to search for all similar looking thumbnails.



in all clusters, or in unlabeled clusters. The found thumbnails are then ordered according to the image similarity, derived directly from the similarity matrix.

Image queries can be processed efficiently by sorting the row of the similarity matrix of the currently searched thumbnail. With this function, it is quite easy to find similar thumbnails in other clusters or perform cluster corrections, e.g., splitting a cluster into two clusters: a thumbnail with the specific content for the new cluster is selected and used for image search. The thumbnails are then ordered according to image similarity allowing an easy rubber band selection to create a new cluster. In some cases, it is hard to decide from the thumbnails alone if a certain element belongs to a cluster because the surrounding context of the thumbnail is missing. Therefore, the complete video frame is shown when hovering with the mouse over a thumbnail in the cluster editor. In the video frame, the gaze position and the thumbnail bounding box are marked. This eases the interpretation of thumbnails by providing the full context of the stimulus.

Video Player

Without knowing the content of the stimuli, it is difficult to understand the data and what the clusters and thumbnails represent. Therefore, it is possible to watch the recordings of individual participants in a video player. The video player (Figure 6.9) shows the video recorded by the eye-tracking hardware together with the respective gaze positions (Figure 6.9 ①). This supports not only the general understanding of the data by showing the stimulus content, but also allows following the gaze movements of an individual participant. Furthermore, it is possible to select a polygonal area in the video frame (Figure 6.9 ②) and perform an image search with the selected area. The thumbnails that are most similar to the selected region are then shown in the cluster editor. In this way, it is possible to create clusters by drawing AOIs in the video as in the traditional approach.

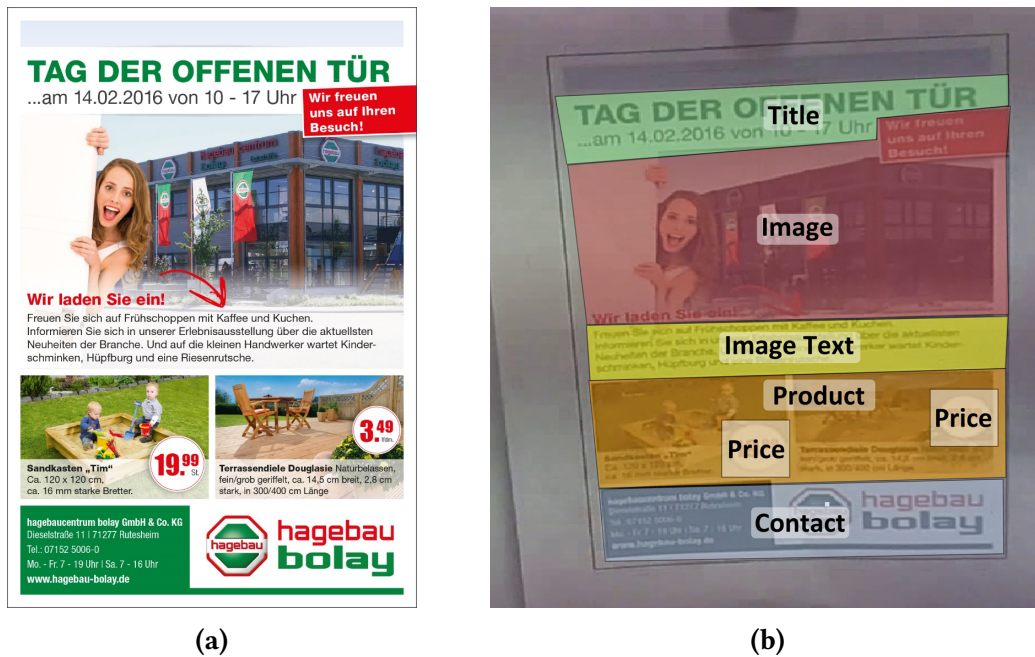


Figure 6.10: (a) Visual stimulus from the investigated user study. (b) Annotated regions on the stimulus represent the labels of the relevant AOIs.

6.2.5 Example: Print Media Study

To evaluate the method on eye-tracking data from a real experiment, the technique was applied to the data from the collaboration partner. This particular dataset was recorded for an eye-tracking experiment in a hardware store. The experiment was part of a research project at the Stuttgart Media University. The question was how different designs of printed advertisements affect the perception of viewers. Since the evaluation of the experiment was also performed by the collaboration partner with traditional methods, a comparison of the annotation time and the results of the study is possible.

Design of the Eye-Tracking Experiment

The stimuli were categorized in three sections of intentional dimensions: *Sale*, *Image*, and *Event*. For each of these dimensions, two different design categories were tested with eye tracking in a between-subject design and an additional post-test interview to compare the gaze distribution on the different stimuli. The first category was a positive design according to the intention, the second category was not. The entire experiment took place at the point of sale in a hardware store, where regular customers were faced with one stimulus after they agreed to be a part of the experiment. In total, 90 persons participated in the experiment. For the comparison, one of the six stimuli

Figure 6.11: Annotation times for thirteen videos. The dynamic AOIs were labeled directly in the video by drawing and tracking bounding shapes. Annotator 1 and 2 applied the presented approach. Their different completion times result from different strategies to solve the task.

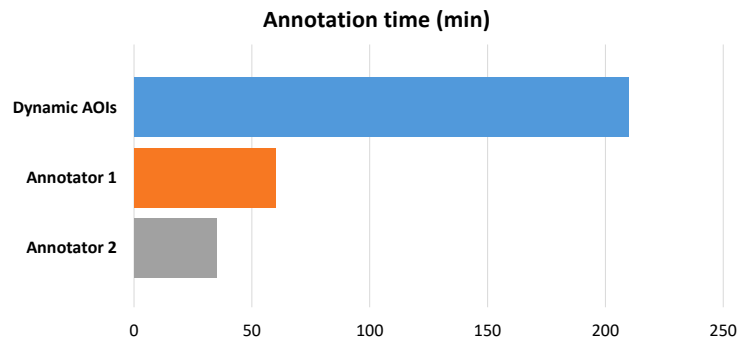
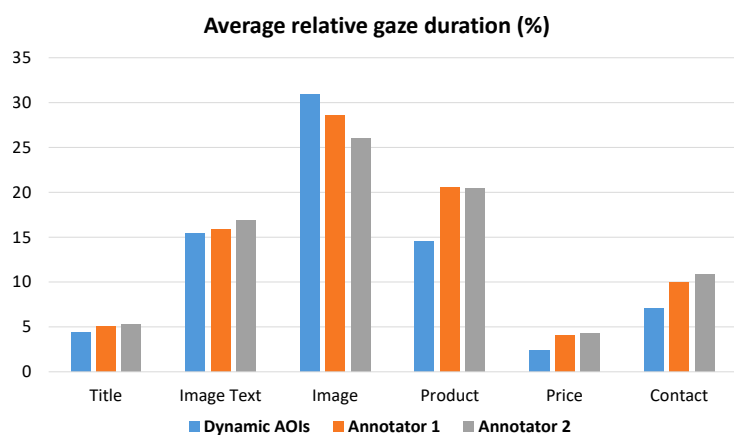


Figure 6.12: Comparison of AOIs with the presented approach. Differences mainly result from gaze points in boundary regions of the AOIs.



(Figure 6.10) was investigated. It is defined as a design with the intention *Event* but also consists of other design objects, like prices and product pictures. 15 participants looked at this stimulus for approximately 20 seconds each, two of them were removed due to calibration issues.

Comparison

The traditional analysis procedure of eye-tracking data, including the annotation of dynamic AOIs directly in the videos, was performed by a group of four students using the SMI software *BeGaze*. To achieve comparable results between the approaches, the same stimulus regions were defined as AOI labels (Figure 6.10b) and the extracted segments were assigned accordingly. The labeling process was performed by two developers, providing an impression of how efficient trained users can apply the technique. Figure 6.11 shows the resulting annotation times. For dynamic AOIs, each video has to be investigated individually and the consistency of the annotated areas has to be considered. This requires concentrated drawing and correction of polygons over time.

In comparison to this traditional approach, the annotation process could be reduced to approximately 17%–30% of this time, depending on the applied annotation strategy.

Annotator 1 used direct searches of areas by drawing query regions in the video. Since these searches required time to process, the annotation was slower. Annotator 2 iterated through the clusters, using the pre-processed similarities between thumbnails to search segments belonging to an AOI, which provided almost instantaneous query results. Since a non-optimized algorithm was used to search for the arbitrary image queries, Annotator 1 would also have finished earlier if the calculations were more efficient.

In Chapter 6.2.2, the requirements and research questions for the visual analytics approach were discussed. First, the gaze distribution on different AOIs (Q_1) has to be derived. Since this was also the main question of the collaboration partner, their results, derived by dynamic AOIs, can be compared with the presented approach. Figure 6.12 shows the average relative gaze duration on the different AOIs.

In summary, congruent results could be achieved with the image-based approach with differences between 0.4%–6% in comparison to dynamic AOIs. For an average video duration of 20 seconds, the maximum difference between the calculated gaze duration on the AOI *Product* is 1.2 seconds. Between the two annotators who applied the new approach, the differences are between 0.1%–2.6%. Especially fixations in border regions lead to variations in the annotation results. Depending on the size of the drawn AOIs, some fixations might be neglected even if an annotating human user might assign it to the corresponding label. With the image-based approach, such difference in the inter-annotator agreement could be solved by displaying the issued thumbnails in the editor view and let the user decide where a segment belongs to.

To answer the other two questions from Chapter 6.2.2, when AOIs were watched (Q_2) and in which order (Q_3), the annotated scarf plots (Figure 6.13) can be interpreted. For example, it can be identified which participants focused more on the images (red, orange) and when participants started to read the image text (long yellow segments). The black areas indicate that the segments were moved to the garbage can, since they did not belong to any AOI.

6.2.6 Expert User Study

The example showed how trained users can apply the technique. To gather qualitative feedback how untrained but eye-tracking experienced researchers can work with the new technique, a user study was conducted at the *Symposium on Eye Tracking Research and Applications (ETRA 2016)*. Due to temporal restrictions, a smaller dataset was used in this study: the dataset consists of three videos with the participant standing in front of four magazines, looking at the covers and picking up one of them to browse through (Figure 6.9). All three videos have an approximate length of 30 seconds. The four

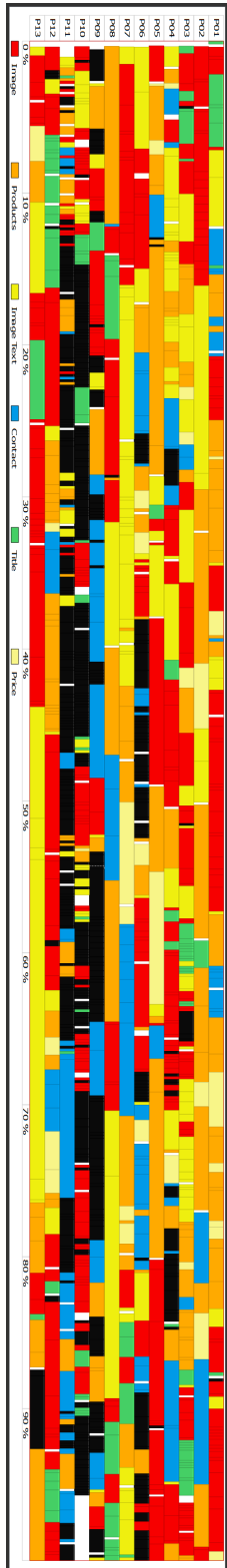


Figure 6.13: Scarf plots of 13 participants labeled with our approach. Black areas depict segments that were removed due to gaze points outside the AOIs.

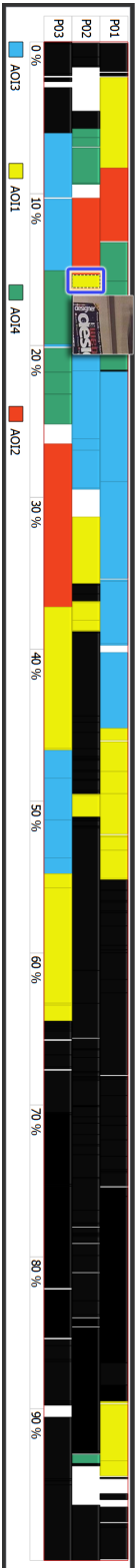


Figure 6.14: Scarf plots of the data that was analyzed in the expert user study. A yellow segment (marked dark blue) was misclassified by two participants.

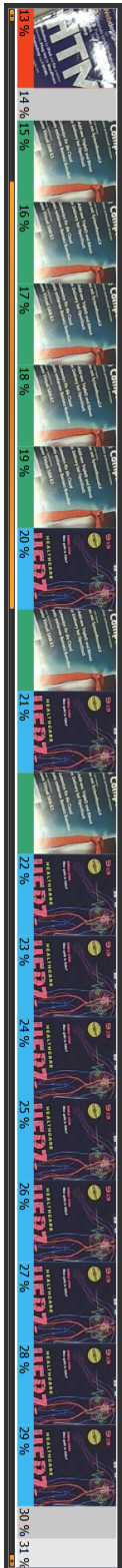


Figure 6.15: Timeline overview showing the longest time interval when two participants looked at the same AOI.

magazine covers are the AOIs, starting with AOI₁ in the upper-left corner and ending with AOI₄ in the lower-right corner. The data was recorded with a free-viewing task as a showcase for the expert study.

Six experts (age 28–40 years) with different degrees of experience in eye tracking (5–10 years) were asked to use the technique to label the four AOIs. Their research fields were psychology, software engineering, virtual reality, and spatial cognition. Additionally, a freelance developer of eye-tracking applications participated. The study took about 45 minutes on average, including an introduction and a demonstration of how to use the different components. Each expert was introduced to the software by a two-sided sheet, explaining the main functions and views. As an example, one cluster from the dataset (not relevant for the following task) was labeled and analyzed to show all functionalities. Then, the experts were free to apply all available functions to label the four AOIs, in order to answer three questions: about (1) the order in which participants looked at the AOIs, (2) the gaze distribution, and (3) the longest common time span two participants spent on the same AOI.

The experts were asked to start with defining the labels of all four AOIs to proceed with the labeling in parallel, in order to prevent that they are slowed down by a sequential search of individual AOIs. An additional questionnaire was handed out to rate the visualization components and collect qualitative feedback about the approach.

Results

First, the results for the three questions, that the experts answered by interpreting the different visualization views, were investigated:

Table 6.1: Which was the order the participants looked at AOI₁ – AOI₄?

Participant	Correct Order
P ₁	AOI ₁ → AOI ₂ → AOI ₄ → AOI ₃
P ₂	AOI ₄ → AOI ₂ → AOI ₁ → AOI ₃
P ₃	AOI ₃ → AOI ₄ → AOI ₂ → AOI ₁

For the first task (Table 6.1), Figure 6.14 shows the scarfs plot of the data. The four task-relevant clusters were labeled, the other clusters were removed. For P₃, all experts answered correctly, for P₁, all but one answers were correct. For P₂, two experts gave the wrong answer. In this case, the segment from AOI₁ between 10% and 20% (Figure 6.14) was not labeled correctly.

For the second task (Table 6.2), the gaze distribution indicates that the attention on AOI₁ and AOI₃ was higher than on the other two AOIs. For AOI₂ and AOI₃, the standard deviation is zero, since all relevant images were labeled by the experts identically.

Table 6.2: What was the average relative gaze duration on AOI₁ – AOI₄?

AOI	Mean	Standard Deviation
AOI ₁	15.68%	0.95%
AOI ₂	6.87%	0.00%
AOI ₃	16.34%	0.00%
AOI ₄	6.91%	0.86%

Differences in the labeling result from gaze points in border regions, for which it is difficult to decide if they belong to an AOI.

Table 6.3: What is the longest time interval with two participants looking at the same AOI?

Correct AOI	Correct Time Interval
AOI ₃	20%–29%

The third task (Table 6.3) could be solved with the timeline overview (Figure 6.15). However, it was observed that the interpretation of this view was not clear at the beginning. In combination with this concrete question and a repetition of the explanation from the beginning, the experts claimed that they finally understood how the view works. Hence, the resulting intervals were correct for all experts.

Questionnaire

The experts were asked to rate the visualization components on a Likert scale from 1 (*not helpful*) to 6 (*very helpful*) with the option to give no rating. The questionnaire also contained free-text questions about the used strategies to solve the task and suggestions for improving the visualization and the analysis process.

Table 6.4 shows the results of the questions about the visualization components. The overall system was rated very useful; especially the cluster editor turned out to be the most useful component to solve the given tasks. All experts stated that they can imagine using the technique for their experiments—except for one expert who stated that their experiments are very standardized and therefore the technique would not be directly applicable. However, some of the components were less used by some of the experts. Two experts did not rate the video player since they did not use it except for some initial testing. One expert rated the timeline overview as not helpful, since the task could also be solved with the other visualization components. Another expert rated the cluster view as less helpful, since the complete time for the annotation was spent in the cluster editor.

Table 6.4: Answers to the question: How useful was the visualization component? 1 (*not helpful*) – 6 (*very helpful*).

Visualization Component	Mean	Standard Deviation
Cluster View	4.5	1.5
Scarf Plots	5.8	0.4
Timeline Overview	4.3	1.7
Cluster Editor	6.0	0.0
Video Player	5.3	1.0
Overall System	5.6	0.5

From the free-text comments, additional suggestions could be derived to improve the usability and the visual representation. In general, the need for more convenient interaction techniques in the cluster editor was stated by most of the experts. For example, the experts missed hot-keys to interact quickly with the editor and order the clusters individually. In the current implementation, the clusters are always sorted in decreasing order of their total duration. Two experts mentioned that the timeline overview might be replaced by a Gantt chart [121], since people might be more familiar with such a representation. Two experts also mentioned that the main cluster view contained too much information. Since the segments of a cluster were also represented in the editor, they stated that the left part of the visualization (Figure 6.7 (A₁)) was not important to them and could be removed from the visualization. One expert missed the information to which video a thumbnail belongs. As a suggestion, another label on the thumbnail showing the video ID could be included.

Applied Strategies

In order to solve the task, the experts applied different strategies. Given the set of described possibilities, the following labeling and analysis strategies were identified:

Video Investigation The video stimulus was relevant to the experts in two situations: (1) for the initial search for the AOIs and (2) for the interpretation of thumbnails in the context of the video. Although it was possible to perform search queries by drawing AOIs in the video (like in the traditional analysis approach), the experts used this function just at the beginning of the task. The main purpose of the video player was to investigate the context of a segment. The experts often selected one of the segments and looked at it in context of the whole video image. Typically, only a couple of consecutive frames were investigated for ambiguous gaze point positions. Except during the initial demonstration, the video player was not used to play longer time spans of one of the involved videos.

Segment Similarity Search Experts using mainly the cluster editor picked images from the clusters to search for similar images in all other or unlabeled clusters. This was usually performed by searching for specific thumbnails of a labeled cluster, either very similar or very dissimilar to the current representative. Query results were then investigated and corresponding thumbnails that belonged to the searched AOI, as well as unlabeled segments that belonged to one of the other AOIs were labeled in parallel. Searching for similar thumbnails just in unlabeled clusters results in an iterative reduction of the set of thumbnails that have to be investigated.

Sequential Cluster Browsing One expert followed the systematic approach to select each cluster after the other to either decide if its content belongs to one of the AOIs or can be removed from the data. All irrelevant clusters were placed in the garbage can. This approach was also followed by Annotator 2 in the use case (Chapter 6.2.5). Although it might seem costly to have to look through all clusters and images, each image is typically investigated only once. In tasks where every segment requires a label and not only a subsection of the images needs to be labeled, this approach can be very efficient. This approach requires the analyst to know the AOIs and which segments can be discarded, which is typically the case in hypothesis-driven experiment settings.

Scarf Plot Annotation One expert mainly focused on identifying long segments in the scarf plots. By selecting one of the unlabeled segments, the editor showed the corresponding cluster. Labeling and correcting this cluster led to the colorization of the respective segments in the scarf plots showing other time spans with attention on this specific AOI. With this approach, the annotation time is reduced by focusing on the most relevant long segments first. In many cases, small segments of the same AOI as the investigated long segment are labeled as well, without the analyst having to look at them again.

In summary, the experts applying the last two strategies were the most efficient ones. In a thorough analysis scenario, it is suggested to identify all relevant AOIs at the beginning and then proceed through the clusters or scarf plots to either assign the correct labels or mark the thumbnails as irrelevant for the analysis.

6.2.7 Discussion

The presented technique provides eye-tracking experts with an overview of the gaze distribution on different AOIs, even for multiple videos with unconstrained conditions from different participants. The example showed that the annotation with the presented technique is far more efficient than the common state-of-the-art approaches based on dynamic AOIs. Annotation results could be further improved by letting multiple annotators label the data. Issues in the inter-annotator agreement can then be checked

by looking at ambiguous segments again. This approach could be integrated into the cluster editor by adjusting the borders of ambiguous thumbnails with the colors of the different AOIs they have been assigned to. The user could then filter for the most ambiguous elements and decide where they belong to.

Because of the user-centered design of this approach, the main focus lies on the analysis of hypothesis-driven experiments with predefined AOIs. For an application to unconstrained scenarios, such as the one presented in Chapter 6.1.3 for personal encounters, interesting areas have to be discovered during the analysis. With the presented approach, clusters could be identified if the participant attended to an object. If no gaze was spent on potentially interesting objects, the current approach would exclude this data. With more specific knowledge about the environment, the approach could also be adapted to include clusters where no gaze was spent on an interesting area. This would require additional visual coding of these elements. In general, the approach could be applied to any time-dependent image series to identify and label similar content, provided that the applied similarity metric is appropriate for the comparison.

For future work, an extension of the approach to long-term experiments should be considered. For example, scenarios such as car-driving where long video sequences with some static (i.e., the dashboard elements) and some highly dynamic AOIs have to be analyzed. This is especially challenging since the dynamic content (e.g., a short moment without attention) can be hard to identify in the recorded data.

To further improve the scalability, the technique can be extended by an interactive classification component. When a sample of the recorded data has been labeled, the applied bag-of-features approach can directly be used to train a *Support Vector Machine (SVM)* classifier with the labeled segments as positives and the other clusters as negative samples. Time segments from new participants could then be analyzed by the trained classifiers and depicted in the editor before assigning them to the labels. With an appropriate sample size, this idea could be further extended by deep learning methods which provide currently good results for classification tasks.

Conclusion

In this thesis, the principles of visual analytics were applied to video and eye-tracking data in order to improve the coupled analysis of these data sources. To achieve this, the thesis combines knowledge from video visualization and evaluation methodology as follows:

- ▶ **Video visualization:** Visualization techniques for attention-guidance and movie analysis provide ideas on how to display video data in an abstracted way to analyze it in combination with gaze data.
- ▶ **Evaluation of visualization:** The evaluation of visualization techniques plays an important role in validating developed concepts. Including eye tracking into the methodology provides new possibilities to gain insights into participants' behavior during a user study.
- ▶ **Visual analytics for eye tracking:** The technical contributions focus on new approaches for the analysis of gaze data and videos with and without AOI annotations. The investigated scenarios mainly comprise the analysis of a single stimulus watched by multiple participants and the efficient annotation and examination of mobile eye-tracking data.

This chapter summarizes the thesis and provides an outlook on how the presented work can be applied for further research. First, the previous chapters are summarized (Chapter 7.1), concluding with an overarching discussion (Chapter 7.2), and future directions for research (Chapter 7.3).

7.1 Summary of Chapters

This thesis comprises work on visualization techniques and evaluation methodology with a special focus on eye tracking. The topic is investigated in both directions: eye tracking to evaluate visualization and applying visualization to analyze eye-tracking data. This approach covers the topic more thoroughly than unilateral research.

Visual Support for Video Analysis The low-level computer vision techniques (image comparison, optical flow, shot detection) described in Chapter 2 were combined with interactive visualizations to provide effective means for video visual analytics. Techniques for exploratory data analysis (EDA), knowledge discovery in databases (KDD), and information retrieval (IR) facilitate analytical reasoning for the support of existing, or the formalization of new hypotheses. As an example, an approach developed for visual movie analytics was presented. It showcased how data from video and text sources can be abstracted on hierarchical timelines to support the annotation of time spans in movies with high-level semantics. In the subsequent chapters, similar concepts were applied to depict annotated gaze data.

User-Based Evaluation of Visualization As discussed in Chapter 3, established qualitative methods such as think aloud, questionnaires, and expert reviews were applied to evaluate the implemented work in this thesis. Furthermore, the repertory grid was discussed as a complementary means to extend the methodology. Quantitative research was performed with performance studies quantifying error rates. Eye tracking extends the quantitative methods by providing spatio-temporal measures of gaze positions. It was discussed how eye tracking can be integrated in existing evaluation procedures and how it is currently applied in visualization research. The depiction of *speaker-following subtitles* was compared with traditional subtitles. In this case, eye tracking shows that with speaker-following subtitles, significantly less attention is spent on text and more on the actual content, making it a promising alternative for future applications.

Visualization of Eye-Tracking Data Chapter 4 discussed the state of the art for the visualization of eye tracking data. A taxonomy of visualization techniques was presented, considering the analysis task and categories related to gaze data, the visualization, and the stimulus. According to the analysis task, existing approaches are applied to answer questions categorized in: *where*, *when*, *who*, *compare*, *relate*, and *detect*. With respect to this taxonomy, existing techniques lacked methods for dynamic stimuli, especially for data from mobile eye tracking. Hence, the techniques discussed in the following chapters contributed to advance the current state of the art. Furthermore, a benchmark dataset was presented to test new visualization techniques.

Analyzing a Single Video and Multiple Participants Chapter 5 focused on techniques for the analysis of a single stimulus and data from multiple participants. The presented space-time cube and motion-compensated heat maps provide static overviews of the spatio-temporal data. With shot-sensitive clustering of gaze points, potential AOIs are identified without the need to skim through the video. The implemented *ISeeCube* framework was extended by techniques for AOI-based analysis, i.e., AOI timelines, scarf plots, and AOI transition trees. The presented image-based techniques, i.e., gaze stripes, fixation-image charts, and gaze-guided slit-scans aim at providing rich information about a dynamic stimulus for point-based visualization. The presented techniques are suitable for an overview without annotations and for scanpath comparisons based on image similarities.

Visual Analytics for Mobile Eye Tracking Chapter 6 considered the scenario of mobile eye tracking. It was discussed how pervasive eye tracking could be applied in the future for personal visual analytics scenarios. The presented *AOI cloud* is a visualization for the overview of AOIs and fast navigation through multiple videos. Furthermore, an approach for the efficient annotation and analysis of mobile eye-tracking data from multiple participants was presented. The visual analytics approach is based on the image-based techniques applied in the previous chapter. It was evaluated in two ways: a comparison with the annotation procedure from an established software suite and with an expert user study conducted with external eye-tracking experts. The results showed that with significantly less time and effort, annotation results comparable to established methods can be achieved.

7.2 Overarching Discussion

To draw meaningful conclusions, it is necessary to evaluate the presented work in the context of the research questions stated in Chapter 1. Hence, the three questions are addressed in this thesis as follows:

7.2.1 Research Question 1

How can we enhance/abstract video material to support specific tasks?

Research Question 1 poses a general question that applies to all implemented techniques related to video content. Here, we can differentiate between the techniques for videos without and with eye tracking. The techniques in Chapter 2 focus on video without eye tracking. The attention-guiding visualizations improve video material and render a new video as a result, wherein interaction with the visualization is not necessary. The results provide valuable insights into how to distribute attention between multiple

objects in a video. The potential of video visual analytics is presented for the analysis of movies with subtitles and script text. The developed multi-level timelines summarize semi-automatic annotations for multiple hours of video. The rich semantic information provided by movie scripts restricts this approach to a subset of possible video stimuli. Hence, a direct application to eye-tracking data is limited. However, the presented concept provides visual abstractions that are modified and applied in techniques such as the AOI timelines and scarf plots (Chapter 5.2.2). The discussion on videos with eye-tracking data is covered under Research Question 3.

The visual analytics techniques presented in this thesis follow the information seeking mantra [264], providing an overview and multiple levels of detail, always including the video stimulus as the highest level of detail. This facilitates all analysis tasks discussed in Chapter 4.2.1, because one can investigate the visualization, identify potentially important time spans, and analyze those in detail. The abstraction of the video material is based on complementary data sources such as text and gaze data.

7.2.2 Research Question 2

How can we leverage eye tracking to evaluate visualization techniques?

Research Question 2 is approached by surveying current applications of eye tracking in visualization research (Chapter 3.4.4). This research shows that eye tracking is used to examine the distribution of visual attention, sequential characteristics of eye movements, and for the comparison of gaze sequences and participant groups. The work from other research fields related to eye tracking shows that additional methods such as cognitive modeling, data fusion with other time-oriented sources, and retrospective think aloud might also be beneficial for visualization research in the future. Hence, eye tracking should be incorporated into existing methodology, as suggested by the presented evaluation pipeline (Chapter 3.4.2). Furthermore, exploratory data analysis for hypothesis building becomes more important if applied to eye tracking of complex visual analytics frameworks. As a consequence, it is necessary to further extend the pool of existing techniques with new visual analytics approaches for the analysis of complex gaze data.

7.2.3 Research Question 3

How can we improve the state of the art of visualizations for eye tracking?

Research Question 3 concerns in particular techniques for the analysis of video with eye-tracking data. To answer this question, this thesis surveys current methods (Chapter 4.3) and identifies important but less represented categories of a taxonomy. According to the presented taxonomy, a lack of interactive techniques for dynamic stimuli with active

and passive content becomes notable. The techniques in Chapter 5 provide interactive techniques for dynamic stimuli without active content changes. The presented space-time cube, motion-compensated heat map, and the image-based techniques provide the means for exploratory data analysis without AOIs. With semantic annotations on important objects and regions of a stimulus, the AOI-based techniques can be applied. AOI timelines provide an overview when an AOI was visible and when participants looked at it. Scarf plots, in combination with interactive comparison methods, support the identification of common and outlier behavior. For dynamic stimuli with active content, data from mobile eye-tracking is analyzed (Chapter 6). The important difference for such data is the fact that each recorded participant records an individual video that is hard to compare with point-based methods. Consequently, AOIs are necessary and often, manual annotation is required for each individual video. The developed visual analytics approach supports an efficient annotation of thumbnails on gaze positions. Once the data is annotated, existing analysis techniques, for example, the presented *AOI Cloud* can be applied.

To prove their applicability, the developed techniques were also applied to investigate data from conducted user studies. For example, *ISeeCube* was used during the evaluation of subtitles (Chapter 3.4.3) for exploratory data analysis: A pilot study was conducted and the hypotheses were derived based on statistical results and on patterns identified with the visualization. Space-time cubes were also included in the publication to communicate the results.

This thesis covers all three questions for the type of data that was investigated. Eye tracking of videos without interaction plays an important role in controlled lab studies with predefined stimulus properties. The presented techniques help investigate the results of automatic processing steps and visualize the combination of video and gaze data on different levels of abstraction. For the evaluation of visualizations on a perceptual level, current and future experiments will generate data that can be analyzed with the proposed techniques. For experiments including interaction with the stimulus, techniques such as the STC are less suitable. Although not tested, the image-based methods might be more appropriate to be adapted for interactive stimuli. For example, gaze stripes provide much information about the stimulus even when recordings of participants cannot be synchronized. Generally, mobile eye tracking covers the majority of scenarios, because the experimental setting emulates a natural interaction of the participant with the environment. However, for the analysis of visualization and visual analytics in desktop environments, a remote setup with additional measures (e.g., interaction logs) is more reasonable. But for the increasing count of mobile applications and for collaborative scenarios, the inclusion of mobile eye tracking is necessary.

The topic provides opportunities for future research directions that were not fully addressed in the focus of this work. These directions will be discussed in the following.

7.3 Future Directions

As stated in the introduction, this thesis focuses on eye tracking and visualization in two directions: (1) eye tracking for the evaluation of visualization and (2) visualization/visual analytics for the analysis of eye tracking. Both directions have great potential for future work that can build on the findings from this thesis.

Extending Evaluation Methodology

As discussed in Chapter 3.4.4, existing evaluation methodology should be further extended with eye tracking. For example, events such as the BELIV workshop¹ foster the development of new evaluation procedures for visualization beyond traditional methods. Since the beginning of this workshop, numerous papers on eye-tracking methodology for visualization have been presented there. This can be seen as an indicator for both, the interest of applying eye tracking for evaluation, and for the potential of extending existing methodology in the context of visualization and visual analytics. One important part of this process is an interdisciplinary collaboration to gather expertise and findings from empirical studies to advance from the investigation of gaze distributions and scanpath sequences to deriving models for complex visual analysis scenarios. Such models could also serve as simulations for user behavior to improve visualization design.

This thesis provides methods for the extraction of findings and surveys the current state of the art of this topic. Future work should extend the techniques with respect to exploratory data analysis for hypothesis building, data fusion with complementary measures, the dissemination of insights, and the availability of tools for researchers without a background in computer science.

Pervasive Eye-Tracking Analysis

Mobile eye-tracking data was declared as one of the most complicated scenarios to analyze. The presented approaches focused more on the examination of multiple short videos than on the analysis of long-term recordings. Under the assumption that eye tracking will become ubiquitous, it is necessary to extend existing analysis techniques accordingly. Machine learning will play a significant role in such scenarios for the automatic processing of video and gaze data. As stated in Chapter 1, human interpretation will be necessary for some of the discussed analysis tasks (Chapter 4.2). It is reasonable to assume that future computer vision approaches will be able to provide answers to questions considering *where*, *when*, and *who* for trained scenarios. Automatic comparison, relation, and the detection of similarities and outliers is also possible, but the final reasoning step, concluding *why* some effect occurred, will often be left to a

¹ <https://beliv-workshop.github.io>, last checked: October 13, 2018

human analyst. This is especially the case for untrained scenarios with unexpected events. Hence, future work should aim at automatic processing to ease the reasoning for the analyst as much as possible.

The first step in this direction is the automatic detection of AOIs. For example, the visual analytics approach for the annotation of gaze data (Chapter 6.2) is conceptualized to be extended by automatic classifiers. Hence, the analyst could begin the annotation process while a classifier is trained the background with the input from the analyst. The visual analytics framework could be extended to suggest classifications for new elements that are easy to verify with the visualization. Such an approach could iteratively improve the quality of the automatic processing with the help from a visual interface.

In summary, if evaluation methodology for visualization and visual analytics is extended by eye tracking and visual analytics is also applied to evaluate the data, insights can be derived that surpass what is currently possible with existing methods. One important step is the automation of parts of the analysis process. Here, image-based techniques have much potential because they provide input for computer vision techniques and are easily interpretable for a human analyst.

Author's Work

- [1] T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl. "State-of-the-Art of Visualization for Eye Tracking Data". In: *Proceedings of EuroVis State of the Art Reports*. 2014, pp. 63–82 (on pages 5, 7, 75, 83).
- [2] T. Blascheck, K. Kurzhals, M. Raschke, S. Strohmaier, D. Weiskopf, and T. Ertl. "AOI Hierarchies for Visual Exploration of Fixation Sequences". In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2016, pp. 111–118 (on pages 6, 68, 82).
- [3] T. Blascheck, M. John, K. Kurzhals, S. Koch, and T. Ertl. "VA²: A Visual Analytics Approach for Evaluating Visual Analytics Applications". In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 61–70 (on pages 6, 8, 72, 77, 145).
- [4] T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl. "Visualization of Eye Tracking Data: A Taxonomy and Survey". In: *Computer Graphics Forum* 36.8 (2017), pp. 260–284 (on pages 5–7, 61, 75, 80, 83, 85, 86, 157).
- [5] M. Burch, K. Kurzhals, and D. Weiskopf. "Visual Task Solution Strategies in Public Transport Maps". In: *Proceedings of the International Workshop on Eye Tracking for Spatial Research*. 2014, pp. 32–36 (on page 6).
- [6] M. Burch, K. Kurzhals, M. Raschke, T. Blascheck, and D. Weiskopf. "How Do People Read Metro Maps? An Eye Tracking Study". In: *Proceedings of the 1st International Workshop on Schematic Mapping*. 2014, pp. 1–4 (on page 6).
- [7] M. Burch, K. Kurzhals, N. Kleinhans, and D. Weiskopf. "EyeMSA: Exploring Eye Movement Data with Pairwise and Multiple Sequence Alignment". In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2018, 52:1–52:5 (on page 6).
- [8] M. Höferlin, K. Kurzhals, B. Höferlin, G. Heidemann, and D. Weiskopf. "Evaluation of Fast-Forward Video Visualization". In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (2012), pp. 2095–2103 (on pages 6, 22, 43).
- [9] M. John, K. Kurzhals, S. Koch, and D. Weiskopf. "A Visual Analytics Approach for Semantic Multi-Video Annotation". In: *Proceedings of the 2nd Workshop on Visualization for the Digital Humanities*. 2017, pp. 1–5 (on pages 4, 7, 11).
- [10] M. Koch, K. Kurzhals, and D. Weiskopf. "Image-Based Scanpath Comparison with Slit-Scan Visualization". In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2018, 55:1–55:5 (on pages 5–7, 63, 88, 94, 121, 134).

- [11] R. Krüger, F. Heimerl, Q. Han, K. Kurzhals, S. Koch, and T. Ertl. "Visual Analysis of Visitor Behavior for Indoor Event Management". In: *Proceedings of the 48th Hawaii International Conference on System Sciences (HICSS)*. 2015, pp. 1148–1157 (on pages 6, 104).
- [12] K. Kurzhals, M. Höferlin, and D. Weiskopf. "Evaluation of Attention-Guiding Video Visualization". In: *Computer Graphics Forum* 32.3 (2013), pp. 51–60 (on pages 4, 6, 7, 11, 22, 44, 67).
- [13] K. Kurzhals and D. Weiskopf. "AOI Transition Trees". In: *Proceedings of the Graphics Interface Conference*. 2015, pp. 41–48 (on pages 5–7, 82, 88, 94).
- [14] K. Kurzhals, M. Burch, T. Pfeiffer, and D. Weiskopf. "Eye Tracking in Computer-Based Visualization". In: *Computing in Science Engineering* 17.5 (2015), pp. 64–71 (on pages 5–7, 40).
- [15] K. Kurzhals, F. Heimerl, and D. Weiskopf. "ISeeCube: Visual Analysis of Gaze Data for Video". In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2014, pp. 43–50 (on pages 5–7, 63, 81, 88, 94).
- [16] K. Kurzhals and D. Weiskopf. "Space-Time Visual Analytics of Eye-Tracking Data for Dynamic Stimuli". In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2129–2138 (on pages 5–7, 80, 84, 87, 94).
- [17] K. Kurzhals and D. Weiskopf. "Eye Tracking for Personal Visual Analytics". In: *IEEE Computer Graphics and Applications* 35.4 (2015), pp. 64–72 (on pages 6, 7, 89, 146).
- [18] K. Kurzhals and D. Weiskopf. "Visualizing Eye Tracking Data with Gaze-Guided Slit-Scans". In: *Proceedings of the IEEE Second Workshop on Eye Tracking and Visualization (ETVIS)*. 2016, pp. 45–49 (on pages 5–7, 88, 94, 121).
- [19] K. Kurzhals and D. Weiskopf. "Exploring the Visualization Design Space with Repertory Grids". In: *Computer Graphics Forum* 37.3 (2018), pp. 133–144 (on pages 5–8, 40, 53).
- [20] K. Kurzhals, C. F. Bopp, J. Bäessler, F. Ebinger, and D. Weiskopf. "Benchmark Data for Evaluating Visualization and Analysis Techniques for Eye Tracking for Video Stimuli". In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2014, pp. 54–60 (on pages 5–7, 75, 89).
- [21] K. Kurzhals, B. Fisher, M. Burch, and D. Weiskopf. "Evaluating Visual Analytics with Eye Tracking". In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2014, pp. 61–69 (on pages 5–7, 40).

- [22] K. Kurzhals, B. Fisher, M. Burch, and D. Weiskopf. "Eye Tracking Evaluation of Visual Analytics". In: *Information Visualization* 15.4 (2016), pp. 340–358 (on pages 5–7, 40, 67, 80).
- [23] K. Kurzhals, M. Hlawatsch, M. Burch, and D. Weiskopf. "Fixation-Image Charts". In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2016, pp. 11–18 (on pages 5–8, 88, 94, 121, 133, 157, 161).
- [24] K. Kurzhals, M. Hlawatsch, F. Heimerl, M. Burch, and D. Weiskopf. "Gaze Stripes: Image-Based Visualization of Eye Tracking Data". In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (2016), pp. 1005–1014 (on pages 5–7, 82, 88, 94, 121, 157).
- [25] K. Kurzhals, M. John, F. Heimerl, P. Kuznecov, and D. Weiskopf. "Visual Movie Analytics". In: *IEEE Transactions on Multimedia* 18.11 (2016), pp. 2149–2160 (on pages 4, 6, 7, 11, 33, 38, 134).
- [26] K. Kurzhals, M. Burch, T. Blascheck, G. Andrienko, N. Andrienko, and D. Weiskopf. "A Task-Based View on the Visual Analysis of Eye-Tracking Data". In: *Eye Tracking and Visualization – Foundations, Techniques, and Applications (ETVIS 2015)*. Ed. by M. Burch, L. Chuang, B. Fisher, A. Schmidt, and D. Weiskopf. Springer, Cham Switzerland, 2017, pp. 3–22 (on pages 5–7, 75).
- [27] K. Kurzhals, E. Çetinkaya, Y. Hu, W. Wang, and D. Weiskopf. "Close to the Action: Eye-Tracking Evaluation of Speaker-Following Subtitles". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2017, pp. 6559–6568 (on pages 5–7, 40, 64).
- [28] K. Kurzhals, M. Stoll, A. Bruhn, and D. Weiskopf. "FlowBrush: Optical Flow Art". In: *Proceedings of the Symposium on Computational Aesthetics*. 2017, 1:1–1:9 (on pages 4, 6–8, 11, 15).
- [29] K. Kurzhals, M. Hlawatsch, C. Seeger, and D. Weiskopf. "Visual Analytics for Mobile Eye Tracking". In: *IEEE Transactions on Visualization and Computer Graphics* 23.1 (2017), pp. 301–310 (on pages 6, 7, 89, 146).
- [30] R. Netzel, B. Ohlhausen, K. Kurzhals, R. Woods, M. Burch, and D. Weiskopf. "User Performance and Reading Strategies for Metro Maps: An Eye Tracking Study". In: *Spatial Cognition & Computation* 17.1–2 (2017), pp. 39–64 (on pages 6, 112).
- [31] P. Tanisaro, J. Schöning, K. Kurzhals, G. Heidemann, and D. Weiskopf. "Visual Analytics for Video Applications". In: *it-Information Technology* 57.1 (2015), pp. 30–36 (on page 6).

Bibliography

- [32] A. Ab Aziz. “Repertory Grid Technique: A Pragmatic Approach to Evaluating User Experience in Visualisation Navigation”. PhD thesis. University of Southampton, 2016 (on page 48).
- [33] W. Aigner, S. Miksch, H. Schumann, and C. Tominski. *Visualization of Time-oriented Data*. Springer, London UK, 2011 (on page 113).
- [34] H. Alers, J. A. Redi, and I. Heynderickx. “Examining the Effect of Task on Viewing Behavior in Videos Using Saliency Maps”. In: *Proceedings of SPIE 8291, Human Vision and Electronic Imaging XVII*. 2012, pp. 82910X1–82910X8 (on page 90).
- [35] H. Alt and M. Godau. “Measuring the Resemblance of Polygonal Curves”. In: *Proceedings of the Eighth Annual Symposium on Computational Geometry*. 1992, pp. 102–109 (on pages 62, 137).
- [36] E. W. Anderson. “Evaluating Visualization Using Cognitive Measures”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2012, 5:1–5:4 (on pages 59, 61).
- [37] J. R. Anderson, D. Bothell, and S. Douglass. “Eye Movements Do Not Reflect Retrieval Processes: Limits of the Eye-Mind Hypothesis”. In: *Psychological Science* 15.4 (2004), pp. 225–231 (on page 56).
- [38] N. C. Anderson, F. Anderson, A. Kingstone, and W. F. Bischof. “A Comparison of Scanpath Comparison Methods”. In: *Behavior Research Methods* 47.4 (2015), pp. 1377–1392 (on page 62).
- [39] P. André, M. L. Wilson, A. Russell, D. A. Smith, A. Owens, and m. c. schraefel. “Continuum: Designing Timelines for Hierarchies, Relationships and Scale”. In: *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology*. 2007, pp. 101–110 (on page 106).
- [40] G. Andrienko, N. Andrienko, M. Burch, and D. Weiskopf. “Visual Analytics Methodology for Eye Movement Studies”. In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (2012), pp. 2889–2898 (on pages 72, 80, 106, 122).
- [41] N. Andrienko, G. Andrienko, and P. Gatalsky. “Exploratory Spatio-Temporal Visualization: An Analytical Review”. In: *Journal of Visual Languages & Computing* 14.6 (2003), pp. 503–541 (on page 36).
- [42] B. Bach, P. Dragicevic, D. Archambault, C. Hurter, and S. Carpendale. “A Descriptive Framework for Temporal Data Visualizations Based on Generalized Space-Time Cubes”. In: *Computer Graphics Forum* 36.6 (2017), pp. 36–61 (on page 22).

- [43] W. Bailer and G. Thallinger. “A Framework for Multimedia Content Abstraction and Its Application to Rushes Exploration”. In: *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*. 2007, pp. 146–153 (on page 158).
- [44] M. Banerjee, M. Capozzoli, L. McSweeney, and D. Sinha. “Beyond Kappa: A Review of Interrater Agreement Measures”. In: *Canadian Journal of Statistics* 27.1 (1999), pp. 3–23 (on page 157).
- [45] D. Baum. “Introducing Aesthetics to Software Visualization”. In: *Proceedings of the 23rd International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG)*. 2015, pp. 65–73 (on page 48).
- [46] E. Bekele, Z. Zheng, A. Swanson, J. Crittendon, Z. Warren, and N. Sarkar. “Understanding How Adolescents with Autism Respond to Facial Expressions in Virtual Reality Environments”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.4 (2013), pp. 711–720 (on page 68).
- [47] D. J. Berndt and J. Clifford. “Using Dynamic Time Warping to Find Patterns in Time Series”. In: *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining (KDD)*. 1994, pp. 359–370 (on pages 62, 137).
- [48] P. Bertolino. “Sensarea: A General Public Video Editing Application”. In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. 2014, pp. 3429–3431 (on page 157).
- [49] F. N. Bezerra and E. Lima. “Low Cost Soccer Video Summaries Based on Visual Rhythm”. In: *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*. 2006, pp. 71–78 (on page 134).
- [50] F. N. Bezerra and N. J. Leite. “Using String Matching to Detect Video Transitions”. In: *Pattern Analysis and Applications* 10.1 (2007), pp. 45–54 (on page 134).
- [51] A. Bhattacharyya. “On a Measure of Divergence between Two Multinomial Populations”. In: *Sankhyā: The Indian Journal of Statistics* 7.1 (1946), pp. 401–406 (on pages 13, 132, 137, 163).
- [52] T. Blascheck and T. Ertl. “Towards Analyzing Eye Tracking Data for Evaluating Interactive Visualization Systems”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2014, pp. 70–77 (on page 72).
- [53] T. Blascheck, M. Raschke, and T. Ertl. “Circular Heat Map Transition Diagram”. In: *Proceedings of the Conference on Eye Tracking South Africa*. 2013, pp. 58–61 (on page 82).

- [54] T. Blascheck, M. John, S. Koch, L. Bruder, and T. Ertl. “Triangulating User Behavior Using Eye Movement, Interaction, and Think Aloud Data”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2016, pp. 175–182 (on page 70).
- [55] J. S. Boreczky and L. A. Rowe. “Comparison of Video Shot Boundary Detection Techniques”. In: *Journal of Electronic Imaging* 5.2 (1996), pp. 122–128 (on page 15).
- [56] J. Boreczky, A. Girgensohn, G. Golovchinsky, and S. Uchihashi. “An Interactive Comic Book Presentation for Exploring Video”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2000, pp. 185–192 (on page 26).
- [57] R. Borgo, M. Chen, B. Daubney, E. Grundy, G. Heidemann, B. Höferlin, M. Höferlin, H. Leitte, D. Weiskopf, and X. Xie. “State of the Art Report on Video-Based Graphics and Video Visualization”. In: *Computer Graphics Forum* 31.8 (2012), pp. 2450–2477 (on pages 21, 29).
- [58] R. P. Botchen, S. Bachthaler, F. Schick, M. Chen, G. Mori, D. Weiskopf, and T. Ertl. “Action-Based Multifield Video Visualization”. In: *IEEE Transactions on Visualization and Computer Graphics* 14.4 (2008), pp. 885–899 (on page 97).
- [59] A. C. Bovik, ed. *Handbook of Image and Video Processing*. Elsevier Academic Press, Burlington MA, 2010 (on page 11).
- [60] M. Brehmer and T. Munzner. “A Multi-Level Typology of Abstract Visualization Tasks”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2376–2385 (on page 84).
- [61] M. Brehmer, S. Carpendale, B. Lee, and M. Tory. “Pre-Design Empiricism for Information Visualization: Scenarios, Methods, and Challenges”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2014, pp. 147–151 (on page 40).
- [62] G. Brône, B. Oben, and T. Goedemé. “Towards a More Effective Method for Analyzing Mobile Eye-Tracking Data: Integrating Gaze Data with Object Recognition Algorithms”. In: *Proceedings of the 1st International Workshop on Pervasive Eye Tracking and Mobile Eye-Based Interaction*. 2011, pp. 53–56 (on page 157).
- [63] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. “High Accuracy Optical Flow Estimation Based on a Theory for Warping”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2004, pp. 25–36 (on page 13).
- [64] A. Bulling and H. Gellersen. “Toward Mobile Eye-Based Human-Computer Interaction”. In: *IEEE Pervasive Computing* 9.4 (2010), pp. 8–12 (on page 146).

- [65] M. Burch, G. Andrienko, N. Andrienko, M. Höferlin, M. Raschke, and D. Weiskopf. “Visual Task Solution Strategies in Tree Diagrams”. In: *Proceedings of the IEEE Pacific Visualization Symposium (PacificVis)*. 2013, pp. 169–176 (on pages 68, 81).
- [66] M. Burch, A. Kull, and D. Weiskopf. “AOI Rivers for Visualizing Dynamic Eye Gaze Frequencies”. In: *Computer Graphics Forum* 32.3 (2013), pp. 281–290 (on page 113).
- [67] M. Burch, N. Konevtsova, J. Heinrich, M. Höferlin, and D. Weiskopf. “Evaluation of Traditional, Orthogonal, and Radial Tree Diagrams by an Eye Tracking Study”. In: *IEEE Transactions on Visualization and Computer Graphics* 17.12 (2011), pp. 2440–2448 (on pages 68, 90).
- [68] M. D. Byrne, J. R. Anderson, S. Douglass, and M. Matessa. “Eye Tracking the Visual Search of Click-Down Menus”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1999, pp. 402–409 (on page 70).
- [69] S. K. Card and J. Mackinlay. “The Structure of the Information Visualization Design Space”. In: *Proceedings of the IEEE Symposium on Information Visualization*. 1997, pp. 92–99 (on page 84).
- [70] S. K. Card, J. D. Mackinlay, and B. Shneiderman, eds. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann, San Francisco CA, 1999 (on pages 18, 19, 33).
- [71] R. Carmi and L. Itti. “Visual Causes Versus Correlates of Attentional Selection in Dynamic Scenes”. In: *Vision Research* 46.26 (2006), pp. 4333–4345 (on pages 90, 100).
- [72] S. Carpendale. “Evaluating Information Visualizations”. In: *Information Visualization* (2008), pp. 19–45 (on pages 41, 42, 53).
- [73] P. Cavanagh and G. Alvarez. “Tracking Multiple Targets with Multifocal Attention”. In: *Trends in Cognitive Sciences* 9.7 (2005), pp. 349–354 (on page 92).
- [74] J. F. Cerveny and R. P. Cerveny. “Capturing Manager’s Mental Models Using Kelly’s Repertory Grid”. In: *Proceedings of the 25th Hawaii International Conference on System Sciences*. 1992, pp. 435–442 (on page 47).
- [75] S.-H. Cha. “Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions”. In: *International Journal of Mathematical Models and Methods in Applied Sciences* 1.4 (2007), pp. 300–307 (on page 13).
- [76] J.-Y. Chang, W.-F. Hu, M.-H. Cheng, and B.-S. Chang. “Digital Image Translational and Rotational Motion Stabilization Using Optical Flow Technique”. In: *IEEE Transactions on Consumer Electronics* 48.1 (2002), pp. 108–115 (on page 13).
- [77] K. Charmaz. *Constructing Grounded Theory: A Practical Guide Through Qualitative Analysis*. Sage, London UK, 2006 (on page 42).

- [78] B. W. Chen, J. C. Wang, and J. F. Wang. “A Novel Video Summarization Based on Mining the Story-Structure and Semantic Relations among Concept Entities”. In: *IEEE Transactions on Multimedia* 11.2 (2009), pp. 295–312 (on page 26).
- [79] H.-W. Chen, J.-H. Kuo, W.-T. Chu, and J.-L. Wu. “Action Movies Segmentation and Summarization Based on Tempo Analysis”. In: *Proceedings of the 6th ACM SIGMM International Workshop on Multimedia Information Retrieval*. 2004, pp. 251–258 (on page 28).
- [80] M. Chen, R. Botchen, R. Hashim, D. Weiskopf, T. Ertl, and I. Thornton. “Visual Signatures in Video Visualization”. In: *IEEE Transactions on Visualization and Computer Graphics* 12.5 (2006), pp. 1093–1100 (on page 97).
- [81] E. H. Chi. “A Taxonomy of Visualization Techniques Using the Data State Reference Model”. In: *Proceedings of the IEEE Symposium on Information Visualization*. 2000, pp. 69–75 (on page 84).
- [82] M.-C. Chi, C.-H. Yeh, and M.-J. Chen. “Robust Region-of-Interest Determination Based on User Attention Model through Visual Rhythm Analysis”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 19.7 (2009), pp. 1025–1038 (on page 134).
- [83] J. P. Chin, V. A. Diehl, and K. L. Norman. “Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1988, pp. 213–218 (on page 42).
- [84] N. A. Chinchor, J. J. Thomas, P. C. Wong, M. G. Christel, and W. Ribarsky. “Multimedia Analysis + Visual Analytics = Multimedia Analytics”. In: *IEEE Computer Graphics and Applications* 30.5 (2010), pp. 52–60 (on page 28).
- [85] M. Christel and D. Martin. “Information Visualization within a Digital Video Library”. In: *Journal of Intelligent Information Systems* 11.3 (1998), pp. 235–257 (on page 122).
- [86] D. Comaniciu and P. Meer. “Mean Shift: A Robust Approach toward Feature Space Analysis”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.5 (2002), pp. 603–619 (on page 100).
- [87] R. Cooper and J. Ängeslevä. “The ‘last’ Clock”. In: *ACM SIGGRAPH Emerging Technologies*. 2004, pp. 15–15 (on page 143).
- [88] J. Corbin and A. Strauss. *Basics of Qualitative Research*. SAGE Publishing, Thousand Oaks CA, 2014 (on page 47).
- [89] T. Cour, C. Jordan, E. Miltsakaki, and B. Taskar. “Movie/Script: Alignment and Parsing of Video and Text Transcription”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2008, pp. 158–171 (on page 28).

- [90] F. Courtemanche, E. Aïmeur, A. Dufresne, M. Najjar, and F. Mpondo. “Activity Recognition Using Eye-gaze Movements and Traditional Interactions”. In: *Interacting with Computers* 23.3 (2011), pp. 202–213 (on page 61).
- [91] T. D’Orazio and M. Leo. “A Review of Vision-Based Systems for Soccer Video Analysis”. In: *Pattern Recognition* 43.8 (2010), pp. 2911–2926 (on page 10).
- [92] G. Daniel and M. Chen. “Video Visualization”. In: *Proceedings of the 14th IEEE Visualization (VIS)*. 2003, pp. 409–416 (on pages 21, 22).
- [93] M. Del Fabro and L. Böszörményi. “State-of-the-Art and Future Challenges in Video Scene Detection: A Survey”. In: *Multimedia Systems* 19.5 (2013), pp. 427–454 (on page 28).
- [94] S. Dey and S. W. Lee. “From Requirements Elicitation to Variability Analysis Using Repertory Grid: A Cognitive Approach”. In: *Proceedings of the 23rd IEEE International Requirements Engineering Conference*. 2015, pp. 46–55 (on page 47).
- [95] W. J. Dixon and J. Massey Frank. *Introduction to Statistical Analysis*. McGraw-Hill Book Company Inc., New York, 1950 (on page 137).
- [96] M. Dorr, T. Martinetz, K. R. Gegenfurtner, and E. Barth. “Variability of Eye Movements When Viewing Dynamic Natural Scenes”. In: *Journal of Vision* 10.10 (2010), 28:1–28:17 (on page 90).
- [97] G. Draper, Y. Livnat, and R. Riesenfeld. “A Survey of Radial Methods for Information Visualization”. In: *IEEE Transactions on Visualization and Computer Graphics* 15.5 (2009), pp. 759–776 (on page 152).
- [98] G. Draper and R. Riesenfeld. “Who Votes for What? A Visual Query Language for Opinion Data”. In: *IEEE Transactions on Visualization and Computer Graphics* 14.6 (2008), pp. 1197–1204 (on page 152).
- [99] A. T. Duchowski. “A Breadth-First Survey of Eye-Tracking Applications”. In: *Behavior Research Methods, Instruments, and Computers* 34.4 (2002), pp. 455–470 (on pages 10, 56).
- [100] A. T. Duchowski and B. H. McCormick. “Gaze-Contingent Video Resolution Degradation”. In: *Proceedings of SPIE 3299, Human Vision and Electronic Imaging III*. 1998, pp. 318–329 (on page 97).
- [101] H. M. Edwards, S. McDonald, and S. M. Young. “The Repertory Grid Technique: Its Place in Empirical Software Engineering Research”. In: *Information and Software Technology* 51.4 (2009), pp. 785–798 (on page 47).
- [102] B. D. Ehret. “Learning Where to Look: Location Learning in Graphical User Interfaces”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2002, pp. 211–218 (on page 70).

- [103] S. Elling, L. Lentz, and M. de Jong. “Retrospective Think-aloud Method: Using Eye Movements As an Extra Cue for Participants’ Verbalizations”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2011, pp. 1161–1170 (on page 70).
- [104] M. Ellouze, N. Boujemaa, and A. M. Alimi. “Im (S)²: Interactive Movie Summarization System”. In: *Journal of Visual Communication and Image Representation* 21.4 (2010), pp. 283–294 (on page 28).
- [105] K. A. Ericsson and H. A. Simon. “Verbal Reports As Data”. In: *Psychological Review* 87.3 (1980), pp. 215–251 (on page 42).
- [106] G. Escamilla, A. L. Craddock, and I. Kawachi. “Women and Smoking in Hollywood Movies: A Content Analysis.” In: *American Journal of Public Health* 90.3 (2000), pp. 412–414 (on page 10).
- [107] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise”. In: *Proceedings of Second International Conference on Knowledge Discovery and Data Mining (KDD)*. 1996, pp. 226–231 (on page 100).
- [108] D. Fallman and J. Waterworth. “Dealing with User Experience and Affective Evaluation in HCI Design: A Repertory Grid Approach”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems Workshop Paper*. 2005, pp. 1–5 (on page 48).
- [109] D. Fallman and J. Waterworth. “Capturing User Experiences of Mobile Information Technology with the Repertory Grid Technique”. In: *Human Technology: An Interdisciplinary Journal on Humans in ICT Environments* 6.2 (2010), pp. 250–268 (on page 48).
- [110] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. “From Data Mining to Knowledge Discovery in Databases”. In: *AI Magazine* 17.3 (1996), pp. 37–54 (on page 25).
- [111] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. “The KDD Process for Extracting Useful Knowledge from Volumes of Data”. In: *Communications of the ACM* 39.11 (1996), pp. 27–34 (on page 80).
- [112] N. Ferdous and R. Jianu. “Using Pupil Size As an Indicator for Task Difficulty in Data Visualization”. In: *IEEE VIS Poster Program* (2014), pp. 9–14 (on page 70).
- [113] C. Fillmore. “Frame Semantics”. In: *Linguistics in the Morning Calm*. Ed. by The Linguistic Society of Korea. Hanshin Publishing Co., Seoul Korea, 1982, pp. 111–137 (on pages 31, 32).
- [114] A. Fouse, N. Weibel, E. Hutchins, and J. D. Hollan. “ChronoViz: A System for Supporting Navigation of Time-Coded Data”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2011, pp. 299–304 (on page 29).

- [115] F. Fransella, R. Bell, and D. Bannister. *A Manual for Repertory Grid Technique*. John Wiley & Sons, West Sussex UK, 2004 (on pages 45, 46, 49, 51, 52).
- [116] M. Fromm. *Introduction to the Repertory Grid Interview*. Waxmann Verlag, Münster Germany, 2004 (on pages 50, 52).
- [117] Y. Fu and X. Zhang. “Exploring the Discrepancies between Users’ and Designers’ Perception to Identify Users’ Real Needs”. In: *Proceedings of the 11th IEEE Conference on Industrial Electronics and Applications*. 2016, pp. 1374–1378 (on page 47).
- [118] Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z.-H. Zhou. “Multi-View Video Summarization”. In: *IEEE Transactions on Multimedia* 12.7 (2010), pp. 717–729 (on page 158).
- [119] M. Furini, F. Geraci, M. Montangero, and M. Pellegrini. “VISTO: Visual Storyboard for Web Video Browsing”. In: *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*. 2007, pp. 635–642 (on pages 26, 158).
- [120] M. Furini, F. Geraci, M. Montangero, and M. Pellegrini. “STIMO: STILL and MOving Video Storyboard for the Web Scenario”. In: *Multimedia Tools and Applications* 46.1 (2009), pp. 47–69 (on pages 26, 28).
- [121] H. L. Gantt. *Work, Wages, and Profits*. The Engineering Magazine Co., New York, 1913 (on page 175).
- [122] P. Gatalsky, N. Andrienko, and G. Andrienko. “Interactive Analysis of Event Data Using Space-Time Cube”. In: *Proceedings of the Eighth International Conference on Information Visualisation (IV)*. 2004, pp. 145–152 (on page 97).
- [123] D. van Gennip, E. van den Hoven, and P. Markopoulos. “The Phenomenology of Remembered Experience: A Repertoire for Design”. In: *Proceedings of the European Conference on Cognitive Ergonomics*. 2016, 11:1–11:8 (on page 48).
- [124] J. H. Goldberg and J. I. Helfman. “Comparing Information Graphics: A Critical Look at Eye Tracking”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2010, pp. 71–78 (on page 61).
- [125] J. H. Goldberg and J. I. Helfman. “Visual Scanpath Representation”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2010, pp. 203–210 (on page 80).
- [126] J. H. Goldberg and J. I. Helfman. “Eye Tracking for Visualization Evaluation: Reading Values on Linear Versus Radial Graphs”. In: *Information Visualization* 10.3 (2011), pp. 182–195 (on page 68).
- [127] M. Graham and J. Kennedy. “A Survey of Multiple Tree Visualisation”. In: *Information Visualization* 9.4 (2010), pp. 235–252 (on page 112).

- [128] A. Griffin and A. Robinson. “Comparing Color and Leader Line Highlighting Strategies in Coordinated View Geovisualizations”. In: *IEEE Transactions on Visualization and Computer Graphics* 21.3 (2015), pp. 339–349 (on page 68).
- [129] T. Grindinger, A. T. Duchowski, and M. W. Sawyer. “Group-Wise Similarity and Classification of Aggregate Scanpaths”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2010, pp. 101–104 (on pages 61, 82).
- [130] Z. Guan and E. Cutrell. “An Eye Tracking Study of the Effect of Target Rank on Web Search”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2007, pp. 417–420 (on page 70).
- [131] Z. Guan, S. Lee, E. Cuddihy, and J. Ramey. “The Validity of the Stimulated Retrospective Think-Aloud Method As Measured by Eye Tracking”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2006, pp. 1253–1262 (on page 70).
- [132] M. J. Haass, L. E. Matzen, K. M. Butler, and M. Armenta. “A New Method for Categorizing Scanpaths from Eye Tracking Data”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2016, pp. 35–38 (on page 139).
- [133] H. Hadizadeh, M. Enriquez, and I. Bajic. “Eye-Tracking Database for a Set of Standard Video Sequences”. In: *IEEE Transactions on Image Processing* 21.2 (2012), pp. 898–903 (on page 90).
- [134] J. Hagedorn, J. Hailpern, and K. G. Karahalios. “VCode and VData: Illustrating a New Framework for Supporting the Video Annotation Workflow”. In: *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI)*. 2008, pp. 317–321 (on page 29).
- [135] A. Hanjalic. “Shot-Boundary Detection: Unraveled and Resolved?” In: *IEEE Transactions on Circuits and Systems for Video Technology* 12.2 (2002), pp. 90–105 (on page 14).
- [136] R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*. Addison-Wesley Longman Publishing Co., Inc. Boston MA, 1992 (on page 11).
- [137] M. Harrower and C. Brewer. “ColorBrewer.org: An Online Tool for Selecting Colour Schemes for Maps”. In: *The Cartographic Journal* 40.1 (2003), pp. 27–37 (on page 108).
- [138] M. Hassenzahl and T. Trautmann. “Analysis of Web Sites with the Repertory Grid Technique”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2001, pp. 167–168 (on page 48).

- [139] M. Hassenzahl and R. Wessler. “Capturing Design Space from a User Perspective: The Repertory Grid Technique Revisited”. In: *International Journal of Human-Computer Interaction* 12.3-4 (2000), pp. 441–459 (on page 48).
- [140] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer, New York, 2009 (on page 111).
- [141] I. Herman, G. Melancon, and M. Marshall. “Graph Visualization and Navigation in Information Visualization: A Survey”. In: *IEEE Transactions on Visualization and Computer Graphics* 6.1 (2000), pp. 24–43 (on page 112).
- [142] S. Hillaire, A. Lecuyer, T. Regia-Corte, R. Cozot, J. Royan, and G. Breton. “Design and Application of Real-time Visual Attention Model for the Exploration of 3D Virtual Environments”. In: *IEEE Transactions on Visualization and Computer Graphics* 18.3 (2012), pp. 356–368 (on page 70).
- [143] J. G. Hixon and W. B. Swann. “When Does Introspection Bear Fruit? Self-Reflection, Self-Insight, and Interpersonal Choices.” In: *Journal of Personality and Social Psychology* 64.1 (1993), pp. 35–43 (on page 147).
- [144] B. Höferlin, M. Höferlin, G. Heidemann, and D. Weiskopf. “Scalable Video Visual Analytics”. In: *Information Visualization* 14.1 (2015), pp. 10–26 (on page 25).
- [145] M. Höferlin, E. Grundy, R. Borgo, D. Weiskopf, M. Chen, I. W. Griffiths, and W. Griffiths. “Video Visualization for Snooker Skill Training”. In: *Proceedings of the 12th Eurographics/IEEE - VGTC Conference on Visualization (EuroVis)*. 2010, pp. 1053–1062 (on page 22).
- [146] M. Höferlin, B. Höferlin, G. Heidemann, and D. Weiskopf. “Interactive Schematic Summaries for Faceted Exploration of Surveillance Video”. In: *IEEE Transactions on Multimedia* 15.4 (2013), pp. 908–920 (on page 22).
- [147] T. Hogan, U. Hinrichs, and E. Hornecker. “The Elicitation Interview Technique: Capturing People’s Experiences of Data Representations”. In: *IEEE Transactions on Visualization and Computer Graphics* 22.12 (2016), pp. 2579–2593 (on pages 47, 49).
- [148] T. Hogan and E. Hornecker. “Blending the Repertory Grid Technique with Focus Groups to Reveal Rich Design Relevant Insight”. In: *Proceedings of the 6th International Conference on Designing Pleasurable Products and Interfaces*. 2013, pp. 116–125 (on pages 47, 48).
- [149] K. Holmqvist, M. Nyström, R. Andersson, R. Dewhurst, H. Jarodzka, and J. Van de Weijer. *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford University Press, Oxford UK, 2011 (on pages 58, 63, 82, 92, 109).
- [150] D. Holten and J. J. van Wijk. “Visual Comparison of Hierarchically Organized Data”. In: *Computer Graphics Forum* 27.3 (2008), pp. 759–766 (on page 112).

- [151] A. J. Hornof and T. Halverson. “Cognitive Strategies and Eye Movements for Searching Hierarchical Computer Displays”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2003, pp. 249–256 (on page 70).
- [152] W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank. “A Survey on Visual Content-Based Video Indexing and Retrieval”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 41.6 (2011), pp. 797–819 (on page 28).
- [153] Y. Hu, J. Kautz, Y. Yu, and W. Wang. “Speaker-Following Video Subtitles”. In: *ACM Transactions on Multimedia Computing, Communications, and Applications* 11.2 (2015), 32:1–32:17 (on page 64).
- [154] D. Huang, M. Tory, B. Aseniero, L. Bartram, S. Bateman, S. Carpendale, A. Tang, and R. Woodbury. “Personal Visualization and Personal Visual Analytics”. In: *IEEE Transactions on Visualization and Computer Graphics* 21.3 (2014), pp. 420–433 (on page 148).
- [155] J. Huang, R. W. White, and S. Dumais. “No Clicks, No Problem: Using Cursor Movements to Understand and Improve Search”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2011, pp. 1225–1234 (on page 70).
- [156] W. Huang. “Using Eye Tracking to Investigate Graph Layout Effects”. In: *Proceedings of the Asia-Pacific Symposium on Visualization (APVis)*. 2007, pp. 97–100 (on page 67).
- [157] W. Huang and P. Eades. “How People Read Graphs”. In: *Proceedings of the Asia-Pacific Symposium on Visualization (APVis)*. 2005, pp. 51–58 (on page 68).
- [158] J. S. Hunter. “The Exponentially Weighted Moving Average.” In: *Journal of Quality Technology* 18.4 (1986), pp. 203–210 (on page 100).
- [159] S. T. Iqbal, P. D. Adamczyk, X. S. Zheng, and B. P. Bailey. “Towards an Index of Opportunity: Understanding Changes in Mental Workload during Task Execution”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2005, pp. 311–320 (on page 70).
- [160] P. Isenberg, T. Zuk, C. Collins, and S. Carpendale. “Grounded Evaluation of Information Visualizations”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2008, 6:1–6:8 (on page 55).
- [161] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, and T. Möller. “A Systematic Review on the Practice of Evaluating Visualization”. In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2818–2827 (on page 40).

- [162] Y. Ishiguro and J. Rekimoto. “Gazecloud: A Thumbnail Extraction Method Using Gaze Log Data for Video Life-Log”. In: *Proceedings of the 16th International Symposium on Wearable Computers*. 2012, pp. 72–75 (on page 122).
- [163] H. Jänicke, R. Borgo, J. S. D. Mason, and M. Chen. “Soundriver: Semantically-rich Sound Illustration”. In: *Computer Graphics Forum* 29.2 (2010), pp. 357–366 (on pages 22, 28).
- [164] H. Jarodzka, K. Scheiter, P. Gerjets, and T. Van Gog. “In the Eyes of the Beholder: How Experts and Novices Interpret Dynamic Stimuli”. In: *Learning and Instruction* 20.2 (2010), pp. 146–154 (on page 62).
- [165] R. S. Jasinschi, N. Dimitrova, T. McGee, L. Agnihotri, J. Zimmerman, and D. Li. “Integrated Multimedia Processing for Topic Segmentation and Classification”. In: *Proceedings of the International Conference on Image Processing*. 2001, pp. 366–369 (on page 151).
- [166] A. R. Jensenius. “Some Video Abstraction Techniques for Displaying Body Movement in Analysis and Performance”. In: *Leonardo* 46.1 (2013), pp. 53–60 (on page 134).
- [167] R. Jianu, A. Rusu, Y. Hu, and D. Taggart. “How to Display Group Information on Node-Link Diagrams: An Evaluation”. In: *IEEE Transactions on Visualization and Computer Graphics* 20.11 (2014), pp. 1530–1541 (on page 68).
- [168] F. Jurie and B. Triggs. “Creating Efficient Codebooks for Visual Recognition”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Vol. 1. 2005, pp. 604–610 (on pages 13, 163).
- [169] M. A. Just and P. A. Carpenter. “Using Eye Fixations to Study Reading Comprehension”. In: *New Methods in Reading Comprehension Research*. Ed. by D. Kieras and M. A. Just. Hillsdale, New Jersey, 1984, pp. 151–182 (on page 56).
- [170] E. Karapanos, J.-B. Martens, and M. Hassenzahl. “Accounting for Diversity in Subjective Judgments”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2009, pp. 639–648 (on page 52).
- [171] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon. “Visual Analytics: Definition, Process, and Challenges”. In: *Information Visualization - Human-Centered Issues and Perspectives*. Ed. by A. Kerren, J. T. Stasko, J.-D. Fekete, and C. North. Springer, Berlin Germany, 2008, pp. 154–175 (on pages 10, 25).
- [172] G. A. Kelly. *The Psychology of Personal Constructs: Theory and Personality*. Norton, New York, 1955 (on pages 44–46, 48).
- [173] H. Kim, J. Lee, J.-H. Yang, S. Sull, W. M. Kim, and S. M.-H. Song. “Visual Rhythm and Shot Verification”. In: *Multimedia Tools and Applications* 15.3 (2001), pp. 227–245 (on page 134).

- [174] S.-H. Kim, Z. Dong, H. Xian, B. Upatising, and J. S. Yi. “Does an Eye Tracker Tell the Truth about Visualizations? Findings While Investigating Visualizations for Decision Making”. In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (2012), pp. 2421–2430 (on pages 67, 82).
- [175] Y. Kim and A. Varshney. “Persuading Visual Attention through Geometry”. In: *IEEE Transactions on Visualization and Computer Graphics* 14.4 (2008), pp. 772–782 (on page 67).
- [176] M. Kipp. “ANVIL: A Universal Video Research Tool”. In: *Handbook of Corpus Phonology*. Ed. by J. Durand, U. Gut, and G. Kristofferson. Oxford University Press, Oxford UK, 2014, pp. 420–436 (on page 29).
- [177] J. Klingner, R. Kumar, and P. Hanrahan. “Measuring the Task-Evoked Pupillary Response with a Remote Eye Tracker”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2008, pp. 69–72 (on pages 59, 61).
- [178] T. Kobayashi, T. Toyamaya, F. Shafait, M. Iwamura, K. Kise, and A. Dengel. “Recognizing Words in Scenes with a Head-Mounted Eye-Tracker”. In: *Proceedings of the IAPR International Workshop on Document Analysis Systems (DAS)*. 2012, pp. 333–338 (on page 158).
- [179] J. B. Kruskal and J. M. Landwehr. “Icicle Plots: Better Displays for Hierarchical Clustering”. In: *The American Statistician* 37.2 (1983), pp. 162–168 (on pages 112, 114).
- [180] K. Kunze, M. Iwamura, K. Kise, S. Uchida, and S. Omachi. “Activity Recognition for the Mind: Toward a Cognitive Quantified Self”. In: *Computer* 46.10 (2013), pp. 105–108 (on page 147).
- [181] M. Kwak, K. Hornbæk, P. Markopoulos, and M. Bruns Alonso. “The Design Space of Shape-Changing Interfaces: A Repertory Grid Study”. In: *Proceedings of the Conference on Designing Interactive Systems*. 2014, pp. 181–190 (on page 48).
- [182] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale. “Empirical Studies in Information Visualization: Seven Scenarios”. In: *IEEE Transactions on Visualization and Computer Graphics* 18.9 (2012), pp. 1520–1536 (on pages 39, 40, 43).
- [183] K. Lawonn, A. Baer, P. Saalfeld, and B. Preim. “Comparative Evaluation of Feature Line Techniques for Shape Depiction”. In: *Proceedings of Vision, Modeling and Visualization (VMV)*. 2014, pp. 31–38 (on page 47).
- [184] B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete, and N. Henry. “Task Taxonomy for Graph Visualization”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2006, pp. 1–5 (on page 84).

- [185] P. Legg, D. H. S. Chung, M. L. Parry, M. W. Jones, R. Long, I. W. Griffiths, and M. Chen. “MatchPad: Interactive Glyph-Based Visualization for Real-Time Sports Performance Analysis”. In: *Computer Graphics Forum* 31.3 (2012), pp. 1255–1264 (on page 22).
- [186] S. Lessing and L. Linge. “Iicap: A New Environment for Eye Tracking Data Analysis”. MA thesis. University of Lund, Sweden, 2002 (on page 107).
- [187] V. Levenshtein. “Binary Codes Capable of Correcting Deletions, Insertions, and Reversals”. In: *Soviet Physics-Doklady* 10.8 (1966), pp. 707–710 (on pages 63, 137).
- [188] X. Li, A. Çöltekin, and M.-J. Kraak. “Visual Exploration of Eye Movement Data Using the Space-Time-Cube”. In: *Proceedings of the 6th International Conference on Geographic Information Science*. 2010, pp. 295–309 (on page 97).
- [189] Y. Li and C.-C. J. Kuo. *Video Content Analysis Using Multimodal Information: For Movie Content Extraction, Indexing and Representation*. Springer, New York, 2013 (on page 28).
- [190] Y. Li, S.-H. Lee, C.-H. Yeh, and C.-C. J. Kuo. “Techniques for Movie Content Analysis and Skimming: Tutorial and Overview on Video Abstraction Techniques”. In: *IEEE Signal Processing Magazine* 23.2 (2006), pp. 79–89 (on pages 26, 28).
- [191] Y. Li, B. Merialdo, M. Rouvier, and G. Linares. “Static and Dynamic Video Summaries”. In: *Proceedings of the 19th ACM International Conference on Multimedia*. 2011, pp. 1573–1576 (on page 28).
- [192] D. J. Liebling and S. Preibusch. “Privacy Considerations for a Pervasive Eye Tracking World”. In: *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. 2014, pp. 1169–1177 (on page 150).
- [193] R. Lienhart, S. Pfeiffer, and W. Effelsberg. “The MoCA Workbench: Support for Creativity in Movie Content Analysis”. In: *Proceedings of the 3rd IEEE International Conference on Multimedia Computing and Systems*. 1996, pp. 314–321 (on page 28).
- [194] R. Likert. “A Technique for the Measurement of Attitudes”. In: *Archives of Psychology* 22.140 (1932), pp. 1–55 (on page 42).
- [195] A. Liu and Z. Yang. “Watching, Thinking, Reacting: A Human-Centered Framework for Movie Content Analysis”. In: *International Journal of Digital Content Technology and its Applications* 4.5 (2010), pp. 23–37 (on pages 26, 29).
- [196] A. Liu, S. Tang, Y. Zhang, Y. Song, J. Li, and Z. Yang. “A Hierarchical Framework for Movie Content Analysis: Let Computers Watch Films like Humans”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2008, pp. 1–8 (on page 29).

- [197] D. G. Lowe. “Object Recognition from Local Scale-Invariant Features”. In: *Proceedings of IEEE International Conference on Computer Vision (ICCV)*. Vol. 2. 1999, pp. 1150–1157 (on pages 13, 132, 163).
- [198] H. Luo, J. Fan, J. Yang, W. Ribarsky, and S. Satoh. “Exploring Large-scale Video News Via Interactive Visualization”. In: *Proceedings of the IEEE Symposium on Visual Analytics Science And Technology (VAST)*. 2006, pp. 75–82 (on page 158).
- [199] M. Möttus, E. Karapanos, D. Lamas, and G. Cockton. “Understanding Aesthetics of Interaction: A Repertory Grid Study”. In: *Proceedings of the 9th Nordic Conference on Human-Computer Interaction*. 2016, 120:1–120:6 (on page 48).
- [200] J. Mackinlay. “Automating the Design of Graphical Presentations of Relational Information”. In: *ACM Transactions on Graphics* 5.2 (1986), pp. 110–141 (on page 18).
- [201] P. Majoranta and A. Bulling. “Eye Tracking and Eye-Based Human-Computer Interaction”. In: *Advances in Physiological Computing*. Ed. by S. H. Fairclough and K. Gilleade. Springer, London UK, 2014, pp. 39–65 (on page 146).
- [202] L. Manovich. “Media Visualization: Visual Techniques for Exploring Large Media Collections”. In: *The International Encyclopedia of Media Studies Volume VI: Media Studies Futures*. Ed. by K. Gates. Blackwell Publishing Ltd., Hoboken NJ, 2013, pp. 95–116 (on page 122).
- [203] J. Martinho and T. Chambel. “ColorsInMotion: Interactive Visualization and Exploration of Video Spaces”. In: *Proceedings of the 13th International MindTrek Conference: Everyday Life in the Ubiquitous Era*. 2009, pp. 190–197 (on page 134).
- [204] S. Mathe and C. Sminchisescu. “Dynamic Eye Movement Datasets and Learnt Saliency Models for Visual Action Recognition”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2012, pp. 842–856 (on page 90).
- [205] R. Matthews. “Storks Deliver Babies ($p=0.008$)”. In: *Teaching Statistics* 22.2 (2000), pp. 36–38 (on page 41).
- [206] E. Mayr, G. Schreder, M. Smuc, and F. Windhager. “Looking at the Representations in Our Mind: Measuring Mental Models of Information Visualizations”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2016, pp. 96–103 (on page 47).
- [207] S. McKenna, D. Mazur, J. Agutter, and M. Meyer. “Design Activity Framework for Visualization Design”. In: *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), pp. 2191–2200 (on page 40).
- [208] C. McKnight. “The Personal Construction of Information Space”. In: *Journal of the American Society for Information Science* 51.8 (2000), pp. 730–733 (on page 47).

- [209] L. A. McNamara and N. Orlando-Gay. “Reading, Sorting, Marking, Shuffling: Mental Model Formation through Information Foraging”. In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2012 (on page 48).
- [210] L. Meng. “About Egocentric Geovisualisation”. In: *Proceedings of the 12th International Conference on Geoinformatics – Geospatial Information Research: Bridging the Pacific and Atlantic*. 2004, pp. 7–14 (on page 47).
- [211] M. Meyer, M. Sedlmair, P. S. Quinan, and T. Munzner. “The Nested Blocks and Guidelines Model”. In: *Information Visualization* 14.3 (2015), pp. 234–249 (on page 40).
- [212] P. Mital, T. Smith, R. Hill, and J. Henderson. “Clustering of Gaze during Dynamic Scene Viewing Is Predicted by Motion”. In: *Cognitive Computation* 3.1 (2011), pp. 5–24 (on pages 90, 97).
- [213] A. G. Money and H. Agius. “Video Summarisation: A Conceptual Framework and Survey of the State of the Art”. In: *Journal of Visual Communication and Image Representation* 19.2 (2008), pp. 121–143 (on page 28).
- [214] T. Munzner. “A Nested Model for Visualization Design and Validation”. In: *IEEE Transactions on Visualization and Computer Graphics* 15.6 (2009), pp. 921–928 (on pages 40, 54, 159).
- [215] A. Navab, K. Gillespie-Lynch, S. P. Johnson, M. Sigman, and T. Hutman. “Eye-Tracking As a Measure of Responsiveness to Joint Attention in Infants at Risk for Autism”. In: *Infancy* 17.4 (2012), pp. 416–431 (on page 151).
- [216] S. B. Needleman and C. D. Wunsch. “A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins”. In: *Journal of Molecular Biology* 48.3 (1970), pp. 443–453 (on pages 63, 137).
- [217] R. Netzel, M. Burch, and D. Weiskopf. “Comparative Eye Tracking Study on Node-Link Visualizations of Trajectories”. In: *IEEE Transactions on Visualization and Computer Graphics* 20.12 (2014), pp. 2221–2230 (on page 67).
- [218] R. Netzel, M. Burch, and D. Weiskopf. “Interactive Scanpath-Oriented Annotation of Fixations”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2016, pp. 183–187 (on page 158).
- [219] N. Niu and S. Easterbrook. “So, You Think You Know Others’ Goals? A Repertory Grid Study”. In: *IEEE Software* 24.2 (2007), pp. 53–61 (on page 55).
- [220] N. Niu and S. Easterbrook. “Discovering Aspects in Requirements with Repertory Grid”. In: *Proceedings of the International Workshop on Early Aspects at ICSE*. 2006, pp. 35–42 (on page 55).

- [221] C. North. “Toward Measuring Visualization Insight”. In: *Computer Graphics and Applications* 26.3 (2006), pp. 6–9 (on pages 61, 71).
- [222] M. Nyström and K. Holmqvist. “Effect of Compressed Offline Foveated Video on Viewing Behavior and Subjective Quality”. In: *ACM Transactions on Multimedia Computing, Communications, and Applications* 6.1 (2010), 4:1–4:14 (on page 126).
- [223] Y. Onoue, N. Kukimoto, N. Sakamoto, K. Misue, and K. Koyamada. “Layered Graph Drawing for Visualizing Evaluation Structures”. In: *IEEE Computer Graphics and Applications* 37.2 (2017), pp. 20–30 (on page 52).
- [224] Y. Onoue, N. Kukimoto, N. Sakamoto, and K. Koyamada. “E-Grid: A Visual Analytics System for Evaluation Structures”. In: *Journal of Visualization* 19.4 (2016), pp. 753–768 (on page 52).
- [225] A. A. Ozok. “Survey Design and Implementation in HCI”. In: *Human-Computer Interaction: Development Process*. Ed. by A. Sears and J. A. Jacko. Taylor & Francis, Boca Raton FL, 2009, pp. 254–270 (on page 42).
- [226] L. Paletta, K. Santner, G. Fritz, H. Mayer, and J. Schrammel. “3D Attention: Measurement of Visual Saliency Using Eye Tracking Glasses”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2013, pp. 199–204 (on page 145).
- [227] O. Palinko, A. L. Kun, A. Shyrovkov, and P. Heeman. “Estimating Cognitive Load Using Remote Eye Tracking in a Driving Simulator”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2010, pp. 141–144 (on page 70).
- [228] F. I. Parke. “Adaptation of Scan and Slit-scan Techniques to Computer Animation”. In: *ACM SIGGRAPH Computer Graphics* 14.3 (1980), pp. 178–181 (on page 134).
- [229] T. Partala, M. Jokiniemi, and V. Surakka. “Pupillary Responses to Emotionally Provocative Stimuli”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2000, pp. 123–129 (on page 70).
- [230] K. Pearson. “On Lines and Planes of Closest Fit to Systems of Points in Space”. In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (1901), pp. 559–572 (on page 52).
- [231] R. J. Peters and L. Itti. “Computational Mechanisms for Gaze Direction in Interactive Visual Environments”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2006, pp. 27–32 (on page 70).
- [232] T. Pfeiffer and P. Renner. “EyeSee3D: A Low-Cost Approach for Analyzing Mobile 3D Eye Tracking Data Using Computer Vision and Augmented Reality Technology”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2014, pp. 369–376 (on pages 58, 84, 145).

- [233] T. Pfeiffer, P. Renner, and N. Pfeiffer-Leßmann. “EyeSee3D 2.0: Model-Based Real-time Analysis of Mobile Eye-Tracking in Static and Dynamic Three-Dimensional Scenes”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2016, pp. 189–196 (on pages 58, 84, 157).
- [234] P. Pirolli and S. Card. “The Sensemaking Process and Leverage Points for Analyst Technology As Identified through Cognitive Task Analysis”. In: *Proceedings of the International Conference on Intelligence Analysis*. 2005, pp. 1–6 (on pages 19, 20, 76).
- [235] D. Ponceleon and A. Dieberger. “Hierarchical Brushing in a Collection of Video Data”. In: *Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS)*. 2001, pp. 1–8 (on page 29).
- [236] D. F. Pontillo, T. B. Kinsman, and J. B. Pelz. “SemantiCode: Using Content Similarity and Database-Driven Matching to Code Wearable Eyetracker Gaze Data”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2010, pp. 267–270 (on pages 122, 158).
- [237] A. Poole and L. Ball. “Eye Tracking in HCI and Usability Research”. In: *Encyclopedia of Human Computer Interaction*. Ed. by C. Ghaoui. Idea Group, Hershey PA, 2006, pp. 211–219 (on page 60).
- [238] K.-J. Räihä, A. Aula, P. Majoranta, H. Rantala, and K. Koivunen. “Static Visualization of Temporal Eye-Tracking Data”. In: *Human-Computer Interaction - INTERACT 2005*. Ed. by M. F. Costabile and F. Paternò. Vol. 3585. Springer, Berlin, 2005, pp. 946–949 (on page 82).
- [239] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. “You Only Look Once: Unified, Real-time Object Detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 779–788 (on page 10).
- [240] E. M. Reingold, N. Charness, M. Pomplun, and D. M. Stampe. “Visual Span in Expert Chess Players: Evidence from Eye Movements”. In: *Psychological Science* 12.1 (2001), pp. 48–55 (on page 68).
- [241] R. A. Rensink. “When Good Observers Go Bad: Change Blindness, Inattentional Blindness, and Visual Experience”. In: *Psyche* 6.9 (2000), pp. 288–298 (on page 23).
- [242] D. C. Richardson and R. Dale. “Looking to Understand: The Coupling between Speakers’ and Listeners’ Eye Movements and Its Relationship to Discourse Comprehension”. In: *Cognitive Science* 29.6 (2005), pp. 1045–1060 (on pages 81, 107).
- [243] G. Ristovski, M. Hunter, B. Olk, and L. Linsen. “EyeC: Coordinated Views for Interactive Visual Exploration of Eye-Tracking Data”. In: *Proceedings of the 17th International Conference on Information Visualisation*. 2013 (on pages 106, 110).

- [244] J. Roberts. “State of the Art: Coordinated Multiple Views in Exploratory Visualization”. In: *Proceedings of the International Conference on Coordinated and Multiple Views in Exploratory Visualization*. 2007, pp. 61–71 (on page 101).
- [245] M. Romero, J. Summet, J. Stasko, and G. Abowd. “Viz-A-Vis: Toward Visualizing Video through Computer Vision”. In: *IEEE Transactions on Visualization and Computer Graphics* 14.6 (2008), pp. 1261–1268 (on page 94).
- [246] O. de Rooij, J. van Wijk, and M. Worrying. “MediaTable: Interactive Categorization of Multimedia Collections”. In: *IEEE Computer Graphics and Applications* 30.5 (2010), pp. 42–51 (on page 158).
- [247] O. de Rooij and M. Worrying. “Browsing Video along Multiple Threads”. In: *IEEE Transactions on Multimedia* 12.2 (2010), pp. 121–130 (on page 158).
- [248] R. Rosenholtz, Y. Li, J. Mansfield, and Z. Jin. “Feature Congestion: A Measure of Display Clutter”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2005, pp. 761–770 (on page 84).
- [249] D. M. Russell, M. J. Stefik, P. Pirolli, and S. K. Card. “The Cost Structure of Sensemaking”. In: *Proceedings of the INTERACT and CHI Conference on Human Factors in Computing Systems*. 1993, pp. 269–276 (on page 19).
- [250] M. Ryan and M. Lenos. *An Introduction to Film Analysis: Technique and Meaning in Narrative Film*. Bloomsbury Academic, London UK, 2012 (on page 26).
- [251] G. Salton and C. Buckley. “Term-Weighting Approaches in Automatic Text Retrieval”. In: *Information Processing & Management* 24.5 (1988), pp. 513–523 (on page 32).
- [252] D. D. Salvucci. “Inferring Intent in Eye-Based Interfaces: Tracing Eye Movements with Process Models”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1999, pp. 254–261 (on page 70).
- [253] D. D. Salvucci and J. H. Goldberg. “Identifying Fixations and Saccades in Eye-tracking Protocols”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2000, pp. 71–78 (on pages 58, 80, 97).
- [254] J. Sang and C. Xu. “Character-Based Movie Summarization”. In: *Proceedings of the 18th ACM International Conference on Multimedia*. 2010, pp. 855–858 (on page 28).
- [255] A. Santella and D. DeCarlo. “Robust Clustering of Eye Movement Recordings for Quantification of Visual Interest”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2004, pp. 27–34 (on pages 97, 100).

- [256] Y. Sawahata, R. Khosla, K. Komine, N. Hiruma, T. Itou, S. Watanabe, Y. Suzuki, Y. Hara, and N. Issiki. “Determining Comprehension and Quality of TV Programs Using Eye-Gaze Tracking”. In: *Pattern Recognition* 41.5 (2008), pp. 1610–1626 (on page 97).
- [257] K. Schoeffmann, M. A. Hudelist, and J. Huber. “Video Interaction Tools: A Survey of Recent Work”. In: *ACM Computing Surveys* 48.1 (2015), 14:1–14:34 (on page 27).
- [258] K. Schoeffmann, M. Taschwer, and L. Boeszoermyeni. “The Video Explorer: A Tool for Navigation and Searching within a Single Video Based on Fast Content Analysis”. In: *Proceedings of the First Annual ACM Conference on Multimedia Systems*. 2010, pp. 247–258 (on pages 15, 29, 134).
- [259] J. Schöning, P. Faion, and G. Heidemann. “Pixel-Wise Ground Truth Annotation in Videos – an Semi-Automatic Approach for Pixel-Wise and Semantic Object Annotation”. In: *Proceedings of the International Conference on Pattern Recognition Applications and Methods (ICPRAM)*. 2016, pp. 690–697 (on page 157).
- [260] H.-J. Schulz, S. Hadlak, and H. Schumann. “The Design Space of Implicit Hierarchy Visualization: A Survey”. In: *IEEE Transactions on Visualization and Computer Graphics* 17.4 (2011), pp. 393–411 (on page 112).
- [261] 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 (2013), pp. 2366–2375 (on page 61).
- [262] H. Schütze, C. D. Manning, and P. Raghavan. *Introduction to Information Retrieval*. Cambridge University Press, New York, 2008 (on page 25).
- [263] M. Sedlmair, M. Meyer, and T. Munzner. “Design Study Methodology: Reflections from the Trenches and the Stacks”. In: *IEEE Transactions on Visualization and Computer Graphics* 18.12 (2012), pp. 2431–2440 (on page 40).
- [264] B. Shneiderman. “The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations”. In: *Proceedings of the IEEE Symposium on Visual Languages*. 1996, pp. 336–343 (on pages 84, 182).
- [265] H. Siirtola, T. Laivo, T. Heimonen, and K.-J. Raiha. “Visual Perception of Parallel Coordinate Visualizations”. In: *Proceedings of the 13th International Conference on Information Visualization (IV)*. 2009, pp. 3–9 (on page 67).
- [266] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. “Content-based Image Retrieval at the End of the Early Years”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.12 (2000), pp. 1349–1380 (on pages 9, 27, 164).

- [267] M. A. Smith and T. Kanade. “Video Skimming and Characterization through the Combination of Image and Language Understanding Techniques”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1997, pp. 775–781 (on pages 26, 28).
- [268] T. Smith and J. Henderson. “Attentional Synchrony in Static and Dynamic Scenes”. In: *Journal of Vision* 8.6 (2008), pp. 773–773 (on pages 92, 126).
- [269] R. R. Sokal and F. J. Rohlf. “The Comparison of Dendrograms by Objective Methods”. In: *Taxon* 11.2 (1962), pp. 33–40 (on page 138).
- [270] H. Song, J. Yun, B. Kim, and J. Seo. “GazeVis: Interactive 3D Gaze Visualization for Contiguous Cross-Sectional Medical Images”. In: *IEEE Transactions on Visualization and Computer Graphics* 20.5 (2014), pp. 726–739 (on page 68).
- [271] D. Sprague and M. Tory. “Exploring How and Why People Use Visualizations in Casual Contexts: Modeling User Goals and Regulated Motivations”. In: *Information Visualization* 11.2 (2012), pp. 106–123 (on page 150).
- [272] J. Stasko and E. Zhang. “Focus+Context Display and Navigation Techniques for Enhancing Radial, Space-Filling Hierarchy Visualizations”. In: *Proceedings of the IEEE Symposium on Information Visualization*. 2000, pp. 57–65 (on page 112).
- [273] S. Stellmach, L. Nacke, and R. Dachsel. “Advanced Gaze Visualizations for Three-Dimensional Virtual Environments”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2010, pp. 109–112 (on pages 81, 107).
- [274] M. Stengel, P. Bauszat, M. Eisemann, E. Eisemann, and M. Magnor. “Temporal Video Filtering and Exposure Control for Perceptual Motion Blur”. In: *IEEE Transactions on Visualization and Computer Graphics* 21.5 (2014), pp. 663–671 (on page 67).
- [275] S. S. Stevens. *Psychophysics: Introduction to Its Perceptual, Neural and Social Prospects*. Wiley, New York, 1975 (on page 43).
- [276] N. Sundaram, T. Brox, and K. Keutzer. “Dense Point Trajectories by GPU-Accelerated Large Displacement Optical Flow”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2010, pp. 438–451 (on page 13).
- [277] C. Swindells, M. Tory, and R. Dreezer. “Comparing Parameter Manipulation with Mouse, Pen, and Slider User Interfaces”. In: *Computer Graphics Forum* 28.3 (2009), pp. 919–926 (on page 67).
- [278] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. “DeepFace: Closing the Gap to Human-Level Performance in Face Verification”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2014, pp. 1701–1708 (on page 9).

- [279] Y. Takahashi, N. Nitta, and N. Babaguchi. “Video Summarization for Large Sports Video Archives”. In: *Proceedings of the IEEE International Conference on Multimedia and Expo*. 2005, pp. 1170–1173 (on page 122).
- [280] F. B. Tan and M. G. Hunter. “The Repertory Grid Technique: A Method for the Study of Cognition in Information Systems”. In: *MIS Quarterly* 26.1 (2002), pp. 39–57 (on page 47).
- [281] Y.-P. Tan, S. Kulkarni, and P. Ramadge. “A Framework for Measuring Video Similarity and Its Application to Video Query by Example”. In: *Proceedings of the International Conference on Image Processing*. 1999, pp. 106–110 (on page 127).
- [282] A. Tang, S. Greenberg, and S. Fels. “Exploring Video Streams Using Slit-Tear Visualizations”. In: *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI)*. 2008, pp. 191–198 (on pages 22, 134).
- [283] A. Telea and D. Auber. “Code Flows: Visualizing Structural Evolution of Source Code”. In: *Computer Graphics Forum* 27.3 (2008), pp. 831–838 (on page 112).
- [284] D. R. Thomas. “A General Inductive Approach for Analyzing Qualitative Evaluation Data”. In: *American Journal of Evaluation* 27.2 (2006), pp. 237–246 (on page 51).
- [285] J. J. Thomas and K. A. Cook, eds. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Center, 2005 (on pages 25, 150).
- [286] N. Thota. “Repertory Grid: Investigating Personal Constructs of Novice Programmers”. In: *Proceedings of the 11th Koli Calling International Conference on Computing Education Research*. 2011, pp. 23–32 (on page 47).
- [287] D. Tofan, M. Galster, and P. Avgeriou. “Capturing Tacit Architectural Knowledge Using the Repertory Grid Technique (NIER Track)”. In: *Proceedings of the 33rd International Conference on Software Engineering (ICSE)*. 2011, pp. 916–919 (on page 47).
- [288] M. Tory and T. Möller. “Rethinking Visualization: A High-Level Taxonomy”. In: *Proceedings of the IEEE Symposium on Information Visualization*. 2004, pp. 151–158 (on page 84).
- [289] M. Tory and C. Swindells. “Comparing ExoVis, Orientation Icon, and In-Place 3D Visualization Techniques”. In: *Proceedings of Graphics Interface*. 2003, pp. 57–64 (on page 99).
- [290] M. Tory, M. S. Atkins, A. E. Kirkpatrick, M. Nicolaou, and G.-Z. Yang. “Eyegaze Analysis of Displays with Combined 2D and 3D Views”. In: *Proceedings of the IEEE Visualization Conference*. 2005, pp. 519–526 (on pages 42, 68).

- [291] T. Toyama, T. Kieninger, F. Shafait, and A. Dengel. “Gaze Guided Object Recognition Using a Head-Mounted Eye Tracker”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2012, pp. 91–98 (on page 157).
- [292] J. Trümper, J. Döllner, and A. Telea. “Multiscale Visual Comparison of Execution Traces”. In: *Proceedings of the IEEE 21st International Conference on Program Comprehension (ICPC)*. 2013, pp. 53–62 (on page 112).
- [293] B. T. Truong and S. Venkatesh. “Video Abstraction: A Systematic Review and Classification”. In: *ACM Transactions on Multimedia Computing, Communications, and Applications* 3.1 (2007), (3)1–(3)7 (on page 28).
- [294] H. Y. Tsang, M. Tory, and C. Swindells. “ESeeTrack – Visualizing Sequential Fixation Patterns”. In: *IEEE Transactions on Visualization and Computer Graphics* 16.6 (2010), pp. 953–962 (on pages 82, 112–114, 158).
- [295] P.-H. Tseng, R. Carmi, I. G. Cameron, D. P. Munoz, and L. Itti. “Quantifying Center Bias of Observers in Free Viewing of Dynamic Natural Scenes”. In: *Journal of Vision* 9.7 (2009), pp. 1–16 (on page 100).
- [296] J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, Boston MA, 1977 (on page 25).
- [297] U. Turdukulov, A. O. Calderon Romero, O. Huisman, and V. Retsios. “Visual Mining of Moving Flock Patterns in Large Spatio-Temporal Data Sets Using a Frequent Pattern Approach”. In: *International Journal of Geographical Information Science* 28.10 (2014), pp. 2013–2029 (on page 94).
- [298] F. B. Viégas and M. Wattenberg. “TIMELINES: Tag Clouds and the Case for Vernacular Visualization”. In: *interactions* 15.4 (2008), pp. 49–52 (on page 153).
- [299] H. D. Wactlar, T. Kanade, M. A. Smith, and S. M. Stevens. “Intelligent Access to Digital Video: Informedia Project”. In: *Computer* 29.5 (1996), pp. 46–52 (on page 28).
- [300] Y. Wang, Z. Liu, and J.-C. Huang. “Multimedia Content Analysis: Using Both Audio and Visual Clues”. In: *IEEE Signal Processing Magazine* 17.6 (2000), pp. 12–36 (on page 28).
- [301] M. O. Ward, G. Grinstein, and D. Keim. *Interactive Data Visualization: Foundations, Techniques, and Applications*. Taylor & Francis, Boca Raton FL, 2010 (on page 114).
- [302] M. Wattenberg and F. Viégas. “The Word Tree, an Interactive Visual Concordance”. In: *IEEE Transactions on Visualization and Computer Graphics* 14.6 (2008), pp. 1221–1228 (on page 112).

- [303] N. Weibel, A. Fouse, C. Emmenegger, S. Kimmich, and E. Hutchins. “Let’s Look at the Cockpit: Exploring Mobile Eye-Tracking for Observational Research on the Flight Deck”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications (ETRA)*. 2012, pp. 107–114 (on pages 81, 107).
- [304] D. Weiskopf. “Iterative Twofold Line Integral Convolution for Texture-based Vector Field Visualization”. In: *Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration*. Ed. by T. Möller, B. Hamann, and R. D. Russell. Springer Berlin Heidelberg, 2009, pp. 191–211 (on page 18).
- [305] D. Weiskopf and G. Erlebacher. “Overview of Flow Visualization”. In: *The Visualization Handbook*. Ed. by C. D. Hansen and C. R. Johnson. Elsevier, Amsterdam, 2005, pp. 261–278 (on page 94).
- [306] C. Y. Weng, W. T. Chu, and J. L. Wu. “RoleNet: Movie Analysis from the Perspective of Social Networks”. In: *IEEE Transactions on Multimedia* 11.2 (2009), pp. 256–271 (on page 28).
- [307] J. M. West, A. R. Haake, E. P. Rozanski, and K. S. Karn. “EyePatterns: Software for Identifying Patterns and Similarities across Fixation Sequences”. In: *Proceedings of the ACM Symposium on Eye Tracking Research and Applications*. 2006, pp. 149–154 (on pages 82, 112).
- [308] S. Winkler and R. Subramanian. “Overview of Eye Tracking Datasets”. In: *Proceedings of the 5th International Workshop on Quality of Multimedia Experience*. 2013, pp. 212–217 (on page 90).
- [309] K. Wongsuphasawat and J. Lin. “Using Visualizations to Monitor Changes and Harvest Insights from a Global-Scale Logging Infrastructure at Twitter”. In: *Proceedings of the IEEE Conference on Visual Analytics Science and Technology (VAST)*. 2014, pp. 113–122 (on page 112).
- [310] M. Worrying, P. Sajda, S. Santini, D. A. Shamma, A. F. Smeaton, and Q. Yang. “Where Is the User in Multimedia Retrieval?” In: *IEEE MultiMedia* 19.4 (2012), pp. 6–10 (on page 27).
- [311] L. Xu, J. Jia, and Y. Matsushita. “Motion Detail Preserving Optical Flow Estimation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.9 (2012), pp. 1744–1757 (on page 13).
- [312] A. L. Yarbus. *Eye Movements and Vision*. Ed. by L. A. Riggs. Plenum Press, New York, 1967 (on pages 81, 83).
- [313] J. S. Yi, Y.-a. Kang, J. T. Stasko, and J. A. Jacko. “Understanding and Characterizing Insights: How Do People Gain Insights Using Information Visualization?” In: *Proceedings of the Workshop Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV)*. 2008, 4:1–4:6 (on page 84).

- [314] A. Yoshitaka and T. Ichikawa. “A Survey on Content-Based Retrieval for Multimedia Databases”. In: *IEEE Transactions on Knowledge and Data Engineering* 11.1 (1999), pp. 81–93 (on page 28).
- [315] T. Yue, Y. Dai, and Y. Liu. “A Hue-Saturation Histogram Difference Method to Vehicle Detection”. In: *Proceedings of the International Conference on Multimedia Technology*. 2011, pp. 31–34 (on page 127).
- [316] L. Zelnik-Manor and P. Perona. “Self-Tuning Spectral Clustering”. In: *Proceedings of the 17th International Conference on Neural Information Processing Systems (NIPS)*. 2004, pp. 1601–1608 (on page 163).
- [317] Z.-J. Zha, H. Zhang, M. Wang, H. Luan, and T.-S. Chua. “Detecting Group Activities with Multi-Camera Context”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 23.5 (2013), pp. 856–869 (on page 9).