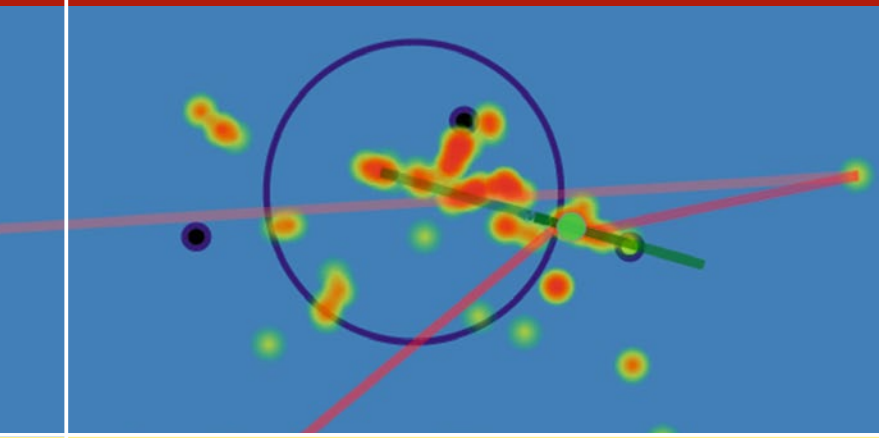
Mathematics and Visualization



Michael Burch · Lewis Chuang
Brian Fisher · Albrecht Schmidt
Daniel Weiskopf *Editors*

Eye Tracking and Visualization

Foundations, Techniques, and Applications. ETVIS 2015

ETVIS

Springer

Mathematics and Visualization

Series Editors
Hans-Christian Hege
David Hoffman
Christopher R. Johnson
Konrad Polthier
Martin Rumpf

More information about this series at http://www.springer.com/series/4562

Michael Burch • Lewis Chuang • Brian Fisher •
Albrecht Schmidt • Daniel Weiskopf
Editors

Eye Tracking and Visualization

Foundations, Techniques, and Applications. ETVIS 2015

Springer

Editors
Michael Burch
VISUS, University of Stuttgart
Stuttgart, Germany

Lewis Chuang
Max Planck Institute for Biological Cybernetics
Tübingen, Germany

Brian Fisher
SIAT, Simon Fraser University
Surrey
BC, Canada

Albrecht Schmidt
VIS, University of Stuttgart
Stuttgart, Germany

Daniel Weiskopf
VISUS, University of Stuttgart
Stuttgart, Germany

ISSN 1612-3786 ISSN 2197-666X (electronic)
Mathematics and Visualization
ISBN 978-3-319-47023-8 ISBN 978-3-319-47024-5 (eBook)
DOI 10.1007/978-3-319-47024-5

Library of Congress Control Number: 2017931067

Mathematics Subject Classification (2010): 62-07, 62-09, 62P15, 68-04, 68-06, 68U05, 68U20

© Springer International Publishing AG 2017
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Cover illustration from “Visualizing Eye Movements in Formal Cognitive Models” by J. Timothy Balint, Brad Reynolds, Leslie M. Blaha, and Tim Halverson with kind permission by the authors

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Technological advances in computer vision algorithms and sensor hardware have greatly reduced the implementation and financial costs of eye tracking, making this data acquisition method accessible to a large population of researchers. Thus, it is unsurprising to witness a significant increase in its use as a research tool in fields beyond the traditional domains of biological vision, psychology, and neuroscience, in particular, in visualization and human-computer interaction research. Recording the observer's gaze can reveal how dynamic graphical displays are visually accessed and it can show the visual information that is relevant to the observer at any given point in time. Nonetheless, standardized practices for technical implementations and data interpretation remain unresolved.

One of the key challenges lies in the analysis, interaction, and visualization of complex spatio-temporal datasets of gaze behavior, which is further complicated by complementary datasets such as semantic labels, user interactions and/or accompanying physiological sensor recordings. Ultimately, the research objective is to allow eye-tracking data to be effectively interpreted in terms of the observer's decision-making and cognitive processes. To achieve this, it is necessary to draw upon our current understanding of gaze-behavior across various and related fields, from vision and cognition to visualization.

Therefore, the analysis and visualization of such spatio-temporal gaze data—along with additionally attached data from the stimulus or further physiological sensor recordings—becomes a challenging factor in this emerging discipline. From the perspectives of human-computer interaction and cognitive science, this will be key to a better understanding of human behavior and the cognitive processes that underlie it, which would translate to genuine advances in user-centered computing systems. Taken together, this makes *eye tracking* an important field to be understood, be it in the sense of *data analysis* and *visualization*, *interaction*, or *user-based evaluation of visualization*.

ETVIS Workshop

To foster this growing field of eye tracking and visualization, we organized a workshop at the IEEE VIS Conference (http://ieeevis.org): *The First Workshop on Eye Tracking and Visualization (ETVIS)*. The workshop took place in Chicago, Illinois, USA, on October 25, 2015. Its web page is www.etvis.org.

The goal of this workshop was twofold. First, we intended to build a community of eye-tracking researchers within the visualization community. Since eye-tracking related research has been conducted in *information visualization*, *scientific visualization*, and *visual analytics*, this workshop served as a way of connecting between these subfields that are represented at IEEE VIS and share a common interest in eye tracking. Second, the workshop established connections to related fields, in particular, to human-computer interaction, cognitive science, and psychology, promoting the exchange of established practices and innovative use scenarios. We were pleased with the interdisciplinarity of the attendees, including researchers from human-computer interaction, psychology, cognitive science, and eye-tracking research to complement the traditional IEEE VIS audience. In particular, a large portion of the attendees indicated that they were first-time attendees of IEEE VIS and that ETVIS was the reason for attending IEEE VIS 2015. Thus, we saw that this novel initiative worked well in establishing common grounds across diverse disciplines and promoting interdisciplinary dialog.

The program of ETVIS 2015 consisted of two parts: The first part (morning sessions) was organized as the workshop itself. The program started with the keynote presentation by Kenneth Holmqvist (Lund University) on "Measures and Visualizations from Eye Movement Data." He presented a comprehensive overview of the many different measures that researchers use on eye-tracking data and showed several of the visualization techniques that have evolved over the last 15 years. A central question to take away from his talk is: when is visualization of eye gaze data helpful and when does it just provide the people who use it with a "wrong" sense of understanding? He highlighted this issue by using heat maps as an example, which are particularly popular in (web) usability. He discussed interesting examples of un-reflected uses of visualizations and the over-interpretation of heat maps. This provided a great starting point for further discussions.

The keynote was followed by oral presentations of the accepted workshop papers. In the afternoon, there was an accompanying meetup session to foster in-depth and open discussions between all attendees and allow for planning for the future of ETVIS. The success of ETVIS 2015 led us to organize a follow-up workshop at IEEE VIS 2016: ETVIS 2016 will take place October 23, 2016, in Baltimore, Maryland, USA.

More information about the ETVIS workshops can be found on the web page of the workshops: www.etvis.org.

Review Process

The chapters of this book were selected and revised during a review process with several stages. It started with a call for short papers (4 pages and 1 additional page for references) for ETVIS 2015. Submitted papers were reviewed by members of the international program committee (IPC) of ETVIS 2015 (see the list of IPC members on page ix). We had 3–4 reviews per paper. Decisions about acceptance were made by the ETVIS 2015 organizers, who are also the editors of this book. For submissions that had conflicts with some of the organizers, the review process was hidden from these organizers and decisions were made by the other organizers. We accepted 13 papers to the workshop. The workshop papers were not published as archival papers.

For the book, we invited the ETVIS authors to submit revised and extended versions of their papers. In addition, we solicited two additional chapter submissions in order to complement the contents and scope of the book: "A Task-Based View on the Visual Analysis of Eye Tracking Data" (by Kurzhals et al.) as a survey of visualization techniques for analyzing eye-tracking data, and "Unsupervised Clustering of EOG as a Viable Substitute for Optical Eye Tracking" (Flad et al.) as a complementary basis of data acquisition in eye-tracking experiments. All submitted chapters were reviewed by 2–3 expert reviewers (see page ix for the list of reviewers); for most book chapter submissions, there was (partial) reviewer continuity from the ETVIS review process. Decisions about acceptance were made by the editors of this book. Again, we took care that reviewing and decision-making were hidden from editors with conflicts. At the end, we accepted 12 extended ETVIS papers and the two additional chapters for the book.

Organization of this Book

The book is organized in two parts with a total of 14 chapters. The first part covers "Visualization, Visual Analytics, and User Interfaces". Its first chapter provides an overview of visualization approaches toward analyzing data that is acquired in the context of eye-tracking experiments: the chapter entitled "A Task-Based View on the Visual Analysis of Eye-Tracking Data" (by Kurzhals et al.). It includes references to several other chapters of this book to provide an overarching perspective on how visualization can be used to improve the analysis of eye-tracking data. The following two chapters cover visualization techniques that highlight the temporal aspect of attention and eye movement patterns: "Interactive Visualization for Understanding of Attention Patterns" (Nguyen et al.) and "The VERP Explorer: A Tool for Exploring Eye Movements of Visual-Cognitive Tasks Using Recurrence Plots" (Demiralp et al.). Then, Löwe et al. show gaze information in the context of immersive video stimuli ("Gaze Visualization for Immersive Video"). Blaha et al. describe a visual-motor analytics dashboard that supports the joint study of

eye movement and hand/finger movement dynamics ("Capturing You Watching You: Characterizing Visual-Motor Dynamics in Touchscreen Interactions"). The following chapter integrates the visualization of eye-tracking data in the context of the ACT-R cognitive architecture: "Visualizing Eye Movements in Formal Cognitive Models" (Balint et al.). Then, Beck et al. discuss "Word-Sized Eye-Tracking Visualizations" to augment transcribed recordings for protocol analysis with eye-tracking data. Finally, Tateosian et al. describe "GazeGIS: A Gaze-based Reading and Dynamic Geographic Information System".

The second part contains chapters that focus on "Data and Metrics". Flad et al. investigate electrooculography (EOG) as an alternative of acquiring information about eye movements in the chapter on "Unsupervised Clustering of EOG as a Viable Substitute for Optical Eye Tracking". This is followed by a chapter on data acquisition with optical monocular gaze tracking—with a focus on accuracy: "Accuracy of Monocular Gaze Tracking on 3D Geometry" (Wang et al.). Ma et al. apply tomography methods to obtain a description of 3D saliency ("3D Saliency from Eye Tracking with Tomography"). Schulz et al. address the issue of incorrect eye-tracking data and how such data can be cleaned with visualization support ("Visual Data Cleansing of Low-Level Eye-Tracking Data"). The last two chapters of this book discuss metrics for eye tracking: "Visualizing Dynamic Ambient/Focal Attention with Coefficient $\mathscr{K}$" (Duchowski and Krejtz) and "Eye Fixation Metrics for Large Scale Evaluation and Comparison of Information Visualizations" (Bylinskii et al.).

We hope that this book will stimulate further research in the interdisciplinary area of eye tracking and visualization, fostering the interaction between researchers from visualization and other disciplines.

Stuttgart, Germany — Michael Burch
Tübingen, Germany — Lewis Chuang
Surrey, Canada — Brian Fisher
Stuttgart, Germany — Albrecht Schmidt
Stuttgart, Germany — Daniel Weiskopf
July 2016

Acknowledgements

The initiative for the First Workshop on Eye Tracking and Visualization (ETVIS) was triggered by the Collaborative Research Center SFB/Transregio 161 ("Quantitative Methods for Visual Computing", www.sfbtrr161.de). We acknowledge the financial support by the German Research Foundation (DFG) for the SFB/Transregio 161 and the organization of ETVIS 2015.

Special thanks go to Kuno Kurzhals for his great help in organizing ETVIS 2015 and preparing this book compilation.

Reviewers and Members of the International Program Committee

We thank the following expert reviewers for their support of ETVIS 2015 as members of the international program committee and/or their help as reviewers of the extended book chapter submissions:

Natalia Andrienko	Kenneth Holmqvist
Florian Alt	Tony Huang
Stella Atkins	Peter Kiefer
Tanja Blascheck	Andrew Kun
Andreas Bulling	Kuno Kurzhals
Arzu Çöltekin	Lars Linsen
Raimund Dachselt	Aidong Lu
Loes van Dam	Radosław Mantiuk
Stephan Diehl	Rudolf Netzel
Andrew Duchowski	Sebastian Pannasch
Steve Franconeri	Thiess Pfeiffer
Wayne Gray	Mario Romero
John Henderson	Sophie Stellmach

Contents

Part I
Visualization, Visual Analytics, and User Interfaces

A Task-Based View on the Visual Analysis of Eye-Tracking Data

Kuno Kurzhals, Michael Burch, Tanja Blascheck, Gennady Andrienko, Natalia Andrienko, and Daniel Weiskopf

Abstract The visual analysis of eye movement data has become an emerging field of research leading to many new visualization techniques in recent years. These techniques provide insight beyond what is facilitated by traditional attention maps and gaze plots, providing important means to support statistical analysis and hypothesis building. There is no single "all-in-one" visualization to solve all possible analysis tasks. In fact, the appropriate choice of a visualization technique depends on the type of data and analysis task. We provide a taxonomy of analysis tasks that is derived from literature research of visualization techniques and embedded in our pipeline model of eye-tracking visualization. Our task taxonomy is linked to references to representative visualization techniques and, therefore, it is a basis for choosing appropriate methods of visual analysis. We also elaborate on how far statistical analysis with eye-tracking metrics can be enriched by suitable visualization and visual analytics techniques to improve the extraction of knowledge during the analysis process.

K. Kurzhals (✉) • M. Burch • D. Weiskopf
University of Stuttgart, Allmandring 19, 70569, Stuttgart, Germany
e-mail: Kuno.Kurzhals@visus.uni-stuttgart.de; Michael.Burch@visus.uni-stuttgart.de; Daniel.Weiskopf@visus.uni-stuttgart.de

T. Blascheck
University of Stuttgart, Universitätsstr. 38, 70569, Stuttgart, Germany
e-mail: Tanja.Blascheck@vis.uni-stuttgart.de

G. Andrienko
Fraunhofer Institute IAIS, Schloss Birlinghoven, 53757, Sankt Augustin, Germany

City University London, London EC1V OHB, United Kingdom
e-mail: gennady.andrienko@iais.fraunhofer.de

N. Andrienko
Fraunhofer Institute IAIS, Schloss Birlinghoven, 53757, Sankt Augustin, Germany
e-mail: natalia.andrienko@iais.fraunhofer.de

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_1

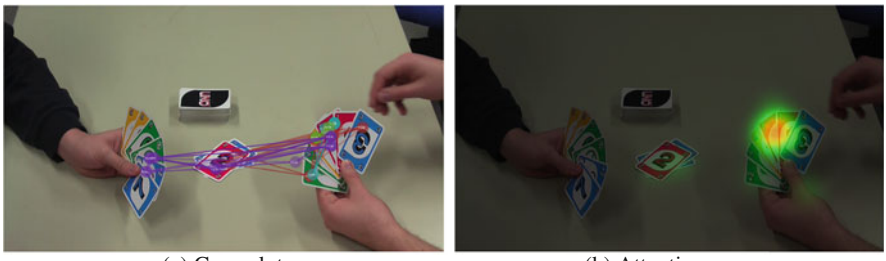
1 Introduction

The application of eye-tracking technology as a means of evaluating human behavior has been established in many different research fields [15]. Due to the interdisciplinary constellation of researchers, the specific analysis tasks may also differ between the fields. While one researcher might be interested in the physiological measures (e.g., eye movement speed [27]), another wants to know in what order specific areas of interest on a visual stimulus were investigated [8]. Despite the differences between the research fields, it is possible to derive a high-level task categorization from a data perspective. Since the structure of the recorded data is usually identical in all eye-tracking experiments, we can categorize the analysis tasks according to three main data dimensions and three elementary analysis operations.

Depending on the research question, a statistical analysis of established eye-tracking metrics [23] can be sufficient. However, the more complex the analysis task becomes, the more visual aid is usually required to interpret the data. Regarding the increasing amount of eye-tracking data recorded during experiments [2], it is reasonable to incorporate visual analytics techniques that combine automatic data processing with interactive visualization [1] into the analysis process.

As a starting point, the analysis of eye-tracking data is usually supported by some basic visualization techniques. For statistical measures, the application of statistical plots depicting the changes of a variable over time can already be helpful to interpret the data. In these cases, the visual stimulus is neglected. If the visual stimulus is important for the analysis, additional visualization techniques are usually included in the software suites of the major eye-tracking vendors.

For many years, gaze plots and attention maps (Fig. 1) were (and still are) the most popular visualizations that include information about the underlying visual stimulus. However, not all analysis tasks are facilitated by these techniques. For example, even though animated versions of the techniques in Fig. 1 exist, it is hard to interpret changes over time by simply replaying the animation [46]. Therefore, many new techniques have been developed over the last years to address this and



(a) Gaze plot (b) Attention map

Fig. 1 Typical eye-tracking data visualizations: The (**a**) gaze plot and the (**b**) visual attention map are the most common depictions of recorded gaze data

many other analysis tasks, summarized by Blascheck et al. [4]. Additionally, as a beneficial but also challenging aspect, apart from the pure eye movement data a wealth of additional data sources can be integrated into an experiment [2]. Such a collection of heterogeneous data sources often impairs a combined analysis by statistical means and makes a visual approach indispensable.

With that said, our goal is twofold: We define typical analysis tasks when visualization techniques for eye movement data come into play. Our high-level categorization is based on data dimensions directly focusing on recorded eye movement data but also on basic analysis operations. As a second goal, we discuss for each task category to which degree statistical and visual analysis can be applied to perform the given task, and present the suitable techniques. We base the list of examined visualization techniques on the collection provided in the state-of-the-art report by Blascheck et al. [4], which we consider fairly complete.

The overarching intention of this article is to support analysts working in the field of eye tracking to choose appropriate visualizations depending on their analysis task.

2 The Eye-Tracking Visualization Pipeline

We formulate the way from conducting an eye-tracking experiment to gaining insight in the form of a pipeline (Fig. 2) that is an extended version of the generic visualization pipeline [11, 21]. The acquired data consisting of eye movement data and complementary data sources is processed and optionally annotated before a visual mapping, creating the visualization, is performed. 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 [36], to confirm, reject, or build new hypotheses from where knowledge can be derived. Since the analysis task plays an

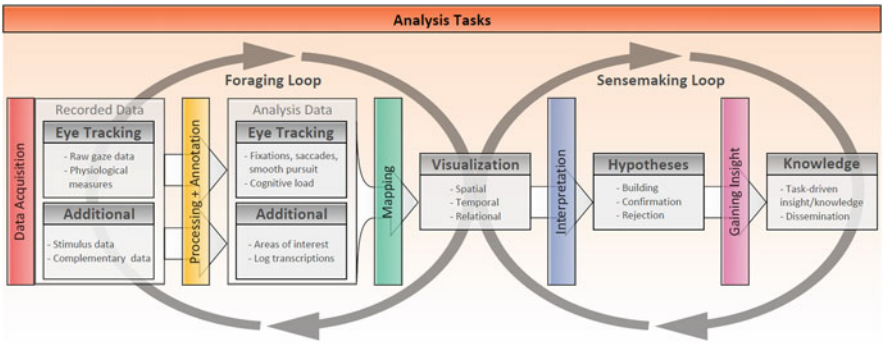


Fig. 2 Extended visualization pipeline for eye-tracking data: The recorded data passes multiple transformation steps before knowledge can be extracted. Each step from data acquisition, processing, mapping, interpretation, to gaining insight is influenced by the analysis task

important role in all steps of the pipeline, we first discuss the underlying data and how it is processed before we introduce our categorization of analysis tasks.

2.1 Data Acquisition

Eye movement data combines several data dimensions of spatio-temporal nature. We distinguish between dimensions directly stemming from the recording of 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 eye movement data. Typically, the displayed stimuli are an additional data source that can usually be included in the analysis process, since they are the foundation of most experiments anyway. Additional data sources provide complementary data such as verbal feedback, electroencephalography (EEG) data, and key press protocols.

The analysis task, or more precisely, the research question, typically 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 much more difficult to control the experiment parameters.

2.2 Processing and Annotation

From the time-varying sequence of raw gaze points, more data constructs can be derived in a processing step. We identified fixations, saccades, smooth pursuits, and scanpaths as the most important data constructs [23]. In this processing step, automatic data-mining algorithms can be applied to filter and aggregate the data. Clustering and classification are prominent processing steps: For example, raw gaze points can be clustered into fixations and labeled. As another example, the convex hull of a subset of gaze points can be extracted to automatically identify areas of interest (AOIs). In general, the annotation of AOIs plays an important role in this step.

From the visual content of a stimulus (e.g., a picture or a video), AOIs can be annotated, providing semantic interpretation of the stimulus. With this information, additional data such as transition sequences between AOIs can be derived. Therefore, analysts can either rely on automatic, data-driven approaches to detect AOIs, or define them manually. Basically, there are two approaches: either defining areas or objects by bounding regions on the stimulus and calculating hits with the gaze data, or labeling each fixation individually based on the investigated content. Especially for video sequences, this annotation is a time-consuming step that often takes more effort than the rest of the analysis process.

From the additional data sources, recorded protocols and log files can typically be derived. It should be noted that each additional data source requires a synchronization with the recorded eye movement data, which can be difficult considering different sampling rates and irregularly sampled data (e.g., think-aloud comments) [3]. The processed data can finally be used for the mapping 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, visualization and processing are tightly coupled in a foraging loop, where the analyst can identify relevant data artifacts through interaction with the visualization.

2.3 Mapping

The mapping step projects the analysis data to a visual representation. According to Blascheck et al. [4], the main categories of state-of-the-art visualization techniques for eye tracking are spatial, temporal, and relational data representations. Therefore, our 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 eye movement data also take into account the additional data sources for an enhanced visual design in order to explore the data. We think that this is actually noteworthy since those data sources may build meaningful input for sophisticated data analyses if they are combined with the traditional eye movement data.

As mentioned before, the analysis task plays the most important role for the choice of 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 certain analysis task.

2.4 Interpretation

For the interpretation of the visualization, we can distinguish between two strategies: Applying visualization to support statistical measures and 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 additionally support these calculations. 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 is particularly useful to analyze data from pilot studies. Building new hypotheses, the experiment design can be adjusted and appropriate metrics can be defined 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. It allows one to investigate the data in general before the analysis task is specified.

2.5 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 eye-tracking metrics and attention maps are displayed to help the reader better understand the statistical results.

3 Categorization of Analysis Tasks

The visualization pipeline for eye-tracking data (Fig. 2) 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 our 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. In this section, we present a categorization of analysis tasks that aims at helping with choosing appropriate visualizations. We discuss the main properties of the involved data constructs, typical measures for these questions, and propose visualizations that fit the tasks.

To provide a systematic overview of typical analysis tasks, we first derive the three independent data dimensions in eye-tracking data:

- **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 where 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 can be applied to display dependent data constructs (e.g., fixation durations). Since many visualization techniques may not be restricted to just one of these dimensions but may facilitate different combinations of them, we focus our subsections on techniques where the name-giving dimension can be considered as the main dimension for the visualization.

Additionally, we can derive general analytical operations that can be related to other taxonomies (e.g., the knowledge discovery in databases (KDD) process [17]):

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

This categorization is based on the survey by Blascheck et al. [4], the work of Andrienko et al. [1], and the work of Kurzhals et al. [29]. The authors provide an overview of current state-of-the art visualization and visual analytics approaches for the analysis of eye-tracking data. However, they did not include a discussion of the typical analysis tasks performed with the visualization and visual analytics techniques. The proposed metrics are derived from Holmqvist et al. [23].

3.1 Where? – Space-Based Tasks

Typical questions that consider 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 we define a saccade as a vector from one fixation to another, typical *where* questions can also be formulated for saccade directions. If AOIs are available, measures such as the average dwell time on each AOI can be calculated and represented by numbers or in a histogram.

If the stimulus content is important for the analysis, attention maps [7] and gaze plots are typically the first visualizations that come to mind. Attention maps scale well with the number of participants and recorded data points, but totally neglect the sequential order of points. With an appropriate color mapping and supportive statistical measures, an attention map can already be enough to answer many questions where participants looked at, if the investigated stimulus is static.

Space-based tasks for dynamic stimuli, such as videos and interactive user interfaces, require a visualization that takes the temporal dimension into account

considering also the changes of the stimulus over time. If AOIs are available, we refer to the next section, because in this case, *when* and *where* are tightly coupled. In an analysis step before the annotation of AOIs, there are two visualizations techniques that depict *where* participants looked at over time. Those are namely the space-time cube [32, 34] (Fig. 3a) and the gaze stripes [31] (Fig. 3b).

In a space-time cube, the spatial dimension of the stimulus is preserved, while the temporal data is included as a third dimension. Gaze points as well as scanpaths can be investigated over time. Common viewing behavior as well as outliers (Sect. 3.6) can be detected, but the stimulus is usually only available on demand, for example, by sliding a video plane through the space-time volume. Similar representations for one spatial and the temporal dimension are also possible (e.g., de Urabain et al. [13]). Gaze stripes preserve the information about the watched stimulus content by creating small thumbnails of the foveated region for each time step and placing them on a timeline. With this approach, individual participants can be compared. However, the spatial component is in this case implicitly coded by the image content, providing more of an answer to the question what was investigated.

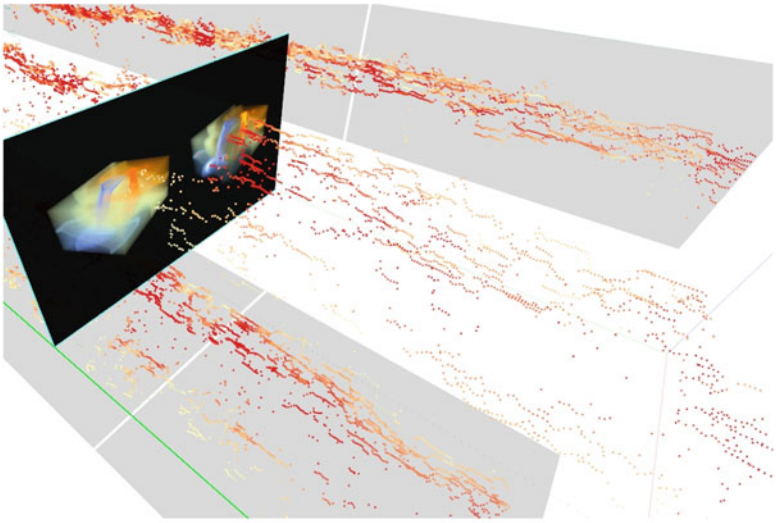
3.2 When? – Time-Based Tasks

Eye movement data has a spatio-temporal nature often demanding for a detailed analysis of changes in variables over time. Questions in this category typically have the focus on a certain event in the data (e.g., a fixation, smooth pursuit) 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 [41], for example, calculates when a fixation started and ended. For semantic interpretations, combining data dimensions to answer questions *when* was *what* investigated, the inclusion of AOIs is common. For statistical analysis, measures such as the "time to first hit" in an AOI can be calculated.

Without AOI information, the visual analysis of the temporal dimension is rather limited. Statistical plots of variables such as the x- and y-component [19], 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 us better understand when attention changes appeared over time.

Timeline visualizations are a good choice to answer questions related to this category. Figure 4 depicts an example where multiple timelines for different AOIs are stacked on top of each other [12, 30]. 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, or not [44, 47]. In Fig. 4, the data dimension *who* was included by displaying histograms inside the bars indicating how many participants looked at the AOI over time. In general, timeline representations depict an additional data dimension or construct, allowing one to combine the data relevant for the analysis with its temporal progress.

(a)



(b)

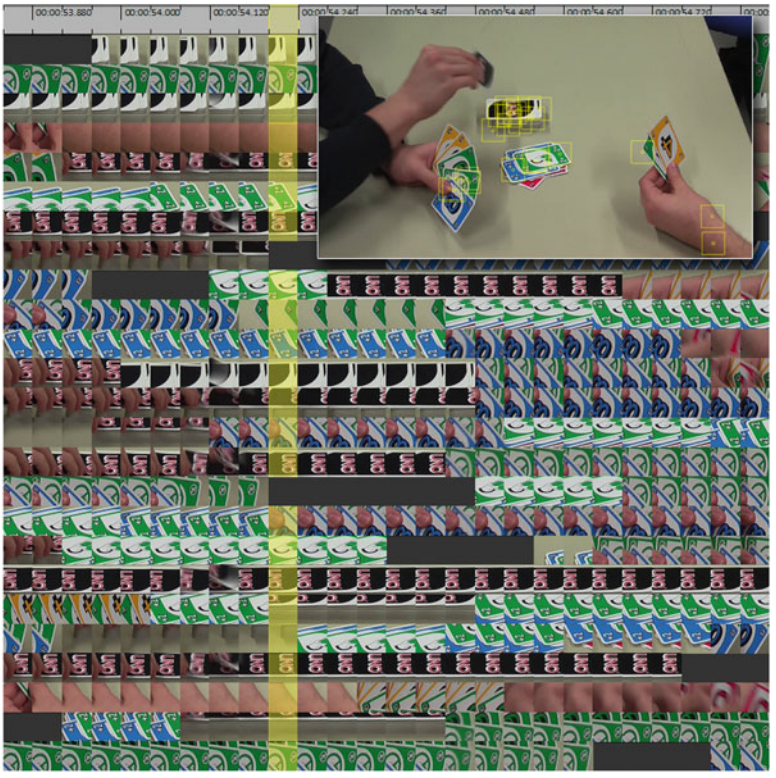


Fig. 3 Two visualization techniques to investigate where participants looked at over time in dynamic stimuli without the need of annotating AOIs. (**a**) Space-time cube. (**b**) Gaze stripes

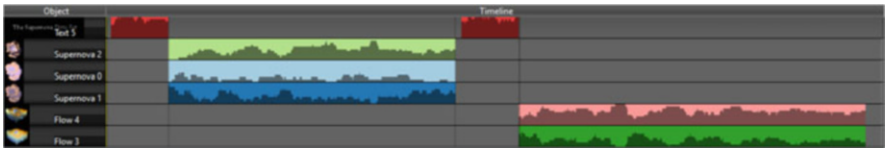


Fig. 4 Timeline visualization showing when an AOI appeared (colored bars) and how many participants looked at it over time (histograms inside the bars)

3.3 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 can provide insights into the visual task solution strategies applied by them (e.g., Burch et al. [8]). For a single participant, a traditional gaze plot is useful to interpret the scanpath, assuming that the recorded trajectory is not too long nor located in just a small stimulus subregion. Generally, most visualization techniques for multiple participants work fine also for an individual participant. For the comparison of multiple participants, gaze plots are less scalable, because of the massive overplotting that occurs when many participants' scanpaths are displayed in one representation.

To ease the comparison of scanpaths, specific metrics to identify similarities between participants can be applied, such as the Levenshtein or Needleman-Wunsch distance [16, 48]. Based on visited AOIs, a string is derived that can be compared by the mentioned similarity measures. As a consequence, scanpaths from many participants can be compared automatically. To interpret the comparison results, a visual representation of the scanpaths that supports the similarity measure can be helpful.

Similar to the concept in Fig. 4, a timeline for individual participants can be created, commonly known as scarf plot [30, 39]. The corresponding color of an AOI is assigned to each point in time it was visited. With a hierarchical agglomerative clustering on the similarity values, a dendrogram can display the similarities between participants. In Fig. 5, participant 4 and 7 are most similar because their sequences of visits to the green and the dark blue AOI shows the highest level of resemblance. Alternatively, one timeline per AOI can be kept and the scanpath can be plotted as a connected line over the timelines [37, 38].

The comparison of participants nowadays benefits from the automatic processing of scanpath similarities. Since the applied similarity measures can lead to different results, a visual interpretation is crucial to avoid misinterpretations.

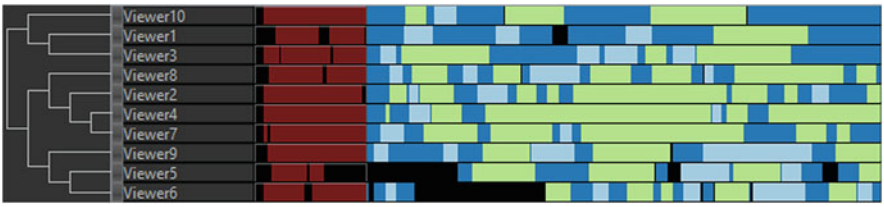


Fig. 5 Timelines for individual participants (scarf plots) depicting their scanpath based on the colors of visited AOIs (*right*). The dendrogram (*left*) displays the hierarchical clustering of the most similar scanpaths, measured by the Levenshtein distance

3.4 Compare

Section 3.3 introduced the comparison of participants based on scanpath similarities. 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 typically calculated by comparing distributions of a dependent variable. For example, fixation durations between different stimulus conditions can be compared with an ANOVA to find out whether a significant difference between the two distributions exists. However, inferential statistics can only provide the information that a difference exists. To interpret the difference between the conditions, a visual comparison is usually a good supplement to the statistical calculations.

Comparison tasks are typically supported by placing several of the visualized data instances next to each other in a side-by-side representation, sometimes denoted as small multiples visualization. Each data instance is visually encoded in the same visual metaphor to facilitate the comparison.

An example of such visual comparison can be found in a seminal eye-tracking experiment conducted by Yarbus [49], with participants investigating the painting "The unexpected visitor". To compare the viewing behavior for different tasks, the resulting eye movement patterns were depicted by rudimentary gaze plots, allowing an easy interpretation of how the task influenced the eye movements. This visualization strategy can be applied to many techniques, for example, to compare investigated stimulus content over time (Fig. 3b), different distributions of attention on AOIs [9, 12] (Fig. 4), and the comparison of participants [30, 38] (Fig. 5).

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 (e.g., by calculating differences) and then the result is visually encoded in a suitable visual metaphor. Although this technique has many benefits concerning a reduction of visual clutter and number of data items to be displayed, it comes with the drawback of deleting data commonalities that might be important to visually explore and understand the data instances to be compared.

An example of such a scenario is the calculation of differences between attention maps. Attention maps represent a distribution of attention and can be subtracted from each other, leaving higher differences between the values where the

distribution was different. The result can again be visualized as an attention map, showing hot spots in the difference regions.

3.5 Relate

In most analysis scenarios, not only a single dimension such as space, time, or participants is in the research focus. A combination of two, three, or even more dimensions and data constructs is included in the analysis to explore the data for correlations and relations between the data dimensions.

Correlations between data dimensions in eye-tracking research are often analyzed statistically. If multiple data dimensions are investigated simultaneously, we typically speak of multivariate data. In a statistical analysis, such multivariate correlations are often evaluated using Pearson's correlation coefficient. It tests how data constructs correlate with each other and how strong this relation is. However, without visual interpretation, correlation values can be hard to interpret. This can be overcome using visualization techniques for multivariate data. Typical examples are scatter plots, scatter plot matrices, or parallel coordinates [22, 24]. Scatter plots have been used in eye movement research for years. For example, Just and Carpenter [25] depicted different metrics such as number of switches, angular disparity, response latency, or duration to investigate cognitive processes. To our knowledge, parallel coordinates have not been used to analyze multiple eye movement metrics so far. However, they could give valuable insights into correlations amongst metrics.

Investigating relations between AOIs or participants is the second important aspect for analysis tasks in this category. The relationship amongst participants has already been discussed in Sect. 3.3. The relationship between AOIs is discussed in the following. Relations between AOIs are often investigated by transitions between them. They can show which AOIs have been looked at in what order. A standard statistical measure is the transition count. Transition matrices or Markov models can give valuable insight into search behavior of a participant [23]. The transition matrices can be extended by coding transition count with color [18], allowing one to detect extrema between AOI transitions efficiently (see Fig. 6a). Blascheck et al. [5] use such transition matrices with an attached AOI hierarchy to show clusters between different AOI groups. Similar to transition matrices, recurrence plots [14] (Fig. 6b) depict the return to fixations or AOIs and thus search behavior of a participant.

Another typical technique for showing relations between elements are graphs and trees. These visualization techniques can be extended to AOI transitions. A transition graph depicts AOIs, or meta information about AOIs (Fig. 6c) as nodes and transitions as links [6, 35]. The example depicted in Fig. 6c is the work of Nguyen et al., which is described in detail in their chapter [35] later in this book. Graphs can be used to represent which AOIs have been focused on and how often. Fig. 6d shows a transition graph where AOIs are depicted as color-coded circle segments. The color corresponds to the dwell time of an AOI. The transitions

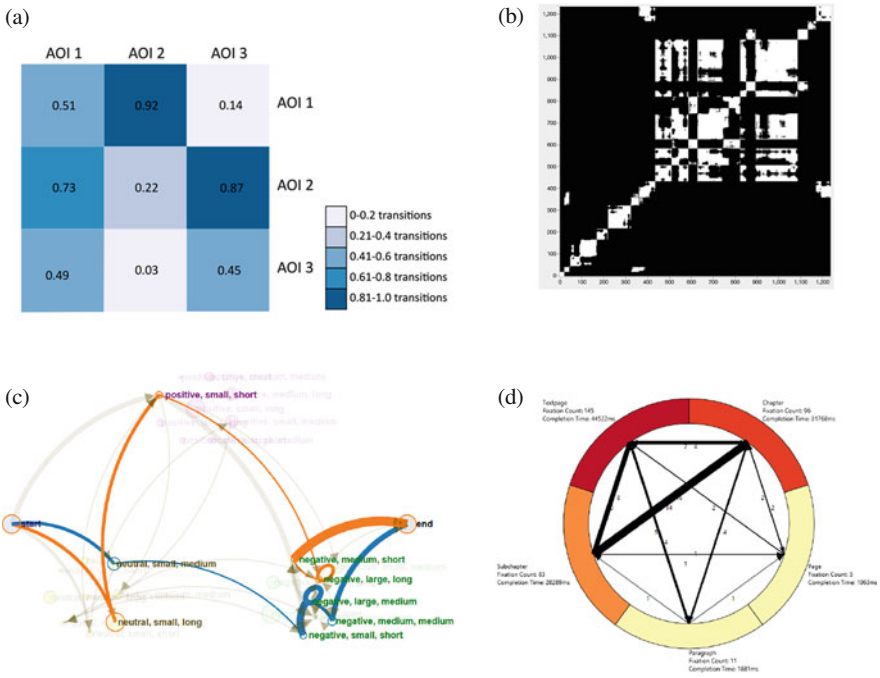


Fig. 6 Visualization techniques to investigate relations in eye-tracking data. (**a**) Transition matrix. (**b**) Recurrence plot (with kind permission from Demiralp et al. [14]). (**c**) State graph (with kind permission from Nguyen et al. [35]). (**d**) Circular transition graph

between AOIs are shown as arrows where the thickness corresponds to the transition count. Trees are typically used to depict the sequence of transitions [5]. These trees can also be used to visually compare the sequences of different participants and depict common strategies in a visual form [33, 45, 48].

3.6 Detect

Detecting patterns of common viewing behavior is important and often achieved by summarization or aggregation of the data. Such summarizations can also be applied to find outliers in the data which might either result from a problem of the hardware or from unexpected and potentially interesting behavior of a participant.

Descriptive statistics are often applied to achieve this goal. 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 additionally depict outliers as a simple-to-understand graph. However, more sophisticated visualization techniques can be utilized to summarize the eye movement data and detect outliers visually.

As mentioned before, summarization visualizations provide a first overview of the data. Summaries can be created for the raw data points, for aggregated data using AOIs, or for the participants. An overview of the point-based data can be visually represented by the fixation coverage displayed on top of the stimulus [10]. This technique allows one to see which parts of the stimulus have been looked at or not. Another possibility is to depict one dimension of the fixation position plotted against time [19]. This allows investigating the general scanning tendency of a participant. Other overview visualizations for point-based data have been described in the previous sections and include the space-time cube (Fig. 3a) and the attention map (Fig. 1b).

Some visualizations are especially designed, or suitable, for detecting outliers and deviations in the data. A visual method for analyzing raw eye movement data can be used to investigate if the raw data is inaccurate or incomplete [42]. Outliers in the recorded, as well as in the processed, analysis data can be identified using visualizations that represent the eye movements in a point-based fashion. Here, timeline visualizations [19, 20, 31] showing one data dimension over time can be applied.

An AOI view facilitates a simple summarization of eye movement data on the basis of AOIs. Here, AOIs are depicted on top of the stimulus and are color coded based on a measure (e.g., gaze duration) [5, 40]. This allows us to analyze how often and which AOIs have been looked at, keeping the spatial context of the AOI on the stimulus. Another technique is to depict AOI rivers [9] (Fig. 7), which represent AOIs on a timeline and where the thickness of each AOI river shows the number of gazes as well as outgoing and incoming transitions.

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 to perform the task. AOI timelines can help answer this question (Fig. 4). As discussed in Sect. 3.4, presenting AOIs next to each other [26, 37] allows a direct comparison to inspect which AOIs have been looked at or not. Furthermore, individual participants may display a different strategy, which can be found when matching participants using scanpath comparison (Sect. 3.3).

4 Example

In this section, we provide a concrete example of how the discussed analysis tasks relate to eye-tracking data. Our example dataset comes from the benchmark provided by Kurzhals et al. [28]. The video shows a 4×4 memory game where the cards are pairwise flipped until all matches are discovered. Participants ($N = 25$) were asked to identify matching cards by watching the video. Figure 8 shows the stimulus and different methods to visualize the recorded gaze data.

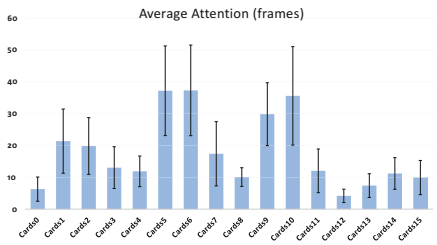
Fig. 7 AOI rivers that show the distribution of attention on different AOIs (colored streams) and transitions between them over time

First, we assume that no information about AOIs is available. According to our pipeline (Fig. 2), the recorded gaze data can be processed by fixation detection algorithms, providing analysis data solely based on gaze information. At this early stage in the analysis process, we could apply established eye-tracking metrics (e.g., average fixation count and saccade length) to derive general information of how the participants watched the stimulus video. This kind of analysis would be typical for tasks in the categories *relate* and *compare*.

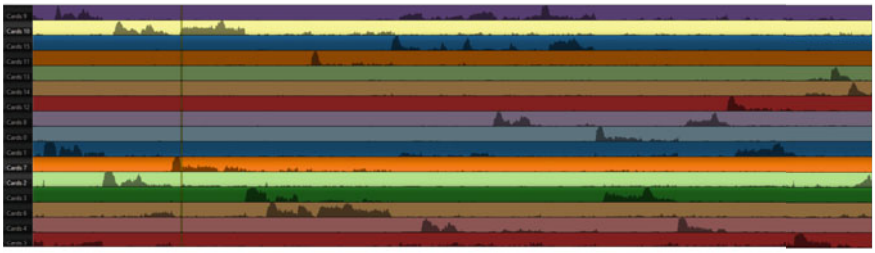
Attentional synchrony [43] is a specific behavior that occurs during the investigation of dynamic stimulus content. It describes timespans when the majority of participants spent their attention on a specific region, which is often an effect of motion as an attention-guiding feature. Identifying attentional synchrony concerns the categories *when*, *where*, and *detect*. With a space-time cube visualization (Fig. 8d), it is possible to *detect* timespans of attentional synchrony in the spatio-temporal context of the stimulus, meaning that it is quite easy to identify *when* many participants looked at the same region (*where*). In our example, this is typically the case when a new card is flipped, drawing the attention of almost all participants in expectation of the new card image.

To this point, the statistical, hypothesis-driven analysis of the recorded data can be interpreted as a linear process where metrics are applied and the results are reported. Complementary, in an interactive visualization such as the space-time cube, the data can be clustered and filtered to explore the dataset and identify events of potential interest for new hypotheses. With these possibilities of interacting with the data, the foraging and sensemaking loops (Fig. 2) are initiated.

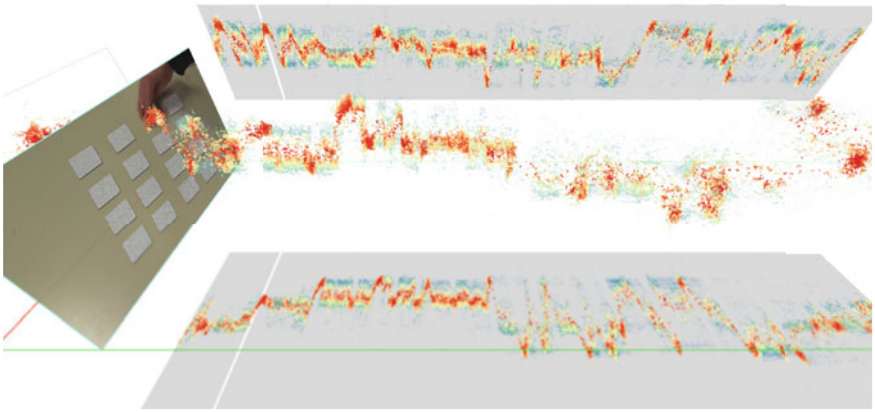
(a) Memory stimulus containing 16 cards that are flipped until all pairs are discovered



(b) Average number and standard deviation of frames with gaze data inside the AOIs



(c) AOI timelines for all cards, indicating when a card was investigated



(d) Space-time cube visualization showing spatio-temporal context of attentional synchrony

Fig. 8 Example of eye-tracking analysis for a video stimulus: (**a**) The video is annotated with AOIs, (**b**) descriptive statistics can be applied to investigate the distribution of attention, (**c**) AOI timelines show detailed information about the temporal distribution, and (**d**) a space-time cube provides spatio-temporal context

In many eye-tracking experiments, the annotation of AOIs, as an additional source for data analysis, is performed. In our example, each individual card represents an AOI (Fig. 8a). With this additional information, hit detection between gaze points and the AOI shapes can be performed to compare the distribution of attention between AOIs (Fig. 8b). As described above, these aggregated metrics

provide an overview, but due to the lack of temporal information, analysis questions considering *when* something happened cannot be answered. To get an overview of the distribution of attention over time, individual timelines for each AOI can be applied (Fig. 8c). The histograms indicate how many participants looked at an AOI at specific points in time. For example, it becomes quite obvious which two cards were flipped in one turn: their corresponding timelines show high values simulatenously. Another example are peaks for single AOIs, indicating attentional synchrony with a focus on the *when* aspect.

Considering the *who* questions, scarf plots could be applied to visualize the AOI sequences of individual participants. Similarity measures between AOI sequences can lead to insights considering different participant groups. As an example, Kurzhals et al. [31] discuss the event when the first matching pair of cards (Card 7 and Card 10) was visible for the first time (vertical line in Fig. 8c). In the turn before this event, Card 2 and Card 10 were flipped. After all cards are covered again, Card 7 is flipped. From this specific point in time, three different viewing patterns can be identified: (1) a group of participants stays on Card 7, possibly trying to remember the position of the matching card; (2) the majority of participants immediately looks at Card 10, where the matching card is; (3) some participants also look at Card 2, indicating problems to remember which card matches. Using a scarf plot visualization, it is possible to identify which participants belong to the identified groups of similar behavior (*detect*). Furthermore, it is possible to apply visualization techniques considering *relate* questions. A transition matrix, for example, could indicate how well individual cards were remembered, i.e., by high transition counts from an AOI to the matching AOI.

In summary, this example provides only a glimpse into possible analysis tasks and applicable visualization techniques that can be covered with the proposed categorization scheme. Based on our example, it also becomes obvious that many eye-tracking related analysis tasks are compositions of the categories and could be solved with different techniques. Therefore, our book chapter wants to provide a guide of possibilities, rather than dogmatic solutions.

5 Conclusion

We have adopted a task-oriented perspective on visualization and visual analysis of eye-tracking data. We have derived a task taxonomy for eye movement data visualization based on a literature research of such visualization approaches, corresponding case studies, and user evaluations. Furthermore, the taxonomy is related to our pipeline of eye-tracking visualization that includes data acquisition via recordings during eye-tracking experiments, processing and annotation of that data, and visualization, finally leading to descriptions and examination of hypotheses and building new knowledge.

One aspect of the task taxonomy adopts the fundamental characteristics of the most important data dimensions in gaze recordings: space (where?), time (when?),

and participant (who?). These data-oriented tasks are complemented by another class of tasks covering analytical operations: compare, relate, and detect. For all tasks, we provide references to representative visualization techniques. In this way, our chapter is meant as a starting point for choosing appropriate methods of visual analysis for the problem at hand. It should be noted that our discussion of previous work does not target comprehensiveness. A systematic review of the literature on eye-tracking visualization can be found elsewhere [4].

Acknowledgements This work was partially supported by the German Research Foundation (DFG) within the Cluster of Excellence in Simulation Technology (EXC 310) at the University of Stuttgart. This work was supported in part by EU in projects datAcron (grant agreement 687591) and VaVeL (grant agreement 688380).

References

1. Andrienko, G.L., Andrienko, N.V., Burch, M., Weiskopf, D.: Visual analytics methodology for eye movement studies. IEEE Trans. Visual. Comput. Graph. **18**(12), 2889–2898 (2012)
2. Blascheck, T., Burch, M., Raschke, M., Weiskopf, D.: Challenges and perspectives in big eye-movement data visual analytics. In: Proceedings of the 1st International Symposium on Big Data Visual Analytics. IEEE, Piscataway (2015)
3. Blascheck, T., John, M., Kurzhals, K., Koch, S., Ertl, T.: VA^2: a visual analytics approach for evaluating visual analytics applications. IEEE Trans. Visual. Comput. Graph. **22**(01), 61–70 (2016)
4. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: State-of-the-art of visualization for eye tracking data. In: EuroVis– STARs, pp. 63–82. The Eurographics Association (2014)
5. Blascheck, T., Kurzhals, K., Raschke, M., Strohmaier, S., Weiskopf, D., Ertl, T.: AOI hierarchies for visual exploration of fixation sequences. In: Proceedings of the Symposium on Eye Tracking Research & Applications. ACM, New York (2016)
6. Blascheck, T., Raschke, M., Ertl, T.: Circular heat map transition diagram. In: Proceedings of the 2013 Conference on Eye Tracking South Africa, pp. 58–61. ACM, New York (2013)
7. Bojko, A.: Informative or misleading? Heatmaps deconstructed. In: Jacko, J. (ed.) Human-Computer Interaction. New Trends, LNCS'09, pp. 30–39. Springer, Berlin (2009)
8. Burch, M., Andrienko, G.L., Andrienko, N.V., Höferlin, M., Raschke, M., Weiskopf, D.: Visual task solution strategies in tree diagrams. In: Proceedings of the IEEE Pacific Visualization Symposium, pp. 169–176. IEEE, Piscataway (2013)
9. Burch, M., Kull, A., Weiskopf, D.: AOI Rivers for visualizing dynamic eye gaze frequencies. Comput. Graph. Forum **32**(3), 281–290 (2013)
10. Bylinskii, Z., Borkin, M.A., Kim, N.W., Pfister, H., Oliva, A.: Eye fixation metrics for large scale evaluation and comparison of information visualizations. In: Burch, M., Chuang, L., Fisher, B., Schmidt, A., Weiskopf, D. (eds.) Eye Tracking and Visualization. Foundations, Techniques, and Applications (ETVIS 2015), pp. 235–255. Springer, Heidelberg (2016)
11. Chi, E.H.: A taxonomy of visualization techniques using the data state reference model. In: IEEE Symposium on Information Visualization, pp. 69–75. IEEE Computer Society, Los Alamitos (2000)
12. Crowe, E.C., Narayanan, N.H.: Comparing interfaces based on what users watch and do. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 29–36. ACM, New York (2000)

13. De Urabain, I.R.S., Johnson, M.H., Smith, T.J.: GraFIX: a semiautomatic approach for parsing low-and high-quality eye-tracking data. Behav. Res. Methods **47**(1), 53–72 (2015)
14. Demiralp, C., Cirimele, J., Heer, J., Card, S.: The VERP Explorer: a tool for exploring eye movements of visual-cognitive tasks using recurrence plots. In: Burch, M., Chuang, L., Fisher, B., Schmidt, A., Weiskopf, D. (eds.) Eye Tracking and Visualization. Foundations, Techniques, and Applications (ETVIS 2015), pp. 41–55. Springer, Heidelberg (2016)
15. Duchowski, A.: Eye Tracking Methodology: Theory and Practice, 2nd edn. Science+Business Media. Springer, New York (2007)
16. Duchowski, A.T., Driver, J., Jolaoso, S., Tan, W., Ramey, B.N., Robbins, A.: Scan path comparison revisited. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 219–226. ACM, New York (2010)
17. Fayyad, U., Piatetsky-Shapiro, G., Smyth, P.: The KDD process for extracting useful knowledge from volumes of data. Commun. ACM **39**(11), 27–34 (1996)
18. Goldberg, J.H., Helfman, J.I.: Scanpath clustering and aggregation. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 227–234. ACM, New York (2010)
19. Goldberg, J.H., Helfman, J.I.: Visual scanpath representation. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 203–210. ACM, New York (2010)
20. Grindinger, T., Duchowski, A., Sawyer, M.: Group-wise similarity and classification of aggregate scanpaths. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 101–104. ACM, New York (2010)
21. Haber, R.B., McNabb, D.A.: Visualization idioms: a conceptual model for visualization systems. In: Nielson, G.M., Shriver, B.D., Rosenblum, L.J. (eds.) Visualization in Scientific Computing, pp. 74–93. IEEE Computer Society Press, Los Alamitos (1990)
22. Heinrich, J., Weiskopf, D.: State of the art of parallel coordinates. In: STAR Proceedings of Eurographics, pp. 95–116. The Eurographics Association (2013)
23. Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., Van de Weijer, J.: Eye Tracking: A Comprehensive Guide to Methods and Measures. Oxford University Press, Oxford (2011)
24. Inselberg, A.: Parallel Coordinates: Visual Multidimensional Geometry and Its Applications. Springer, New York (2009)
25. Just, M.A., Carpenter, P.A.: Eye fixations and cognitive processes. Cognit. Psychol. **8**, 441–480 (1976)
26. Kim, S.H., Dong, Z., Xian, H., Upatising, B., Yi, J.S.: Does an eye tracker tell the truth about visualizations? Findings while investigating visualizations for decision making. IEEE Trans. Visual. Comput. Graph. **18**(12), 2421–2430 (2012)
27. Kirchner, H., Thorpe, S.J.: Ultra-rapid object detection with saccadic eye movements: visual processing speed revisited. Vis. Res. **46**(11), 1762–1776 (2006)
28. Kurzhals, K., Bopp, C.F., Bässler, J., Ebinger, F., Weiskopf, D.: 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, pp. 54–60. ACM, New York (2014)
29. Kurzhals, K., Fisher, B.D., Burch, M., Weiskopf, D.: Evaluating visual analytics with eye tracking. In: Proceedings of the Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization, pp. 61–69. ACM, New York (2014)
30. Kurzhals, K., Heimerl, F., Weiskopf, D.: ISeeCube: visual analysis of gaze data for video. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 43–50. ACM, New York (2014)
31. Kurzhals, K., Hlawatsch, M., Heimerl, F., Burch, M., Ertl, T., Weiskopf, D.: Gaze stripes: image-based visualization of eye tracking data. IEEE Trans. Visual. Comput. Graph. **22**(1), 1005–1014 (2016)
32. Kurzhals, K., Weiskopf, D.: Space-time visual analytics of eye-tracking data for dynamic stimuli. IEEE Trans. Visual. Comput. Graph. **19**(12), 2129–2138 (2013)
33. Kurzhals, K., Weiskopf, D.: AOI transition trees. In: Proceedings of the Graphics Interface Conference, pp. 41–48. Canadian Information Processing Society (2015)

34. Li, X., Çöltekin, A., Kraak, M.J.: Visual exploration of eye movement data using the space-time-cube. In: Fabrikant, S., Reichenbacher, T., Kreveld, M., Christoph, S. (eds.) Geographic Information Science, LNCS'10, pp. 295–309. Springer, Berlin (2010)
35. Nguyen, T.H.D., Richards, M., Isaacowitz, D.M.: Interactive visualization for understanding of attention patterns. In: Burch, M., Chuang, L., Fisher, B., Schmidt, A., Weiskopf, D. (eds.) Eye Tracking and Visualization. Foundations, Techniques, and Applications (ETVIS 2015), pp. 23–39. Springer, Heidelberg (2016)
36. Pirolli, P., Card, S.: The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In: Proceedings of the International Conference on Intelligence Analysis, vol. 5, pp. 2–4 (2005)
37. Räihä, K.J., Aula, A., Majaranta, P., Rantala, H., Koivunen, K.: Static visualization of temporal eye-tracking data. In: Costabile, M.F., Paternò, F. (eds.) Human-Computer Interaction-INTERACT 2005, LNCS'05, vol. 3585, pp. 946–949. Springer, Berlin/New York (2005)
38. Raschke, M., Herr, D., Blascheck, T., Burch, M., Schrauf, M., Willmann, S., Ertl, T.: A visual approach for scan path comparison. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 135–142. ACM, New York (2014)
39. Richardson, D.C., Dale, R.: Looking to understand: the coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. Cognit. Sci. **29**(6), 1045–1060 (2005)
40. Rodrigues, R., Veloso, A., Mealha, O.: A television news graphical layout analysis method using eye tracking. In: Proceedings of the International Conference on Information Visualization (IV), pp. 357–362. IEEE Computer Society, Los Alamitos (2012)
41. Salvucci, D.D., Goldberg, J.H.: Identifying fixations and saccades in eye-tracking protocols. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 71–78. ACM, New York (2000)
42. Schulz, C., Burch, M., Beck, F., Weiskopf, D.: Visual data cleansing of low-level eye-tracking data. In: Burch, M., Chuang, L., Fisher, B., Schmidt, A., Weiskopf, D. (eds.) Eye Tracking and Visualization. Foundations, Techniques, and Applications (ETVIS 2015), pp. 199–216. Springer, Heidelberg (2016)
43. Smith, T.J., Mital, P.K.: Attentional synchrony and the influence of viewing task on gaze behavior in static and dynamic scenes. J. Vis. **13**(8), 16:1–16:24 (2013)
44. Stellmach, S., Nacke, L., Dachselt, R.: Advanced gaze visualizations for three-dimensional virtual environments. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 109–112. ACM, New York (2010)
45. Tsang, H.Y., Tory, M.K., Swindells, C.: eSeeTrack – visualizing sequential fixation patterns. IEEE Trans. Visual. Comput. Graph. **16**(6), 953–962 (2010)
46. Tversky, B., Morrison, J.B., Bétrancourt, M.: Animation: can it facilitate? Int. J. Hum. Comput. Stud. **57**(4), 247–262 (2002)
47. Weibel, N., Fouse, A., Emmenegger, C., Kimmich, S., Hutchins, E.: Let's look at the cockpit: exploring mobile eye-tracking for observational research on the flight deck. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 107–114. ACM, New York (2012)
48. West, J.M., Haake, A.R., Rozanski, E.P., Karn, K.S.: eyePatterns: software for identifying patterns and similarities across fixation sequences. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 149–154. ACM, New York (2006)
49. Yarbus, A.L.: Eye Movements and Vision. Plenum Press, New York (1967)

Interactive Visualization for Understanding of Attention Patterns

Truong-Huy D. Nguyen, Magy Seif El-Nasr, and Derek M. Isaacowitz

Abstract Discovering users' behavior via eye-tracking data analysis is a common task that has important implications in many domains including marketing, design, behavior study, and psychology. In our project, we are interested in analyzing eye-tracking data to investigate differences between age groups in emotion regulation using visual attention. To achieve this goal, we adopted a general-purposed interactive visualization method, namely Glyph, to conduct temporal analysis on participants' fixation data. Glyph facilitates comparison of abstract data sequences to understand group and individual patterns. In this article, we show how a visualization system adopting the Glyph method can be constructed, allowing us to understand how users shift their fixations and dwelling given different stimuli, and how different user groups differ in terms of these temporal eye-tracking patterns. The discussion demonstrates the utility of Glyph not only for the purpose of our project, but also for other eye-tracking data analyses that require exploration within the space of temporal patterns.

1 Introduction

Eye-tracking hardware and software has facilitated research on visual atten-tion and decision-making with implications for marketing and design [23, 31, 39]. For in-stance, in the field of psychology, eye tracking has been used to investigate emotion regulation, visual attention, age variation related to attention, etc. [22, 34]. Moreover, eye-tracking research has been situated in many different environments to tackle different applications, such as advertisement and marketing through looking at visual patterns while watching TV or at a store [9, 17], software design through looking at eye patterns while playing games or using a website [18, 28], and automobile design through eye tracking while driving [24, 27], to mention a few.

T.-H.D. Nguyen (✉)
Texas A&M University-Commerce, 2200 Campbell St, Commerce, TX 75428, USA
e-mail: truong-huy.nguyen@tamuc.edu

M. Seif El-Nasr • D.M. Isaacowitz
Northeastern University, 360 Huntington Avenue, Boston, MA, 02115, USA
e-mail: m.seifel-nasr@neu.edu; D.Isaacowitz@neu.edu

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_2

Eye-tracking tools for analysis and data capture have been gradually evolving for their utility in such wide variety of investigations. In this chapter, we specifically target interactive visualization with eye-tracking data as an analysis method for psychology research.

In particular, we are interested in investigating the effect of age on the use of certain emotion regulation strategies, with eye fixation as the behavioral manifestation of the specific emotion regulation strategy of attentional deployment [15]. The eye-tracking data is collected from an experiment in which participants of different age and instruction condition groups select one video clip at a time to watch. The overall research question for the study was: *how do adults of different ages use affective choices and visual fixation to regulate negative mood states?* As such, eye tracking was used to record visual fixations to selected stimuli.

Typically, researchers who record eye-tracking data focus their analyses on aggregations, such as summations of fixations within pre-specified "Areas of Interest" (AOIs) and compare summed fixations toward some AOIs to others, either within or across groups of research participants. Current methods used to aid in this analysis typically visualize eye-tracking data as point-based and AOI-based visualizations [5]. The point-based approach focuses on the movement of fixations and does not require any semantic an-notation on the data. In the AOI-based approach researchers annotate the stimuli in terms of areas of interest and visualizations are used to show how fixations are associated with such areas of interests, and thus giving them meaning and providing context. None of these previous methods have looked at a way to both (a) analyze temporal shifts and patterns of fixation data, and (b) compare these patterns across groups.

In order to analyze the eye-tracking data to answer the research question put forth above, we adopted a visual analytic technique, called *Glyph* [30], to construct a visualization tool for exploration and understanding of look patterns in a two-dimensional, dynamic, active stimuli setup (i.e., video clip watching activity). The coordinated multi-view interactive visualization system developed, henceforth referred to as Glyph, presents data at two different levels of granularity. Glyph uses an abstract representation of a state transition space represented as a graph to show how each participant makes visual attention choices over time. It coordinates this view with a view that shows how the patterns that users exhibited are either different or similar by having those that are similar close together and those that are different further away. By interacting with the Glyph system, it is expected that users are able to compare attention behaviors, thereby gaining better understanding of participants' choices and how they compare in their patterns. We used *Glyph* before to uncover player action patterns in video games [30]; thanks to *Glyph*'s flexibility in dealing with abstract data, its paradigm can be a promising approach in understanding look patterns in eye-tracking data re-search.

For the rest of the paper, we first discuss our psychology experiment setup and research questions of interest. Next, we summarize common eye-tracking data visualizations that could be useful for our study, while highlighting their limitations in satisfactorily answering our research questions in the experiment. We then

describe the proposed Glyph system. Finally, we discuss some preliminary findings when using the system, before concluding the work.

2 Understanding Emotion Regulation Strategies

In this project, we aimed to investigate the effect of age on the use of certain emotion regulation strategies. While the process model of emotion regulation [15] suggests a number of different emotion regulation strategies that an individual may use, such as reappraisal and suppression, work specifically on aging has suggested that older adults may favor the strategy of attentional deployment [21]. Previous research also indicates a positivity bias in attention and memory on the part of older adults when compared to younger and middle aged adults [33]. This means that, in general, older adults report being happier than their younger counterparts, and also tend to focus their attention and memory on positive rather than negative material [33]. Such findings raise the question: does a more positive attentional deployment strategy help older adults to feel good? Eye-tracking data analysis is a potential device that can help answer these questions, i.e., whether there is an age difference in how individuals modulated their attention to change how they feel [22]. As such, the overall research question for the study was: *how do adults of different ages use affective choices and visual fixation to regulate negative mood states?*

2.1 The Study

In order to answer the above question, we designed an experiment, in which participants are tasked to choose from a variety of clips that might elicit more positive and negative emotions when watched. Prior to the task, participants are guided into a relatively more negative mood state, so their behavior in the task could be used as an indicator of their attempts at emotion regulation. Participants choose from among various stimuli what they could watch. For in-stance, a clip selected comes from the last scenes of the movie "Marley and Me", showing memories the main character had with his dog while facing the fact that the pet is dying soon. While the memories generally contain fond and happy moments, the up-close shots of the dying dog could cause great sadness to viewers. As such, a clip can contain periods of scenes that elicit opposite emotions, and participants can choose to look at these scenes or not. After each clip, they can freely select another clip in the clip repository to watch next. By tracking both users' choice of clips and their attention focus via eye fixations, we hope to shed some light on possible strategies users adopt to regulate their emotions when given these dimensions of freedom.

Initially 150 subjects were recruited. After filtering (i.e., only include participants with gaze tracked at least 75 % of the time), there are a total of 42 young adults (18–34 years old, $\mu = 20.52, \sigma = 1.58$), 45 middle-aged (35–64 years old, $\mu = 48.2, \sigma = 6.73$) and 45 old adults ($\geq$65years old, $\mu = 70.85, \sigma = 7.35$) in the

data set. They are further divided into two groups: the treatment group (N = 65) are asked to stay positive throughout the session of clip watching, while the control group (N = 67) are not given any specific instruction for how to watch the videos. Because participants can select which clips to watch, not every participant in the study watched the same clips. Throughout the session, each participant watches the clips alone by themselves to avoid any noise and bias caused from group interaction. The gaze points of participants throughout the session are recorded, showing where they look, when, and for how long.

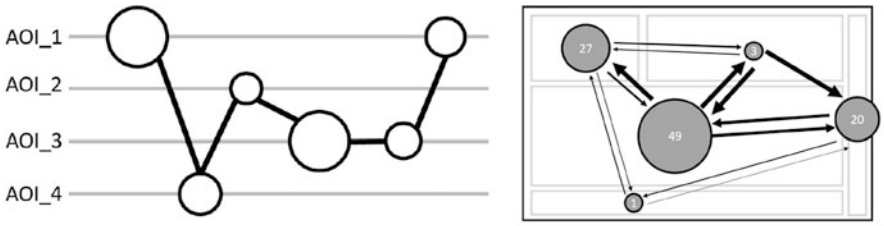
In this project, we are specifically interested in understanding how participants deploy their attention, with instruction condition and age being two between-subject variables that may affect their attention patterns. Exploring the data using visualization is one of the first analysis steps we would like to conduct to better understand the collected data and form hypotheses. Since we are interested in uncovering the temporal attention patterns participants in different groups exhibited, visualization techniques that aim to visualize sequential eye-tracking data are our focus.

3 Related Work

Frequently used visualization techniques designed for eye-tracking data can be roughly divided into two categories: aggregated plots that disregard temporal information, and those that aim to reflect temporal relationships in the data. The first category includes statistical graphs, such as line, bar charts, scatter/box plots, etc. [2, 3, 8, 10, 35], and heat maps [7, 12, 25, 29, 36]. Heat maps are often overlaid on the stimulus as a way to connect the visualized data to its context. The second category comprises of techniques that accumulate eye-tracking data in the visualization without losing temporal information, such as timeline and scan path visualizations [13, 14, 20, 26]. A thorough categorization of visualization techniques designed for exploratory eye-tracking data analysis can be found in the article by Blascheck et al. [5]. The visualization technique presented in this article falls into the second category, as we would like to examine attention paths.

In plotting eye-tracking data trails, there are two main approaches: timelines and relational visualizations. In the former approach, time is represented as an axis in the coordination system, such as the x-(horizontal) axis in a 2D space [16, 32], or a third axis in a 3D space [25]. For instance, time plots [32] represents different AOIs as different lines on the y-axis and time on the x-axis, while the node size represent the duration of attention (Fig. 1a).

The latter approach, on the other hand, does not dedicate a specific dimension to time. Instead, it encodes temporal information as transitions between AOI nodes in a node-link representation [6, 19, 37]. For instance, if the AOIs are represented as nodes, the node size can encode dwelling duration, while the link thickness depicts the frequency of transition (Fig. 1b). More statistical information such as overall dwelling percentage can be displayed as text overlaid on respective nodes.



(a) Time plot (b) Relational AOI visualization

Fig. 1 AOI-based approaches; (**a**) time plots with x-axis as time, circle size as duration and y-axis as AOI index/type, (**b**) relational AOI visualizations with AOIs being nodes and links transitions between them (Figures are adapted respectively from [19, 32])

Our visualization as explained in subsequent sections adopts a similar approach by accumulating AOI sequences of participants into a node-link graph. However, in our graph, each node represents an abstract state (i.e., *look state*), instead of a certain AOI.

One noticeable limitation of the surveyed techniques is that they do not cater to dynamic situations in which the stimuli are both directly controlled by users and changing frequently since they are in video format. Most techniques associate fixations with static regions that stay the same throughout the course of the experience. While there are visualization techniques proposed for visualizing eye-tracking data with dynamic stimuli in videos [11, 25], they do not facilitate data abstraction, which is important for understanding behavior patterns beyond the scope of one single video segments, e.g. from segment to segment, or clip to clip. In the experiment study explained in Sect. 2.1, users can actively select the video clips they are watching at will and in any order they like. Therefore, analyses that track look patterns on static regions or do not facilitate data abstraction will likely show that every participant exhibits a different pattern due to the high degree of freedom, making it hard to identify and understand commonalities and individual uniqueness in the track data.

4 Glyph Visualization

There are many ways to present eye-tracking data visually to researchers (as summarized in Sect. 3), but in order to make it easy to compare participants and understand the common and unique patterns, we opted for visualizing abstracted data instead of the raw counterparts, using a general data visualization system called Glyph (Fig. 2). Glyph requires eye-tracking data to be abstracted into sequences of states, before visualizing the data in a coordinated multi-view interface. The data abstraction step is when users of Glyph can specify the relevant features of interest to be visualized. We will later describe the abstraction adopted in this project and use it to exemplify the data abstraction process needed to prepare the data for Glyph.

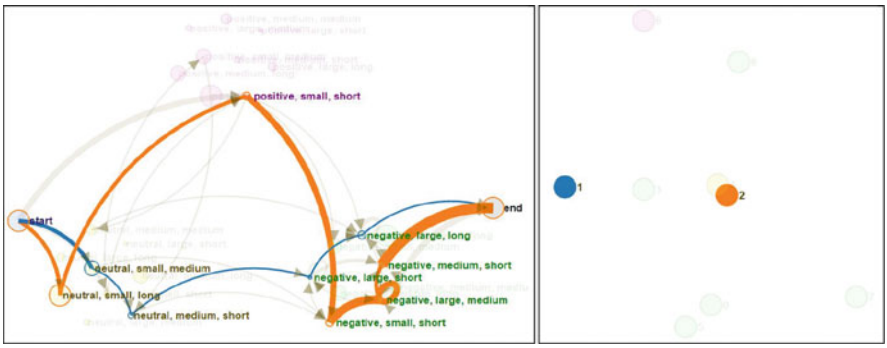


Fig. 2 Comparing visual attention patterns using Glyph. The data used for visualization is abstracted from raw eye-tracking data, capturing features of interest such as the size and affective value of AOIs, as well as participant's fixation duration. The *left graph* shows moment-to-moment patterns, while *the right* shows overall differences

Given eye-tracking data transformed into abstracted sequences, the goal is to visualize the paths that people in different groups took when watching a sequence of stimuli over time, such as those contained in a video clip. Additionally, we would like the visualization to support the basic exploration acts of (1) identifying common and unique paths, and (2) comparing different paths to one another. Glyph comprises of two coordinated views of sequential behavior data, showing the state graph and the sequence graph, which works perfectly for this purpose (Fig. 2).

4.1 Data Abstraction

In our experiment, each clip that participants watch is annotated a priori with AOIs, each of which is a segment of the original clip, and marked up with one of the following affective labels: "positive", "negative", and "neutral". While positive and negative AOIs are those that presumably cause watchers to experience respective emotions, neutral AOIs are visually interesting by themselves but do not carry any emotion meaning. For each AOI, we know where and how much screen estate it takes up within the frames where it appears, as well as how long it lasts.

Participants' eye fixation data was recorded at a temporal resolution of 30 Hz, and an accuracy of .050–1.00 visual angle using an ASL MobileEye XG eye-tracker produced by Applied Science Laboratories in Bedford, MA. Fixations were defined in the system as holding a point of gaze for 100 ms without deviating more than one visual degree. As such the fixation data details with high accuracy where the participants look at and when.

Given AOI information, we could abstract the raw data, turning them into sequences of *look states*, each of which consists of three descriptors:

1. The emotion associated with the AOI: negative, positive, or neutral. Negative AOIs contain unpleasant scenes (e.g., a dying dog), while positive ones contain visuals that are uplifting or pleasant to view (e.g., a dog playing cheerfully with the owner in the back yard). Neutral AOIs on the other hand consist of visually salient objects, which naturally draw the viewers' attention but do not aim at any specific emotion (e.g., a scene showing a stopping motion of a car in an ordinary, non-sudden, non-heads-on manner).
2. The size of the AOI: small, medium, or large. A small AOI covers less than 25 % of the screen during in its duration, medium 25–49.99 % of the screen, and large greater than 50 %.
3. The duration of looking, i.e., short, medium, or long. This attribute is computed in comparison to the total duration of the AOI. A short look only lasts less than one third of the total duration, medium 1/3–2/3, and large greater than 2/3.

Each participant's eye-track log is processed in this way over the course of the clip's duration, to return a sequence of look states.

In our project, we are interested in how significant an act of attention focus is, i.e., the "value" of each look state. In particular, a focus on a small-sized AOI over a long period of time is considered as being a more significant attention act than a short focus on a large-sized AOI over a short period of time. In this case, the first act is a strong sign that the participant is attracted towards or consciously focusing their attention on the AOI, while the second is not as strong. To capture this notion of look significance, we define the value of a look state s as:

$$V(s) = \frac{\text{affect}(s) \times \text{duration}(s)}{\text{size}(s)}$$

whereby

- affect(s) is a numeric representation of the affective state associated with the AOI, i.e., -1 for negative, 0 for neutral, and 1 for positive AOIs. Note that this means all neutral look states have value 0, i.e., the look states associated with neutral AOI are not differentiated based on their size or duration, in comparing full sequences. This treatment is eligible, since we are more interested in emotion regulation with respect to positivity/negativity.
- duration(s): the look's duration; short is 1, medium 2, and long 3.
- size(s): the size of the AOI; small is 1, medium 2, and large 3.

$V(s)$ is therefore positive for look states on positive AOIs, and negative for those on negative AOIs. The magnitude of $V(s)$, i.e., $|V(s)|$, captures the significance of s.

4.2 State Graph

State graph summarizes the look trails exhibited by all participants. It consists of nodes as look states, and links as transition decisions. For instance, a directed link transitioning from look state <*positive, small, short*> to look state <*negative, small, long*> signifies that the participant, after quickly looking ("short") at a small-sized AOI ("small") that elicits positive emotion ("positive"), spends more time ("long") scrutinizing a small-sized AOI ("small") eliciting negativity ("negative").

Node size in this graph is used as visual cues to some property of the look states, such as their popularity, i.e., how frequently the group members land on the state, or the value of look states as shown in Fig. 3. In this case, the bigger the node, the more significant the look state is. For example, in Fig. 3, the positive look state (pink) with many inward and outward links is not significant.

The popularity of transitions is encoded as link thickness. We further use color to depict the affective state of each node: pink is positive, yellow is neutral, and green is negative. The layout of the graph can be force-directed, or clustered according to some prefixed semantic information such as the affective type of the look state, as shown in Fig. 3. The goal of this graph is to allow quick detection of popular transitions, leading to discovery of common group patterns.

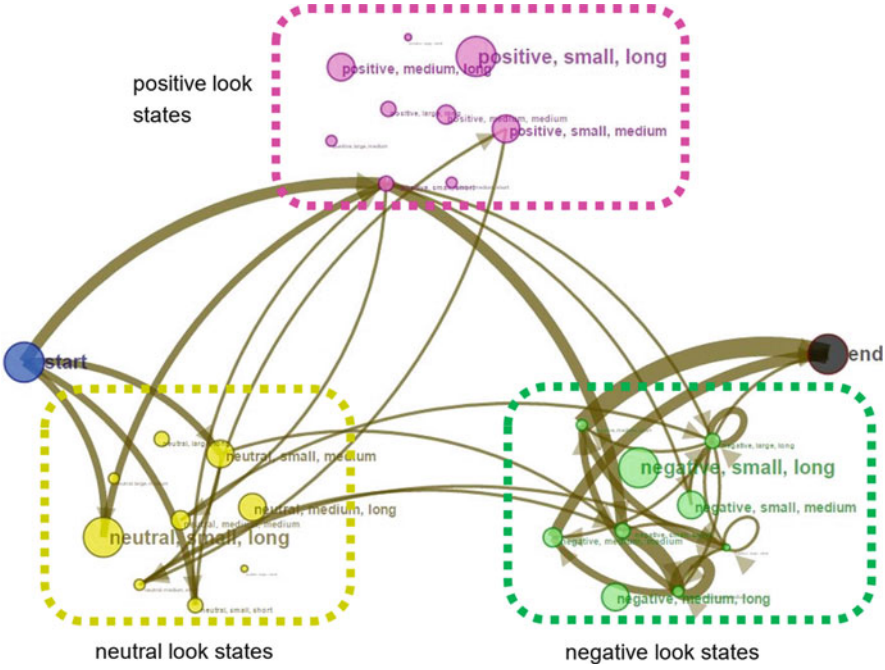


Fig. 3 State graph of treatment group data when watching "Marley and Me" clip. Look states associated with positively affective AOIs are *pink nodes*, *neutrally yellow*, and *negatively gree*n; nodes of the same affective type are clustered in their respective groups. Start (*blue*) and end (*black*) nodes do not correspond to any AOI type; they act as landmarks in the graph

Figure 3 depicts an exemplary state graph resulted from visualizing abstracted data obtained from our case study (discussed below) where participants are asked to watch a video clip called "Marley and Me". In this clip, there are four negative, two positive, and two neutral AOIs, as such the majority of the AOIs are negative. Depending on whether participants choose to look at or skip certain AOIs, and how long their eye fixations are in the former case, each participant exhibits a different path from start to end, all of which are collated to form the links and nodes in the graph. Nodes that do not have any links associated with them either (1) denote AOIs that do not exist in the clip, e.g., there is no large positive AOI in "Marley and Me", or (2) represent behavior not performed by any participant. For example, the graph in Fig. 3 shows that nobody spent significant (long) time on small, negative AOIs in this clip. The same applies for links; no link between two nodes indicates that no such transition is present in the data.

4.3 Sequence Graph

Different from the state graph, the sequence graph's (Fig. 4) role is to present visually an overview of complete state sequences. Specifically, each node in the sequence graph represents one complete sequence, which can be exhibited by one or more participants. Node size here represents the sequence's popularity, i.e., the more participants sharing the same sequence, the larger the corresponding node is. The text displayed by each node details the popularity rank of the node as an additional textual cue complementing the nodes' size; the lower the number, the more popular it is.

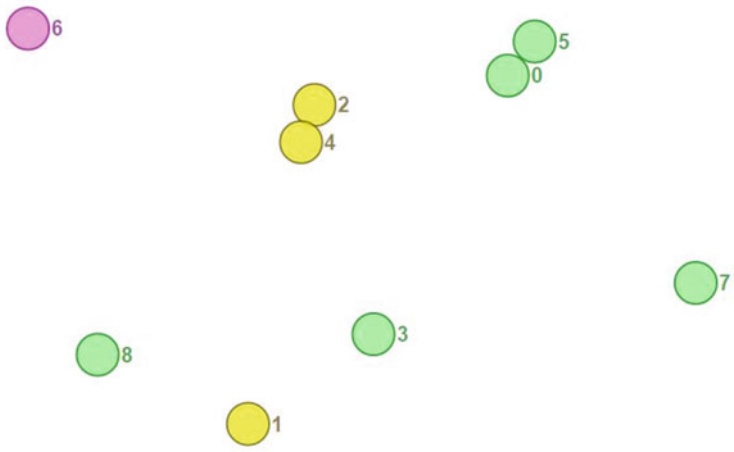


Fig. 4 Sequence graph with nodes colored according to the dominant emotion associated with the traced AOIs; *yellow nodes* are sequences dominated by neutral AOIs, *green negative*, and *pink positive*. Node labels indicate popularity ranks

The distances between these sequence nodes are determined using Dynamic Time Warping [4], a method generalizing Minimum Edit Distance [38] to compute sequence differences by accumulating state differences. In general, any metric function deemed suitable to capture the difference of abstracted states can be used, including traditional scanpath similarity measures [1] with appropriate adaptation to deal with abstracted states.

We first define the difference of look states as $d(s_1, s_2) = |V(s_1) - V(s_2)|$. The state difference, therefore, is large when the looks are vastly different, in terms of emotion, size, and/or look duration. With this metric function, the sequence difference can then be computed using Algorithm 1.

The strength of Dynamic Time Warping is that it can handle comparison of sequences of varying lengths, resulting from the data being recorded at varying time duration and speed.

Finally, each node is color-coded by the dominant emotion in the whole trace, determined as follows:

$$E_{dom}(a) = \text{argmax}_{k \in \{\text{pos,neu,neg}\}} \sum_{i=1}^{n} V(s_i) \cdot \sigma_k(s_i)$$

in which

- $a = \{s_1, s_2, \ldots, s_n\}$ is the sequence of look states represented by the sequence node a,
- $k \in \{\text{pos, neu, neg}\}$ is an emotion type,
- $V(s_i)$ is the value of the look state s_i, and
- $\sigma_k(s_i) = 1$ if $\text{affect}(s_i) = k$, and 0 otherwise

Algorithm 1: Dynamic time warping

Input : State difference function $d(s_k, q_l)$ for any state pair s_k and q_l
Two sequences $a = \{s_1, s_2, \ldots, s_n\}$ and $b = \{q_1, q_2, \ldots, q_m\}$
Output : Sequence difference $D(a, b)$

Notation: Denote $D(i, j)$ as the difference of two subsequences $\{s_1, s_2, , s_i\}$ of a and $\{q_1, q_2, , q_j\}$ of b

$D(0, 0) = 0$
foreach $i \in [1, n]$ **and** $j \in [1, m]$ **do**
  $D(i, 0) = D(0, j) = \infty$
end
for $i = 1 \rightarrow n$ **and** $j = 1 \rightarrow m$ **do**

$$D(i,j) = d(s_i, q_j) + \min \begin{bmatrix} D(i-1,j), \\ D(i,j-1), \\ D(i-1,j-1) \end{bmatrix}$$

end
return $D(n, m)$

In the implementation, we used pink to color-code positive emotion, yellow neutral, and green negative. The dominant emotion as summarized by the color code in the nodes allows a quick inspection of the main type of look states the participants have exhibited in their sessions.

4.4 Visual Coordination: Synchronized Highlighting

The two graphs in Glyph are coordinated through the use of synchronized sequence highlighting, in which the selection of a sequence node will at the same time highlight the rolled out representation in the state graph (Fig. 5). This allows straightforward comparison of participant data at two levels of details: moment-to-moment in the state graph, and full sequence difference in the sequence graph.

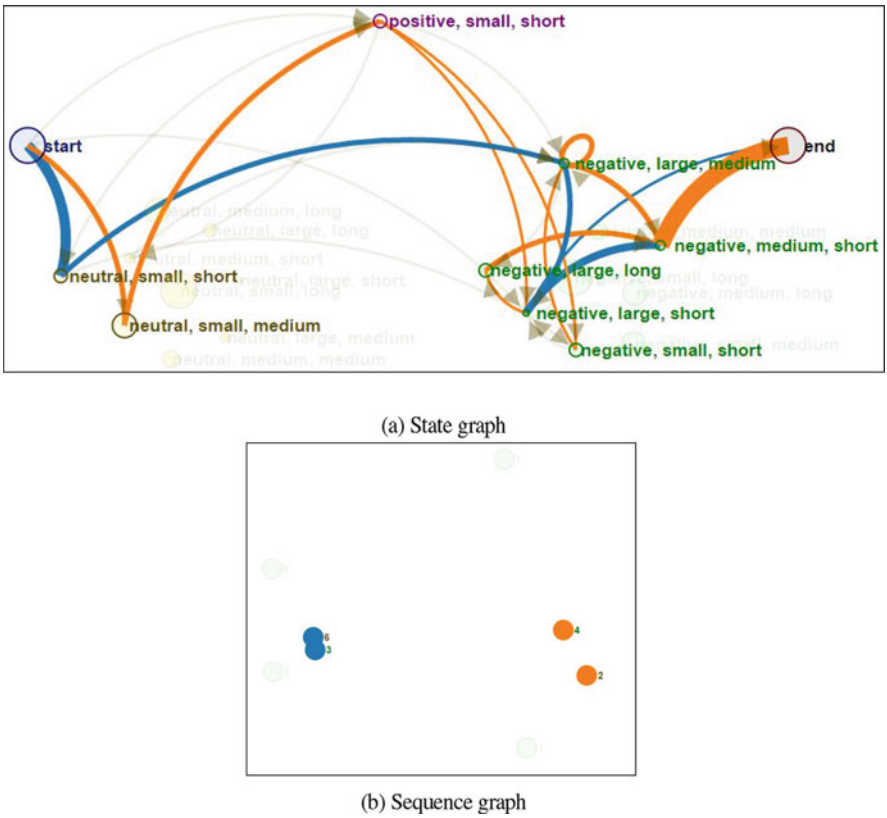


Fig. 5 Coordinated highlighting: selection of sequence nodes in sequence graph (**b**) highlights respective paths in the state graph (**a**) with the same color. The node size (except for the start and end nodes) in the state graph encodes look states' significance values, i.e., the larger the more attentive the participant is

The coupling of state and sequence graphs through visual coordination facilitates three important cognition tasks:

1. *Detection of attentional patterns:* The sequence graph allows quick detection of common patterns, recognized as groups of sequence nodes in close proximity. By selecting a group of similar nodes, users can examine all fixations and states involved in the state graph to come up with hypotheses about this group's behavior.
2. *Detection of unique behavior:* Isolated nodes in the sequence graph represent paths that are significantly different from the population. Examining corresponding paths in the state graph helps user gain in-sights on what happened and thus hypothesize on why.
3. *Comparison of behavior:* Examining vastly different or similar paths (shown as nodes far apart or close by in the sequence graph) in the state graph allows users to understand the nuanced differences between them.

5 Results

As reflected by other researchers on the team when using Glyph, the visualization's most useful feature was the ability to highlight common watching patterns through the video, thereby showing similarity between participants. This feature of the tool is currently used to study differences among subject age and treatment groups in terms of within-group common visual routes. Some use cases of the visualization are demonstrated below.

5.1 Comparison of Participants in Different Age Groups

Using synchronized highlighting, we can color code participants in different age groups with different colors. Figure 6 shows the traces of participants in two groups: middle-aged (blue) and old adults (orange) in the treatment population, i.e., those instructed to stay positive. Notice that in the sequence graph (Fig. 6b), we can observe that the middle aged adults' sequences appear to be somewhat more coherent than the old participants, i.e., respective sequence nodes are more cluttered. Moreover, all of them spent less time looking at the positive AOIs than those in the old group do (Fig. 6a), as evident from the fact that all of them visited the look state *<positive, small, short>*, while some individual in the old group visited the look state *<positive, small, medium>*.

With a small sample size, we cannot generalize these observations to form a hypothesis, but given a larger population, such observations will allow us to construct hypotheses on the general behavior of different age groups, which can then be validated through statistical testing.

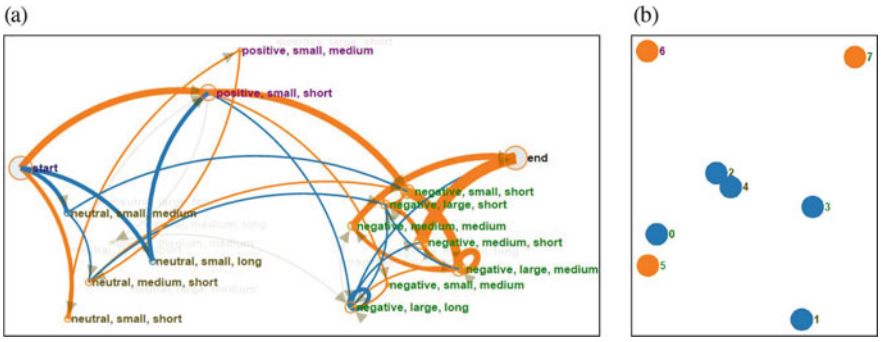


Fig. 6 Highlighting of middle-aged (*blue*) and old adult groups (*orange*) in the treatment population, i.e., instructed to stay positive. (**a**) State graph. (**b**) Sequence graph

5.2 *Comparison of Participants in Different Conditioned Groups*

Another way that Glyph has been used is to compare the look state patterns of the same age groups in different conditioned populations (i.e., treatment versus control) to observe whether there is any discrepancy in how, for example, old people behave when given, as compared to without, instructions.

Figure 7 shows the behavior differences of old adults in the two conditioned groups, suggesting that the treatment group (i.e., instructed to stay happy) appeared to have a more diverse behavior, judging from the state graphs. In particular, the state graph of old adults in control group (Fig. 7a) shows that the number of look states exhibited is smaller than that in the state graph of treatment group (Fig. 7b). This could be a manifestation of the effect of instructions on participants' behavior, which will need to be validated using statistical testing, should we decide this is an observation of significance.

5.3 *Suggestions*

Having tried Glyph, the researchers learnt that this system would benefit from a slightly altered experiment setup. Currently, the clips used in our study are composed of segments of one single emotion (i.e., the choices are vertical). For instance, "Marley and Me" clip shows segments of positive, neutral, or negative emotion but not all of them at a time. Therefore, participants do not have complete freedom in selecting what they want to watch within a video clip. They can only choose to pay more, less, or none at all, attention at each frame. Ideally, if we have clips that comprise of more emotions mixed together with-in a single frame or segment, subjects will be granted more freedom in selecting the region of interest

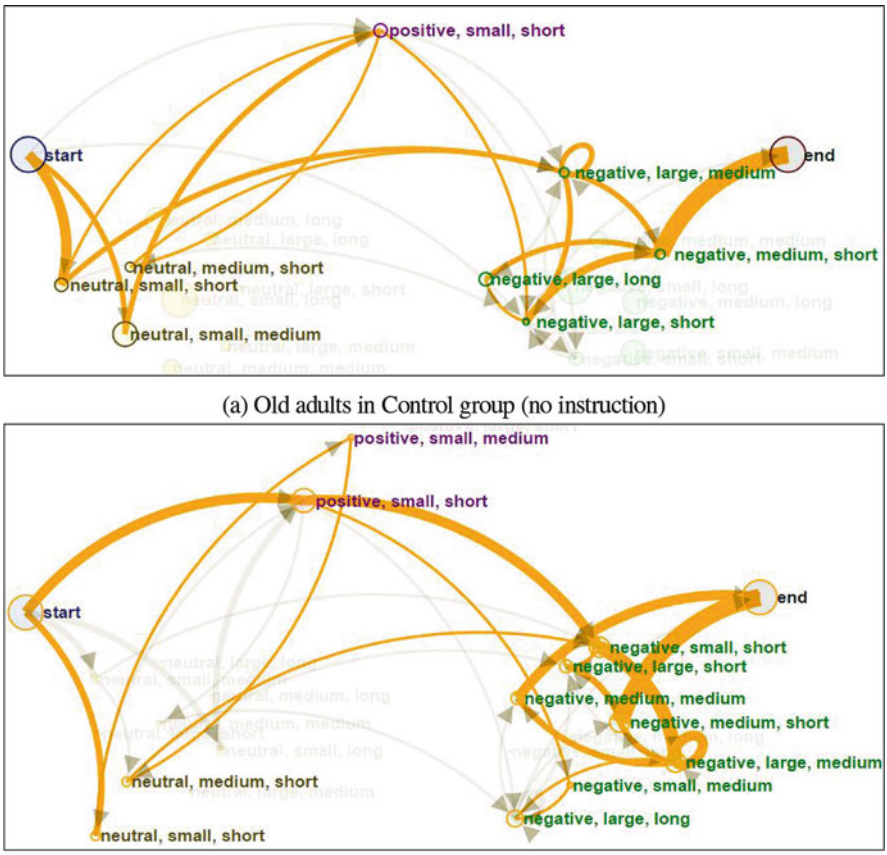


Fig. 7 Look state sequences highlighted for old adults in (**a**) control group, and (**b**) treatment group. Notice that the variety of look states exhibited in the treatment group is higher than that in the control group

at any point of time. For example, if a scene shows at the same time a sad event in one corner of the screen and a happy event in the other, tracking the eye movement of the subject will better inform us about their choices, i.e., whether the subject focuses more on the happy area or the sad area in that clip segment. In such case, the tool would help researchers understand subjects' decisions in the context of multiple alternatives, i.e., horizontally.

Future work would entail more analysis with the current system and utilizing it more in the analysis process within psychological experiments (the described study is still ongoing). We also aim to integrate this system within other eye-tracking experiments to see how well it generalizes and also develop it for better flexibility to allow best utility given the divergent eye-tracking research questions.

6 Conclusion

In this paper, we described a novel approach in analyzing eye-tracking data using a visualization system we developed called Glyph, which facilitated our research goals of uncovering the patterns in emotion regulation when facing stimuli of different affective values in video format. The process includes an abstraction phase where raw eye fixation data are projected into an abstract state space that captures the attention features pertinent to our research questions. Since Glyph's users have freedom in defining a data abstraction of interest, this visualization method is extremely flexible and could handle any type of sequential behavior data. Next, the data is visualized in two graph views, namely state and sequence graphs, which display the data in two forms: state sequences and aggregated representations. Using coordinated highlighting to synchronize the content presented, the final system aims to facilitate user interactions to complete three cognitive tasks: detection of attentional patterns, detection of unique behavior, and comparing behavior sequences. The preliminary assessment of the system with the researchers demonstrated the prospect of the system. As this proposed method is still in its infancy state, we hope to continue developing it in many eye-tracking experiments and would welcome more studies trying this approach to examine temporal patterns of attention. A prototype of the Glyph system is currently hosted at: https://truonghuy.github.io/eyetracking/. Requests to apply this method in eye-tracking studies can be directed to any team member via the contact information listed on the mentioned website.

Acknowledgements This work was supported in part by NIA grant R21 AG044961.

References

1. Anderson, N.C., Anderson, F., Kingstone, A., Bischof, W.F.: A comparison of scanpath comparison methods. Behav. Res. Methods (2014). doi:10.3758/s13428-014-0550-3
2. Atkins, M.S., Jiang, X., Tien, G., Zheng, B.: Saccadic delays on targets while watching videos. In: Proceedings of the Symposium on Eye Tracking Research and Applications (ETRA'12), Santa Barbara, p. 405 (2012). doi:10.1145/2168556.2168648
3. Berg, D.J., Boehnke, S.E., Marino, R.A., Munoz, D.P., Itti, L.: Free viewing of dynamic stimuli by humans and monkeys. J. Vis. **9**(5), 19.1–15 (2009). doi:10.1167/9.5.19
4. Berndt, D.J., Clifford, J.: Using dynamic time warping to find patterns in time series. In: Proceedings of KDD'94: AAAI Workshop on Knowledge Discovery in Databases, Seattle, vol. 10, pp. 359–370 (1994)
5. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: State-of-the-art of visualization for eye tracking data. In: Eurographics Conference on Visualization (EuroVis), Swansea (2014)
6. Blascheck, T., Raschke, M., Ertl, T.: Circular heat map transition diagram. In: Proceedings of the 2013 Conference on Eye Tracking South Africa (ETSA'13). ACM, New York, pp. 58–61 (2013). doi:10.1145/2509315.2509326
7. Bojko, A.: Informative or Misleading? Heatmaps Deconstructed. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in

Bioinformatics), vol. 5610, pp. 30–39 (2009). doi:10.1007/978-3-642-02574-7_4
8. Brasel, S.A., Gips, J.: Points of view: where do we look when we watch TV? Perception **37**(12), 1890–1894 (2008). doi:10.1068/p6253
9. Clement, J.: Visual influence on in-store buying decisions: an eye-track experiment on the visual influence of packaging design. J. Mark. Manag. **23**(9–10), 917–928 (2007). doi:10.1362/026725707X250395
10. Dorr, M., Martinetz, T., Gegenfurtner, K.R., Barth, E.: Variability of eye movements when viewing dynamic natural scenes. J. Vis. **10**(10), 28 (2010). doi:10.1167/10.10.28
11. Duchowski, A.T., McCormick, B.H.: Gaze-contingent video resolution degradation. Hum. Vis. Electron. Imaging III **3299**, 318–329 (1998). doi:10.1117/12.320122
12. Duchowski, A.T., Price, M.M., Meyer, M., Orero, P.: Aggregate gaze visualization with real-time heatmaps. In: Proceedings of the Symposium on Eye Tracking Research and Applications (ETRA'12). ACM, New York, p. 13 (2012). doi:10.1145/2168556.2168558
13. Goldberg, J., Helfman, J.: Visual scanpath representation. In: Proceedings of the 2010 Symposium on Eye-Tracking Research and Applications, Austin, pp. 203–210 (2010). doi:10.1145/1743666.1743717
14. Grindinger, T., Duchowski, A.T., Sawyer, M.: Group-wise similarity and classification of aggregate scanpaths. In: Eye Tracking Research & Applications (ETRA) Symposium, Austin, pp. 101–104 (2010). doi:10.1145/1743666.1743691
15. Gross, J.J.: The emerging field of emotion regulation: an integrative review. Rev. Gen. Psychol. **2**(5), 271–299 (1998). doi:10.1037/1089-2680.2.3.271
16. Havre, S., Hetzler, E., Whitney, P., Nowell, L.: ThemeRiver: visualizing thematic changes in large document collections. IEEE Trans. Visual. Comput. Graph. **8**(1), 9–20 (2002). doi:10.1109/2945.981848
17. Hennessey, C., Fiset, J.: Long range eye tracking: bringing eye tracking into the living room.In: Proceedings of the Symposium on Eye Tracking Research and Applications (ETRA'12), Santa Barbara, pp. 249–252 (2012). DOI:10.1145/2168556.2168608
18. Hervet, G., Guérard, K., Tremblay, S., Chtourou, M.S.: Is banner blindness genuine? Eye tracking internet text advertising. Appl. Cognit. Psychol. **25**, 708–716 (2011). doi:10.1002/acp.1742
19. Holmqvist, K., Holsanova, J., Barthelson, M., Lundqvist, D.: Reading or scanning? A study of newspaper and net paper reading. In: Hyönä, J., Radach, R., Deubel, H. (eds.) The Mind's Eye, pp. 657–670. Elsevier Science BV, Amsterdam, The Netherlands (2003)
20. Hurter, C., Ersoy, O., Fabrikant, S.I., Klein, T.R., Telea, A.C.: Bundled visualization of dynamic graph and trail data. IEEE Trans. Visual. Comput. Graph. **20**(8), 1141–1157 (2014). doi:10.1109/TVCG.2013.246
21. Isaacowitz, D.M.: Mood regulation in real time: age differences in the role of looking. Curr. Dir. Psychol. Sci. **21**, 237–242 (2012). doi:10.1177/0963721412448651
22. Isaacowitz, D.M., Wadlinger, H.A., Goren, D., Wilson, H.R.: Selective preference in visual fixation away from negative images in old age? An eye-tracking study. Psychol. Aging **21**(1), 40–48 (2006). doi:10.1037/0882-7974.21.2.221
23. Jacob, R.J.K., Karn, K.S.: Eye tracking in human computer interaction and usability research: ready to deliver the promises. In: The Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research, pp. 573–605, Amsterdam, Boston (2003). doi:10.1016/B978-044451020-4/50031-1
24. Konstantopoulos, P., Chapman, P., Crundall, D.: Driver's visual attention as a function of driving experience and visibility. Using a driving simulator to explore drivers' eye movements in day, night and rain driving. Accid. Anal. Prev. **42**(3), 827–34 (2010). doi:10.1016/j.aap.2009.09.022
25. Kurzhals, K., Weiskopf, D.: Space-time visual analytics of eye-tracking data for dynamic stimuli. IEEE Trans. Visual. Comput. Graph. **19**(12), 2129–2138 (2013). doi:10.1109/TVCG.2013.194
26. Lankford, C.: Gazetracker: software designed to facilitate eye movement analysis. In: Proceedings of the Symposium on Eye Tracking Research & Applications (ETRA'00), Palm Beach Gardens, pp. 51–55 (2000). doi:10.1145/355017.355025

27. Lethaus, F., Rataj, J.: Do eye movements reflect driving manoeuvres? IET Intell. Transp. Syst. **1**(3), 199 (2007). doi:10.1049/iet-its:20060058
28. Mat Zain, N., Abdul Razak, F., Jaafar, A., Zulkipli, M.: Eye tracking in educational games environment: evaluating user interface design through eye tracking patterns. In: Visual Informatics: Sustaining Research and Innovations, Selangor, vol. 7067, pp. 64–73 (2011). doi:10.1007/978-3-642-25200-6_7
29. Mital, P.K., Smith, T.J., Hill, R.L., Henderson, J.M.: Clustering of gaze during dynamic scene viewing is predicted by motion. Cognit. Comput. **3**(1), 5–24 (2011). doi:10.1007/s12559-010-9074-z
30. Nguyen, T.H.D., Seif El-Nasr, M., Canossa, A.: Glyph: visualization tool for understanding problem solving strategies in puzzle games. In: Foundations of Digital Games (FDG), Pacific Grove (2015)
31. Pieters, R., Wedel, M.: A review of eye-tracking in marketing research. In: Review of Marketing Research, pp. 123–147 (2008). http://dx.doi.org/10.1108/S1548-6435(2008)0000004009
32. Räihä, K.j., Aula, A., Majaranta, P., Rantala, H., Koivunen, K.: Static visualization of temporal eye-tracking data. In: IFIP International Federation for Information Processing, pp. 946–949. Springer, New york (2005). doi:10.1007/11555261_76
33. Reed, A.E., Carstensen, L.L.: The theory behind the age-related positivity effect. Front. Psychol. **3**(SEP) (2012). doi:10.3389/fpsyg.2012.00339
34. Schmid, P.C., Mast, M.S., Bombari, D., Mast, F.W., Lobmaier, J.S.: How mood states affect information processing during facial emotion recognition: an eye tracking study. Swiss J. Psychol. **70**(4), 223–231 (2011). doi:10.1024/1421-0185/a000060
35. Smith, T.J., Mital, P.K.: Attentional synchrony and the influence of viewing task on gaze behavior in static and dynamic scenes. J. Vis. **13**(8) (2013). http://www.ncbi.nlm.nih.gov/pubmed/23863509
36. Stellmach, S., Nacke, L.E., Dachselt, R.: Advanced gaze visualizations for three-dimensional virtual environments. In: Proceedings of the 2010 Symposium on Eyetracking Research Applications, Austin, pp. 109–112 (2010). doi:10.1145/1743666.1743693. http://portal.acm.org/citation.cfm?doid=1743666.1743693
37. Tory, M., Atkins, M.S., Kirkpatrick, A.E., Nicolaou, M., Yang, G.Z.: Eyegaze analysis of displays with combined 2D and 3D views. In: Proceedings of the IEEE Visualization Conference, Minneapolis, p. 66 (2005). doi:10.1109/VIS.2005.37
38. Wagner, R.A., Fischer, M.J.: The string-to-string correction problem (1974). doi:10.1145/321796.321811
39. Wedel, M., Pieters, R.: Eye tracking for visual marketing. Found. Trends® Mark. **1**(4), 231–320 (2006). doi:10.1561/1700000011

The VERP Explorer: A Tool for Exploring Eye Movements of Visual-Cognitive Tasks Using Recurrence Plots

Çağatay Demiralp, Jesse Cirimele, Jeffrey Heer, and Stuart K. Card

Abstract Eye movement based analysis is becoming ever prevalent across domains with the commoditization of eye-tracking hardware. Eye-tracking datasets are, however, often complex and difficult to interpret and map to higher-level visual-cognitive behavior. Practitioners using eye tracking need tools to explore, characterize and quantify patterned structures in eye movements. In this paper, we introduce the VERP (Visualization of Eye movements with Recurrence Plots) Explorer, an interactive visual analysis tool for exploring eye movements during visual-cognitive tasks. The VERP Explorer couples conventional visualizations of eye movements with recurrence plots that reveal patterns of revisitation over time. We apply the VERP Explorer to the domain of medical checklist design, analyzing eye movements of doctors searching for information in checklists under time pressure.

1 Introduction

Eye tracking has been increasingly popular in diverse application areas ranging from neuroscience to marketing due to reduced cost and improved quality in data collection. What makes eye tracking attractive in such a wide range of domains is that eye movements often provide an objective signature for visual-cognitive behavior by tracking the sequential attention of users. However, understanding eye movement trajectories is not straightforward. Practitioners in application domains need tools that facilitate not just the interactive exploration of eye movements but

Ç. Demiralp (✉)
IBM Research, Yorktown Heights, NY, USA
e-mail: cagatay.demiralp@us.ibm.com

J. Cirimele
Tangible Play, Inc., Palo Alto, CA, USA,

J. Heer
University of Washington, Seattle, WA, USA

S.K. Card
Stanford University, Stanford, CA, USA

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_3

also the qualitative and quantitative characterization of patterned structures in eye movements that can be associated with higher-level behavior.

We introduce the VERP (Visualization of Eye movements with Recurrence Plots) Explorer to support the interactive visual and quantitative analysis of eye movements. The VERP Explorer integrates recurrence plots and recurrence based analysis with several spatial eye movement visualizations such as scatter plots, heat maps, gaze plots and alpha patches. We apply the VERP Explorer to evaluate medical checklist designs based on eye movements of doctors searching for information in the checklists to answer a question.

1.1 Visualizing Eye Movements

Advances in eye-tracking technology have made eye movement data collection more practical than ever, increasing the need for developing better visual analysis methods [4]. There are several standard techniques for visualizing eye-tracking data including heat maps, focus maps, and gaze plots (scan paths). Understanding differences and similarities in eye movements across subjects is an important goal in eye-tracking studies. Earlier research introduces several techniques to reduce visual clutter and support multi-subject comparisons (e.g., [8, 24, 25, 30]). Experts often capture semantics of eye movements by tagging areas of interest (AOIs) on the stimulus and associating them with fixations. Typically borrowing from the text visualization literature (e.g., [19, 35]), prior work also proposes visualization techniques to support AOI-based analysis (e.g., [7, 33]).

1.2 Recurrence Plots

Recurrence plots originate from the study of dynamical systems and were introduced for visual analysis of trajectories [14, 27]. Figure 1 shows the recurrence plots for the Lorenz (left) and sine (right) functions. Notice that the Lorenz function is a multidimensional function parametrized by time. To obtain the matrix $[r_{ij}]$ that is

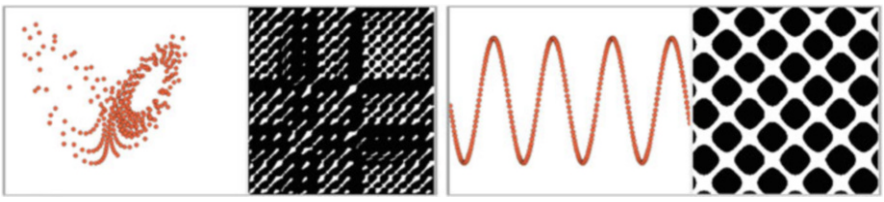


Fig. 1 Recurrence plots of the Lorenz function (*left*)—projected into the plane—and the sine function (*right*). (**a**) Lorenz. (**b**) Sine

the basis for a recurrence plot, we compare each data value f_i (e.g., the eye-tracking sample or value of a time-varying function at time figures/fig*c.pdf*) to all the other values in the sequence, including itself. If the distance d_{ij} between the two compared values is within some small distance ϵ, then we put a 1 at that position in the matrix, otherwise a 0. Formally,

$$r_{ij} = \begin{cases} 1 & \text{if } d_{ij} \leq \epsilon \\ 0 & \text{otherwise} \end{cases}$$

Recurrence plots are essentially thresholded self similarity matrices, where what constitutes to be similar is regulated through choices of the distance measure and the epsilon threshold. The VERP Explorer enables users to create recurrences plots of raw eye movements with ability to dynamically change the values of these two parameters.

Figure 2 illustrates how a recurrence graph for eye movements (Fig. 2a) is constructed. The dotted circles represent the ϵ-distance regions around the locations of eye movements. To analyze a visual text search task, for example, we would set ϵ to be 1.5°, as the radius of the foveal circle in which a person can read the text is about 1.5° [23]. To construct the recurrence plot Fig. 2g of the eye movements shown in Fig. 2a, we start with a blank matrix Fig. 2b. Eye movement 1 is within its own circle so cell (1, 1) is white (Fig. 2c). Likewise, all other eye movements fall within their own circles, so the diagonal (i, i) is white (Fig. 2d). No other eye movement falls within the circle of eye movement 1, so the rest of row 1 is black Fig. 2e. Since the distance metric used is symmetric, the rest of column 1 is black as well (Fig. 2f). Eye movement 2 is also not quite in any other eye movement's circle, therefore, except for the cell (2, 2) on the diagonal, its row and column are

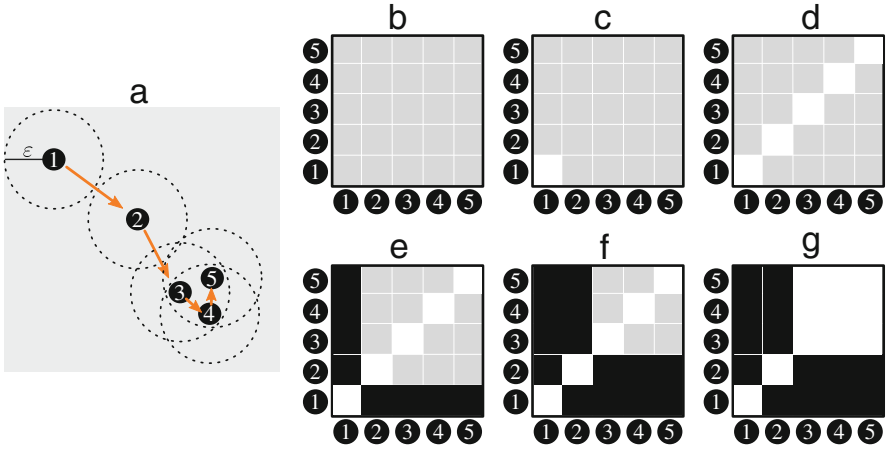


Fig. 2 Construction of a recurrence plot for the eye movements shown in (**a**). The radius of *dotted circles* around the points is ϵ. For every pair of points, we put 1 (*white*) in the corresponding matrix entry if they are within ϵ distance (i.e., their *dotted circles* intersect), otherwise we enter 0 (*black*)

black (Fig. 2f). Eye movement 3 is in the circles of eye movements 4 and 5, so cells (3, 4) and (3, 5) are white, by symmetry, so are cells (4, 3) and (5, 3). Finally, eye movement 4 is in the circle of eye movement 5, so (4, 5) and (5, 4) are also white (Fig. 2g).

Recurrence plots are particularly good at characterizing periodic and semi-periodic sequences in a time series. The recurrence graph of a sine wave shown in Fig. 1, for example, exhibits strong periodic behavior.

Prior work applies recurrence plots to analysis of speaker-listener eye movement coordination [11, 31] and characterization of eye movements in viewing scenes [1, 37]. Facilitating both visual (qualitative) and quantitative analysis is a powerful feature of recurrence plots. Recurrence quantification analysis (RQA) [28] uses scalar descriptors such as Recurrence Rate, Entropy, Determinism, etc. to quantify different recurrence patterns. Anderson et al. [1] apply RQA to characterize the type of the stimulus scene viewed, finding RQA measures to be sensitive to differences between scene types (e.g., indoor vs. outdoor). Building on this work, Wu et al. find that differences in eye movement patterns as quantified by RQA correspond to scene complexity and clutter [37].

To our knowledge, our work is among the first to study the goal-oriented task of visual search using recurrence plots of eye movements. The VERP Explorer simplifies exploratory analysis by integrating spatial eye-tracking visualizations with recurrence plots and quantified recurrence analysis.

2 Design of the VERP Explorer

The goal of the VERP Explorer is to support the interactive visual analysis of eye movements using recurrence plots. To this end, the VERP Explorer couples several spatial eye movement visualizations with recurrence plots through brushing and linking (Fig. 3). The VERP Explorer is a web based application implemented in JavaScript with help of D3 [6], AngularJS [2] and heatmap.js [21] libraries. The source code and a deployed copy of the VERP Explorer can be accessed at https://www.github.com/uwdata/verp/.

We now briefly discuss the visualizations and interactions that the VERP Explorer supports.

2.1 Heat Maps, Focus Maps, and Scatter Plots

The VERP Explorer enables users to visualize eye-tracking positions as heat maps, focus maps, and scatter plots. The three have complementary strengths. *Heat maps* and *focus maps* are two related standard techniques that are useful for providing a synaptic view of eye movements aggregated over time and subjects. The VERP Explorer creates the heat map visualizations by using a Gaussian (radial) blur function with a color gradient (Fig. 4). Users can interactively change the maximum

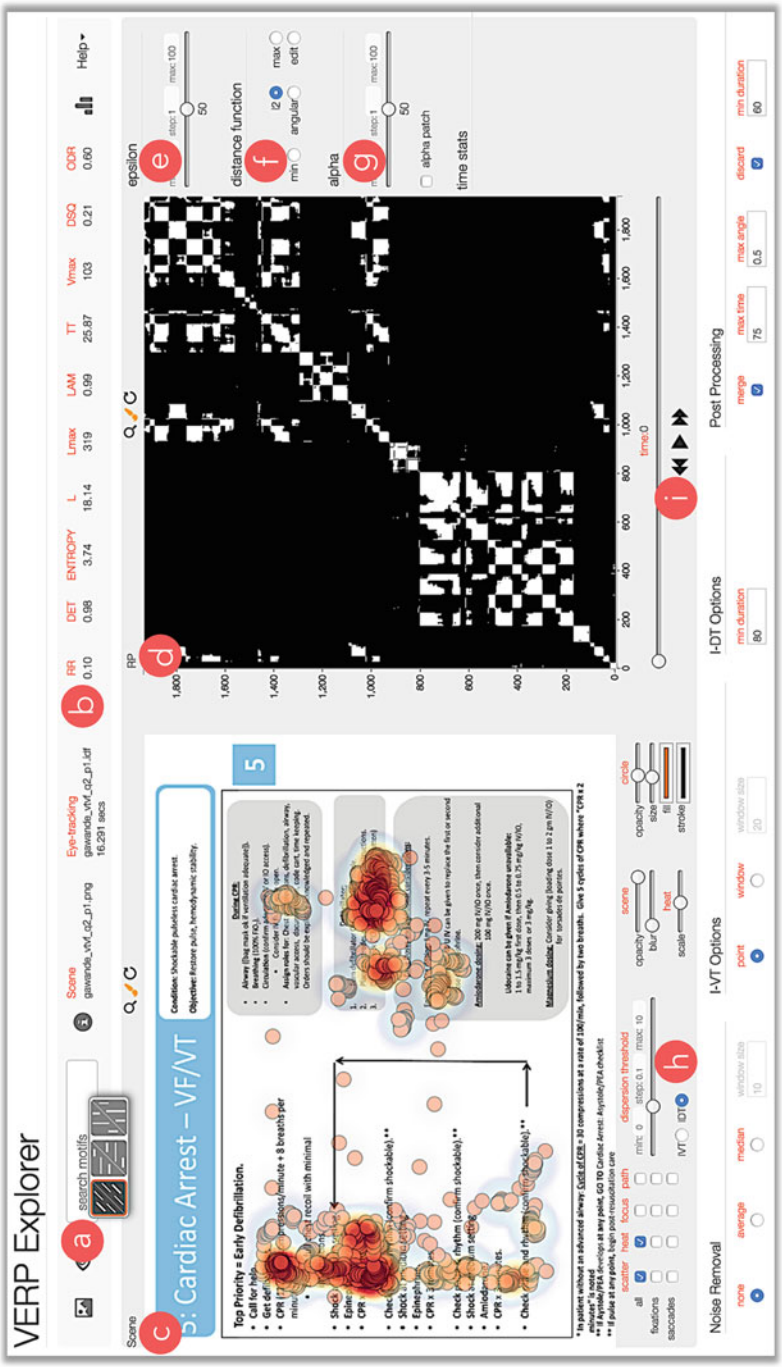


Fig. 3 The VERP Explorer interface has two main views: the scene view (*c*) and the recurrence plot view (*d*). The VERP Explorer combines spatial eye movement visualizations with recurrence plots to support the visual and quantitative analysis of eye movements (See Sect. 2 for descriptions of the interface elements labeled above)

Fig. 4 Three spatial eye-tracking visualizations from the VERP Explorer: Heat map (*left*), focus map (*middle*), and scatter plot (*right*). The three visualizations have complementary advantages. Heat maps and focus maps are particularly useful for providing a continuous aggregate view of eye movements and their negative space. Scatter plots directly encode eye movements (as circular nodes here), enabling the exploration of eye-tracking datasets at the level of individual eye movements

value used for normalizing the heat maps. This enables a dynamic adjustment of the color gradient sensitivity. By painting eye movement point densities, heat maps obscure, however, the areas of attention when overlaid on the stimulus image.

Focus maps visually "invert" heat maps to enable the visibility of the areas of viewer attention. To create a focus map, we first create a uniform image (mask) that has the same size as the underlying stimulus image. We then vary the opacity at each pixel inversely proportional to the opacity of the corresponding heat map pixel. Focus maps are essentially negative space representations, visualizing the negative space of the corresponding heat maps (Fig. 4).

Heat maps and focus maps support visual aggregation while visualizing eye movements indirectly. On the other hand, *scatter plots* provide a discrete view by representing eye movement positions directly, enabling the inspection of patterns and outliers at the level of individual eye movements. The VERP Explorer creates scatter plot views by drawing each eye-tracking position as a circular node in the plane (Fig. 4).

2.2 Scan Paths

In their basic, static configuration, neither heat maps nor focus maps convey the temporal order of eye movements. The VERP Explorer uses scan paths (gaze plots) to provide an aggregate temporal view of eye movements. It creates scan path views by drawing circles centered at the centroids of fixation clusters and connecting two consecutive clusters with arrows. The VERP Explorer numbers the nodes sequentially. It also encodes the temporal order of fixations by coloring the nodes and the arrows using a color map ranging from dark blue to red [18] (Fig. 5).

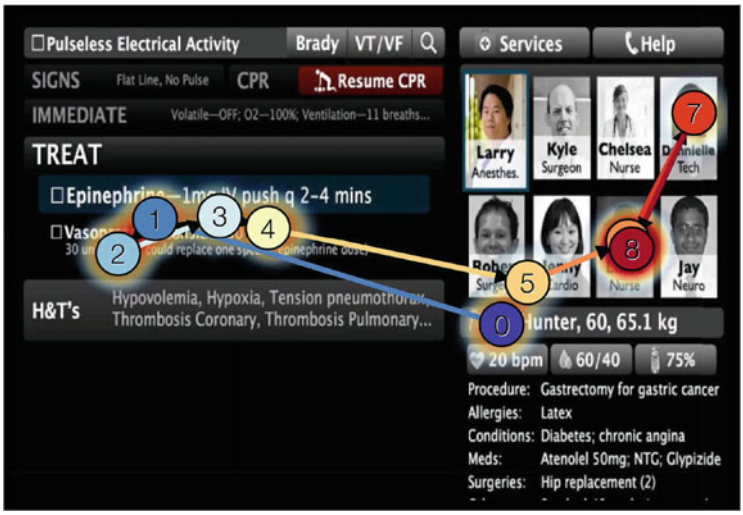


Fig. 5 Scan path visualization of fixation points. The VERP Explorer uses text, shape, and color to encode the temporal order of fixations. It sequentially numbers the nodes that represents fixation clusters, puts an arrow between consecutive nodes, and colors the nodes and the arrows using a color map ranging from *dark blue* to *red*

2.2.1 Identifying Fixations and Saccades

Note that the VERP Explorer does not assume the eye-tracking points are already classified. It provides two different methods for identifying fixations and saccades; velocity-threshold identification (I-VT) and dispersion-threshold identification (I-FT) [32]. I-VT and I-DT are both fast, threshold based algorithms for classifying eye movements into fixations and saccades. I-VT identifies fixations and saccades based on a threshold on point-to-point velocities of eye movements. On other hand, I-DT identifies them using a threshold on spatial dispersion of eye movements. We briefly discuss below our implementation of these two algorithms. See [32] for a comparative discussion of fixation-saccade identification algorithms.

I-VT operates under the assumption that low-velocity eye movements correspond to fixations and high velocities to saccades. Using I-VT, we compute clusters of fixations in three steps. First, we calculate point-to-point velocities for each tracking point. Note that velocities can be computed using spatial or angular distance between consecutive points. We use angular velocities if the head position is provided in the tracking data. We then classify each point as a fixation or saccade using a velocity threshold. If the points velocity is below the threshold, it becomes a fixation point, otherwise it is considered a saccade points. In the final step, we gather consecutive fixation points into clusters (gaze regions).

Due to their low velocities, consecutive fixations have smaller dispersion than consecutive saccadic eye movements. I-DT aims to directly detect fixation clusters using a dispersion threshold on eye-tracking points in a moving window. We start

by placing the window at the first eye-tracking point, spanning a minimum number of points. We then compute the dispersion of the points in the window by adding the width and the height of the bounding box of the points. If the dispersion is above a threshold, the window does not represent a fixation cluster, and we move the window one point to the right. If the dispersion is below the dispersion threshold, the window represents a fixation cluster. In this case, we expand the window to the right as far as its dispersion is below the threshold. We designate the points in the final expanded window as a fixation cluster. We then move the beginning of the window to the point excluded from the last fixation cluster and reset the window size to the minimum number of points. We repeat the above process by moving the window to the right until all the eye-tracking points are processed.

For both algorithms, the VERP Explorer computes measures such as centroid, geometric median, and duration for each found fixation cluster. More importantly, the VERP Explorer enables users to dynamically modify the velocity and dispersion thresholds or tune the parameters of the algorithms (Fig. 3h), while viewing the changing scan path visualizations interactively.

2.3 *Alpha Patches*

Visual clutter is often a concern in analysis of eye-tracking data. We introduce *alpha patches*, alpha shapes [15] of eye movements, to provide a cleaner view of eye-tracking positions through filled polygonal patches.

The alpha shape is a generalization of the convex hull of a point set [15]. Unlike the convex hull, the alpha shape can recover disconnected, non-convex spaces with holes. Crucially, it provides a control over the specificity of the polygonal approximation of the underlying points through a parameter $\alpha \in [0, \infty)$ (Fig. 6). The VERP Explorer enables users to automatically create alpha patches of fixations with a dynamic control over the α parameter (Fig. 3g).

Given an eye-tracking point set (e.g., fixations) and an alpha value, we generate the alpha patch for the point set in three steps. First, we create the Delaunay triangulation of the set. Note that the boundary of the Delaunay triangulation is

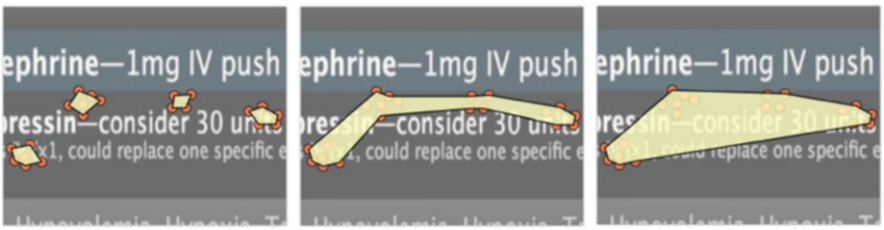


Fig. 6 Four alpha patches with increasing α values from *left to right*. Notice that, when α is sufficiently large, the alpha patch is the convex hull of the points (*right*)

the convex hull of the points in the set. Second, we extract from the Delaunay triangulation the triangles whose vertices are within the alpha distance. The union of the extracted triangles is known as the alpha complex of the point set. In the final step, we determine the boundary of the alpha complex and draw them as simple closed polygons. In our implementation, we create the Delaunay triangulation once and extract alpha complexes for varying—user determined—alpha values as needed.

2.4 Interaction Techniques

The visualizations we have described are interactive, giving rise to a number of exploration techniques:

Zooming & Panning: The VERP Explorer provides zooming and panning interactions on all of the visualizations that it generates. Both zooming and panning are forms of dynamic visual filtering and essential for exploring dense eye movement datasets.

Brushing & Linking: We use brushing & linking in the VERP Explorer to coordinate the scatter plot of the eye-tracking data with the recurrence plot view. This is the main mechanism that allows users to inspect recurrence space and spatial eye movements simultaneously. Brushing over a location on the scene highlights all the corresponding entries in the recurrence view. Conversely, brushing on the recurrence plot highlights corresponding eye movement positions represented as circular scatter plot nodes. Brushing regions can be resized or moved using mouse as well as keyboard.

Epsilon Filtering: Epsilon filtering enables the interactive exploration of epsilon values for recurrence plots (Fig. 3e). Users can also select different distance measures (Fig. 3f). We provide the Euclidean (L_2-Norm), the city block (L_1-Norm), the maximum (L_∞-Norm) and the minimum of the absolute differences along data dimensions. In addition to these general distance measures, users can select eye movement specific distances, including the angular distance and edit distance (to be used if eye movements are associated with textual tags).

Alpha Filtering: Similar to epsilon filtering, alpha filtering allows users to dynamically change the α parameter of the alpha patches. This enables a control over the precision of the polygonal representation for the underlying eye movements (Fig. 3g).

Dynamic Fixation-Saccade Classification: The VERP Explorer also enables users to change the threshold for fixation-saccade classification dynamically. This is particularly useful when angular velocity calculations are not possible or reliable (Fig. 3h).

Motif Search and Quantification: Recurrence plots facilitate pattern-based analysis of time varying data. One of the motivations of the current work is to help relate

behavioral eye movement patterns to visual design through recurrence patterns. The VERP Explorer computes (Fig. 3b) several recurrence quantification measures such as Recurrence Rate (RR) and Determinism (DET), Entropy (ENTROPY), etc. (See [28] for detailed discussion of recurrence quantification measures.) In addition, VERP Explorer enables the search for predefined, arbitrary patterns in the recurrence plots (Fig. 3a). Currently users can search for diagonal, vertical and horizontal recurrence structures.

Timeline Animation: While the scan path visualization provides an aggregated temporal view of the eye movement, it is desirable to be able to directly examine the timeline of the complete data. The VERP Explorer enables users to animate the appearance of eye-tracking points using the scatter plot visualization (Fig. 3i).

3 Illustration of Use: Visual Search in Emergency Medical Checklists

The purpose of visualization is to ease and amplify the work of cognition by re-coding information so as to exploit the perceptual abilities of the eye. To design for the eye, we have principles at a general level—the principles of perception, the gestalt laws, etc.—but to gain more insight, we need to understand the lower-level mechanisms forming these principles. Insights and models derived from lower-level empirical data can inform higher-level visual design principles [13]. Fortunately, eye movements often track the sequential attention of the user, affording a unique window into visual-cognitive interactions. Analyzing eye movement patterns can therefore provide useful insights into the effectiveness of a visual design.

To illustrate the use of the VERP Explorer for exploring a cognitive-visual task, we use the task of designing visual displays for emergency medical checklists. In U.S. hospitals, it is estimated that medical errors cause in excess of 100,000 deaths per year, half of which are thought to be preventable [22]. Checklist use has been found to improve performance in aviation [5, 9, 12] and medicine from surgery to intensive care and crisis response [3, 16, 17, 20, 26, 29, 38]. However, checklists have been criticized for adding delay, attentional load, and complexity [16, 36], slowing down crucial medical procedures. As Verdaasdonk et al. [34] put it, "Time governs willingness and compliance in the use of checklists." It would therefore be desirable to improve the speed (and accuracy) with which aids can be used.

3.1 Comparing Two Checklist Formats

We compare two checklist designs. The first design is from the World Health Organization ("Standard") and is an example of current best practice [22]. The second is a dynamic format ("Dynamic") for which the current checklist step

is enlarged and more distant steps shrunk or hidden [10]. For purposes of the illustration, we consider data only from five participants (doctors) collected while they were searching the checklist to answer a single question: *What is the correct dose of atropine?* We used an eye tracker that is accurate to approximately 0.5 to 1 deg of arc.

To start, we compute the average time to answer the question with each checklist format. The result is that the Dynamic checklist format is 32 % faster than the Standard format. But we would like more insight into why. We therefore analyze the eye movement data with the VERP Explorer. We load the image of the checklist and the eye movement data files into the VERP Explorer. At this point, the many controls of the VERP Explorer allow us to tailor an analysis to our interests.

Figure 7 shows the screenshots from the VERP Explorer for the eye movements of the five doctors. They are arranged in order from the fastest trials to the slowest trials for each format.

Study Squares: The first thing to notice is that the eye movements for searching through text exhibit very different recurrence patterns than the semi-periodic function applications in Fig. 1 investigated in the earlier literature. The recurrence plots of visual text search consist mainly of square patterns (*Study Squares*) comprised mainly of fixations, separated by subsequences of saccades on the diagonal.

To this basic patterns are added off-diagonal lines and squares representing regressive re-viewing of previously seen parts of the display (i.e., cycles). The Study Squares come about as in Fig. 2 from a group of eye fixation points in close proximity, that is, exhibiting locality of reference. The more intensively some part of the scene is looked at, the larger the size of the square. Some squares have a checkerboard character, indicating that the doctor shifted her gaze to another part of the scene and then back. Searches taking more time often appear more scattered, reflecting the disorganization of the search. The brushing tools provided with VERP allow us to discover where square motifs on the recurrence plot are located in the scene.

Inadvertent Detractors: We have concentrated on the general patterns in the eye movement, but since the eyes are controlled both in reaction to visual stimuli as well as cognitively in service of a goal, we also discover unexpected details. Such was the case with these analyses. In four out of five of the Dynamic format screen shots in Fig. 7, the eye has been attracted to the pictures of doctors attending. This features was included in the format, because it is often the case that attending medical personnel do not know each others names, which in turn makes it difficult to address direct requests to a named individual—an important element of disciplined coordination to prevent requesting participant from thinking some task has been done, whereas no one actually accepted responsibility for doing it. It did not occur to the designers of this format that the high contrast of the picture to the dark background would interfere with the acquisition of information in the checklist by inadvertently attracting the eye, although this is obvious once it is pointed out. This is a type of problem that can remain invisible and decrease user performance despite a basically sound design. The VERP Explorer enabled us to find this problem easily.

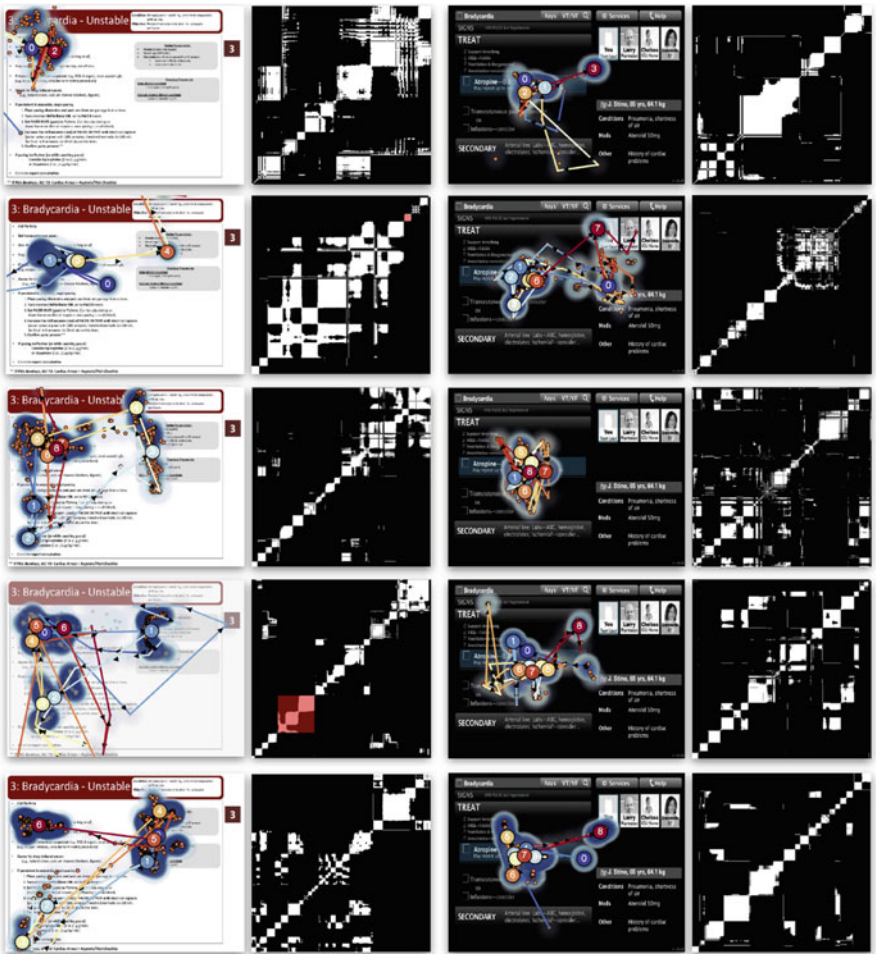


Fig. 7 Analysis eye movements of five participants using two checklist designs. *Left pair of columns* shows eye movements along with gaze plots and recurrence plots for Standard emergency checklist. *Right pair of columns* shows Dynamic checklist. Both are arranged from fastest on *top to slowes*t. Generally there are more of them and they are less organized for slower trials.

4 Discussion and Conclusion

Eye movement based analysis provides a unique opportunity for evaluating the effectiveness of a visual design. Eye movements are, however, lower-level manifestations of visual-cognitive interactions that need to be mapped to the behavior the designer usually needs.

We developed the VERP Explorer to support interactive visual and quantitative analysis of eye-tracking datasets using recurrence plots. The VERP Explorer is an open source web application available at https://www.github.com/uwdata/verp/. We applied it in comparing medical checklist designs based on eye movements of doctors searching for information in the checklists. We focused on visual search task and characterize eye movements of visual search through recurrence plots and other visualizations provided by the VERP Explorer.

Eye-tracking hardware is becoming a commodity with ever expanding range of applications. To better utilize the increasingly ubiquitous eye-tracking data, we need tools to better map patterned structures of eye movements to visual-cognitive behavior in application domains. The VERP Explorer is a contribution to our toolbox for that end.

References

1. Anderson, N.C., Bischof, W., Laidlaw, K., Risko, E., Kingstone, A.: Recurrence quantification analysis of eye movements. Behav. Res. Methods **45**(3):842–856 (2013)
2. AngularJS. http://angularjs.org/
3. Arriaga, A.F., Bader, A.M., et al.: A simulation-based trial of surgical-crisis checklists. New Engl. J. Med. **368**(15):1459–1460 (2013)
4. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: State-of-the-art of visualization for eye tracking data. In: EuroVis – STARs. The Eurographics Association, Swansea (2014)
5. Boorman, D.: Safety benefits of electronic checklists – an analysis of commercial transport accidents. In: Proceedings of the 11th International Symposium on Aviation Psychology, Columbus (2001)
6. Bostock, M., Ogievetsky, V., Heer, J.: D^3: data-driven documents. IEEE Trans. Visual. Comput. Graph. **17**(12):2301–2309 (2011)
7. Burch, M., Kull, A., Weiskopf, D.: AOI rivers for visualizing dynamic eye gaze frequencies. Comput. Graph. Forum **32**:281–290 (2013)
8. Burch, M., Schmauder, H., Raschke, M., Weiskopf, D.: Saccade plots. In: Proceedings of the Symposium on Eye Tracking Research and Applications (ETRA'14). ACM, New York (2014)
9. Burian, B.K., Barshi, I., Dismukes, K.: The challenge of aviation emergency and abnormal situations. Technical report, NASA Ames Research Center (2005)
10. Cirimele, J., Wu, L., Leach, K., Card, S.K., Harrison, T.K., Chu, L., Klemmer, S., RapidRead: step-at-a-glance crisis checklists. In: Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare, Oldenburg (2014)
11. Dale, R., Warlaumont, A.S., Richardson, D.C.: Nominal cross recurrence as a generalized lag sequential analysis for behavioral streams. Int. J. Bifurcation and Chaos **21**(04):1153–1161 (2011)
12. Degani, A., Wiener, E.L.: Human factors of flight-deck checklists: the normal checklist. Technical report, NASA Ames Research Center (1990)
13. Demiralp, Ç., Bernstein, M., Heer, J.: Learning perceptual kernels for visualization design. IEEE Trans. Visual. Comput. Graph. (Proc. InfoVis) **20**(12), 1933–1942 (2014)
14. Eckmann, J.-P., Kamphorst, S.O., Ruelle, D.: Recurrence plots of dynamical systems. EPL (Eur. Lett.) **4**(9), 973 (1987)
15. Edelsbrunner, H., Mcke, E.P.: Three-dimensional alpha shapes. ACM Trans. Graph. **13**(1), 43–72 (1994)

16. Gawande, A.: The Checklist Manifesto: How to Get Things Right. Metropolitan Books, New York (2009)
17. Harrison, T.K., Manser, T., Howard, S.K., Gaba, D.M.: Use of cognitive aids in a simulated anesthetic crisis. Anesthe. Analge. **103**(3), 551–556 (2006)
18. Harrower, M., Brewer, C.A.: ColorBrewer.org: an online tool for selecting colour schemes for maps. Cartogr. J. **40**(1), 27–37 (2003)
19. Havre, S., Hetzler, B., Nowell L.: ThemeRiver: visualizing theme changes over time. In: IEEE Proceedings of InfoVis'00, Salt Lake City (2000)
20. Haynes, A., Weiser, T., Berry, W., Lipsitz, S., Breizat, A.-H., Dellinger, E., Herbosa, T., Joseph, S., Kibatala, P., Lapitan, M., Merry, A., Moorthy, K., Reznick, R., Taylor, B., Gawande, A.: A surgical safety checklist to reduce morbidity and mortality in a global population. New Eng. J. Med. **360**, 491–499 (2009)
21. heatmap.js. https://www.patrick-wied.at/static/heatmapjs/
22. James, J.T.: A new, evidence-based estimate of patient harms associated with hospital care. J. Patient Saf. **9**(3), 122–128 (2013)
23. Kieras, D.E., Hornof, A.J.: Towards accurate and practical predictive models of active-vision-based visual search. In: Proceedings of CHI'14, pp. 3875–3884. ACM, New York (2014)
24. Kurzhals, K., Weiskopf, D.: Space-time visual analytics of eye-tracking data for dynamic stimuli. IEEE Trans. Visual. Comput. Graph. **19**(12), 2129–2138 (2013)
25. Li, X., Çöltekin, A., Kraak, M.-J.: Visual exploration of eye movement data using the space-time-cube. In: Fabrikant, S., Reichenbacher, T., van Kreveld, M., Schlieder, C. (eds) Geographic Information Science, vol. 6292 of LNCS, pp. 295–309. Springer, Berlin/Heidelberg (2010)
26. Makary, M.A., Holzmueller, C.G., Thompson, D., Rowen, L., Heitmiller, E.S., Maley, W.R., Black, J.H., Stegner, K., Freischlag, J.A., Ulatowski, J.A., Pronovost, P.J. Operating room briefings: working on the same page. Jt. Comm. J. Qual. Patient Saf./Jt. Comm. Res. **32**(6), 351355 (2006)
27. Marwan, N.: A historical review of recurrence plots. Eur. Phys. J. Spec. Top. **164**(1), 3–12 (2008)
28. Marwan, N., Romano, M.C., Thiel, M., Kurths, J.: Recurrence plots for the analysis of complex systems. Phys. Rep. **438**(56), 237–329 (2007)
29. Pronovost, P., Needham, D., Berenholtz, S., Sinopoli, D., Chu, H., Cosgrove, S., Sexton, B., Hyzy, R., Welsh, R., Roth, G., Bander, J., Kepros, J., Goeschel, C.: An intervention to decrease catheter-related bloodstream infections in the ICU. New Eng. J. Med. **355**(26), 2725–2732 (2006)
30. Raschke, M., Chen, X., Ertl, T.: Parallel scan-path visualization. In: Proceedings of the Symposium on Eye Tracking Research and Applications (ETRA'12). ACM, Austin (2012)
31. Richardson, D.C., Dale, R., Kirkham, N.Z.: The art of conversation is coordination: common ground and the coupling of eye movements during dialogue. Psychol. Sci. **18**(5), 407–413 (2007)
32. Salvucci, D.D., Goldberg, J.H.: Identifying fixations and saccades in eye-tracking protocols. In: Proceedings of ETRA'00. ACM, Palm Beach Gardens (2000)
33. Tsang, H.Y., Tory, M., Swindells, C.: eSeeTrack–visualizing sequential fixation patterns. IEEE Trans. Visual. Comput. Graph. **16**(6), 953–962 (2010)
34. Verdaasdonk, E.G.G., Stassen, L.P.S., Widhiasmara, P.P., Dankelman, J.: Requirements for the design and implementation of checklists for surgical processes. Surg. Endosc. **23**(4), 715–726 (2008)
35. Wattenberg, M., Viegas, F.: The word tree, an interactive visual concordance. IEEE Trans. Visual. Comput. Graph. **14**(6), 1221–1228 (2008)
36. Winters, B.D., Gurses, A.P., Lehmann, H., Sexton, J.B., Rampersad, C., Pronovost, P.J.: Clinical review: checklists – translating evidence into practice. Crit. Care **13**(6), 210 (2009)

37. Wu, D.W.-L., Anderson, N.C., Bischof, W.F., Kingstone, A.: Temporal dynamics of eye movements are related to differences in scene complexity and clutter. J. Vis. **14**(9), 8 (2014)
38. Ziewacz, J.E., Arriaga, A.F., Bader, A.M., Berry, W.R., Edmondson, L., Wong, J.M., Lipsitz, S.R., Hepner, D.L., Peyre, S., Nelson, S., Boorman, D.J., Smink, D.S., Ashley, S.W., Gawande, A.A.: Crisis checklists for the operating room: development and pilot testing. J. Am. Coll. Surg. **213**(2), 212–217 (2011)

Gaze Visualization for Immersive Video

Thomas Löwe, Michael Stengel, Emmy-Charlotte Förster, Steve Grogorick, and Marcus Magnor

Abstract In contrast to traditional video, immersive video allows viewers to interactively control their field of view in a 360° panoramic scene. However, established methods for the comparative evaluation of gaze data for video require that all participants observe the same viewing area. We therefore propose new specialized visualizations and a novel visual analytics framework for the combined analysis of head movement and gaze data. A novel View Similarity visualization highlights viewing areas branching and joining over time, while three additional visualizations provide global and spatial context. These new visualizations, along with established gaze evaluation techniques, allow analysts to investigate the storytelling of immersive videos. We demonstrate the usefulness of our approach using head movement and gaze data recorded for both amateur panoramic videos, as well as professionally composited immersive videos.

1 Introduction

With the emergence of affordable 360° consumer video cameras, immersive video is becoming increasingly popular [10, 20]. Specialized 360° video players allow users to interactively rotate the viewing direction during playback. Alternatively, head-mounted displays (HMDs) can be used to provide deeper immersion and a more natural control scheme, in which the viewing direction is controlled by the rotation of the head. Recently, YouTube launched support for 360° panoramic video, further heightening public interest in the technology [30].

While immersive video has since been used in sports, marketing and also creative filmmaking, efforts to generate knowledge about storytelling in immersive video have only recently emerged [19]. No specialized methods to evaluate the perception and viewing behavior of the viewer have yet been developed. One of the most common approaches to analyze user attention in traditional video is eye tracking. By recording and aggregating gaze data from multiple participants, experts can gain insight into the viewing behavior of users, e.g. how the eye is guided by content.

T. Löwe (✉) • M. Stengel • E.-C. Förster • S. Grogorick • M. Magnor
Computer Graphics Lab, TU Braunschweig, Braunschweig, Germany
e-mail: loewe@cg.cs.tu-bs.de

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_4

However, established visualization techniques for gaze data for video assume that all participants receive the exact same stimulus. This is not the case with immersive video. While all participants are watching the exact same video, each participant is free to choose their individual field of view. Thus, content that occurs outside of this field of view is missed. In order to gain insight into the viewing behaviour for immersive video, both the eye gaze and the head orientation must therefore be considered. Throughout the rest of this chapter we will differentiate between head orientation (*viewing direction*), and eye focus (*gaze direction*).

Storytellers working with immersive video often layer each frame with multiple subplots. They are particularly interested in knowing how many viewers will follow each subplot and understanding which elements of their video may induce a switch between subplots. We therefore focus on *joins* and *branches* between participants' fields of view. Joins occur when the attention of multiple viewers is drawn towards a common direction, causing their fields of view to overlap, whereas branches occur when their fields of view diverge.

We propose a novel View Similarity visualization that illustrates fields of view branching and joining over time. Our proposed visual analytics workflow includes three additional visualizations: A limited view from the viewer's perspective, a 3D sphere-mapped version of the video to provide spatial context, and an unwrapped view of the entire frame to provide global context. All of these views can be additionally overlaid with established gaze visualizations, such as attention maps [16] or scan paths [17], in order to equip experts with a familiar set of analysis tools.

This chapter is organized as follows: Sect. 2 introduces related work. Section 3 describes our proposed visualizations and details the visual analytics workflow. In Sect. 4 we demonstrate the usefulness of the proposed framework using head movement and gaze data we gathered in a user study (Fig. 1). Section 5 concludes this chapter and outlines future work.

Fig. 1 *Left*: A participant watching a 360° video using a head-mounted display. *Right:* The video is mapped to both eyes, putting the observer at the center of the scene

2 Related Work

Eye tracking is an established tool in many fields of research, and has been used to analyze visual attention in several real-world scenarios, including video compression [4, 18], medicine [23], visual inspection training [7], and commercial sea, rail and air vehicle control [11, 12, 31].

Recently, Blascheck et al. presented a comprehensive State-of-the-Art survey of visualization for eye-tracking data [2], citing numerous methods for the visualization of gaze data for traditional video. Among the most common representations for eye-tracking data in video are attention maps [8, 16] and scan paths [17]. However, these methods can not be directly applied to immersive video, where each viewer controls an individual viewing area.

There have been gaze data visualizations that allow participants to individually inspect static 3D scenes in an interactive virtual environment [21, 22, 27, 29]. Here synchronization between participants is achieved by mapping scan paths or attention maps onto the static geometry. However, in immersive video there is no actual 3D geometry, but rather a recorded 360° video that is mapped onto a sphere around the observer. Additionally, the observer's position is fixed to the center of the sphere, since a free view point is not appropriate for immersive video recorded from a fixed camera position. Thus, immersive video falls into the mid-range between traditional video and 3D scenes, and neither approach directly applies. Our framework therefore combines methods from both scenarios using multiple views.

We further reduce the problem of synchronizing participants to finding moments when the attention, i.e. viewing direction, of many users is drawn to a certain region in the video. These moments are also commonly referred to as moments of attentional synchrony [25].

In traditional video, attentional synchrony is also analyzed by monitoring gaze transitions between manually annotated *areas of interest* (AOI) [3, 13, 14]. However, annotating these AOIs is often time-consuming and exhausting. This is particularly true for immersive video, where the unintuitive distortion of popular texture formats (e.g. equirectangular projection) makes selection more difficult, e.g. AOIs moving around the observer will have to wrap around from the right to the left edge of the video frame. Additionally, multiple stories often occur simultaneously in different parts of the video, further increasing the workload for the annotation.

While we believe that AOIs can be beneficial for the evaluation of immersive video, specialized annotation tools would be required to make working with AOIs feasible. Therefore, our approach avoids dependency on manually annotated AOIs and instead gauges attentional synchrony based on the similarity of the individual viewing directions.

3 Workflow and Visualizations

In an immersive video, each frame is an equirectangular 360° panorama. During playback, the video is mapped onto a sphere, with the observer at its center. In contrast to traditional video, the observer can interactively control their viewing direction. With regular 360° panoramic video players, this is usually done by clicking and dragging the video using a trackball metaphor. In a head-mounted display on the other hand, the viewing direction is directly controlled through head rotation, which can create a deeper sense of immersion. While we focus on users watching immersive video using a head mounted display, our proposed workflow also holds for regular 360° panoramic video players.

In order to be able to evaluate viewing behavior for such immersive video it is not enough to only consider the recorded gaze direction, but also the recorded viewing direction and field of view. An easy way to simultaneously visualize these aspects is to unwrap the video and map both the gaze position, as well as the field of view onto it. However, the warped equirectangular frame is often difficult to interpret (Fig. 2). The otherwise rectangular field of view becomes distorted in this visualization; particularly near the top and bottom (poles) of the frame. Additionally, the frame wraps horizontally, occasionally splitting the field of view. While the frame could be warped in such a way as to rectify and to center a single users field of view, this would further complicate any comparative analysis. We therefore conclude that

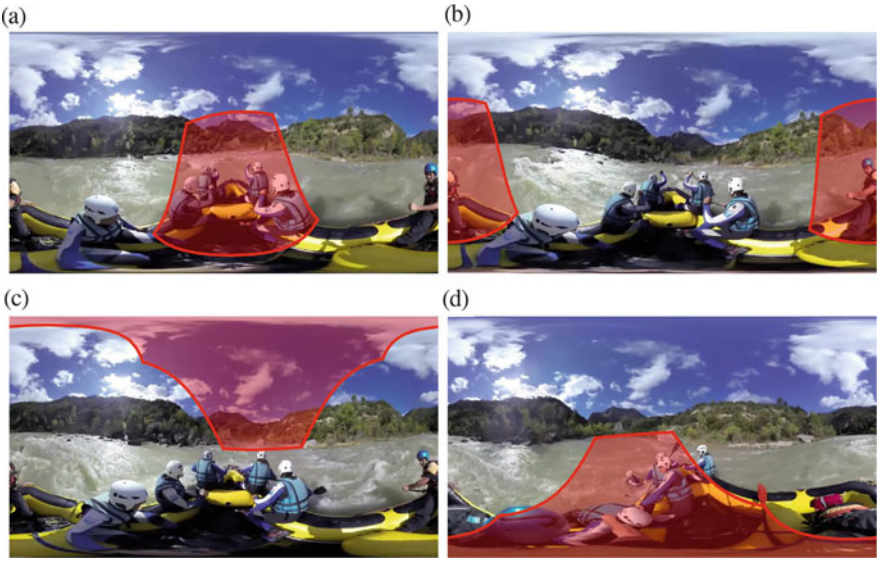


Fig. 2 Distortions introduced by unwrapping a spherical video frame can complicate the interpretation of a participant's otherwise rectangular field of view. The *red area* represents a participant's field of view with (**a**) minor distortions, (**b**) horizontal wrapping, (**c, d**) polar distortions

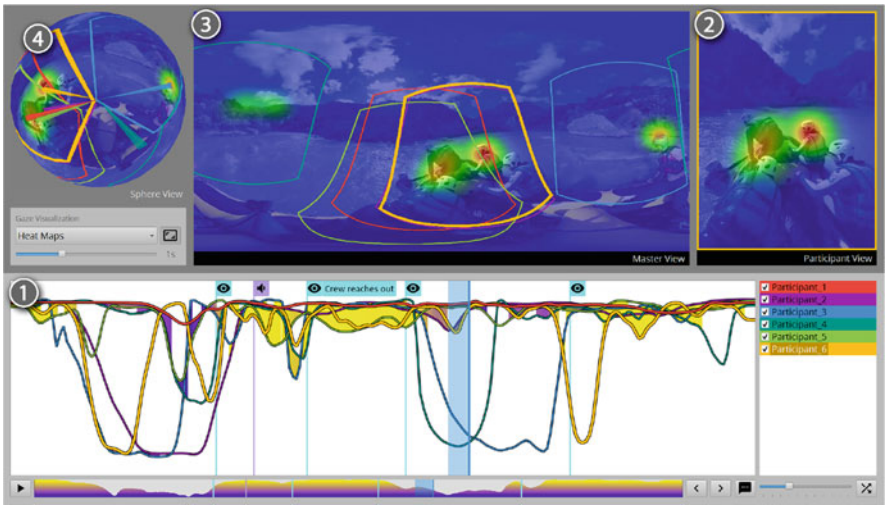


Fig. 3 Overview of our user interface for the immersive video clip "VIDEO 360: RAFTING" [1]. (*1*) View Similarity visualization for the entire clip. (*2*) Participant View for *Participant 6*. (*3*) Master View and (*4*) Sphere View. The color-coded frames indicate the individual field of view for each participant. Additionally, an attention map allows determining the overall gaze distribution

simply unwrapping the frame and applying traditional gaze visualization techniques for regular video is insufficient for analyzing immersive 360° video.

Figure 3 shows an overview of our proposed user interface.

The bottom half of our interface is dedicated to providing a temporal overview of the head-tracking data. A seek slider can be used to select a frame in the video. This slider is additionally overlaid with a quality metric that guides analysts towards potentially relevant sections, i.e. those sections in which many participants are focusing on similar regions of the scene. A specialized View Similarity visualization allows discriminating between individual participants, and combines temporal with spatial context. The viewing direction of each participant is represented by a line, with the proximity of the lines representing the view similarities over time. The closer the lines, the more similar the viewing direction in that frame.

The upper half of our interface is dedicated to analyzing gaze data, and to providing a spatial overview. On the right, a limited user view shows the scene from the currently selected participant's perspective. In the middle, an unwrapped view of the entire scene provides global context. On the left, an interactively rotatable 3D sphere-mapped version of the video allows the analyst to view the frame in a more natural projection. This allows for a better understanding of rotational context that is commonly lost in the unnaturally distorted unwrapped view.

Each of theses views can additionally be overlaid with established gaze visualizations commonly used for the analysis of regular video, such as animated attention maps or animated scan paths. For these traditional gaze visualizations, gaze data is aggregated over a user-controlled temporal window.

In the following we give a detailed description of each visualization and discuss its intended usage and technical details.

(1) View Similarity The most prominent visualization in our interface is the View Similarity visualization. It shows the proximity of all participants' viewing directions over time. This allows analysts to quickly identify moments of attentional synchrony between individual participants. In order to be able to visualize the relationship of multiple 3D viewing directions over time, we use a dimensionality reduction technique. First, we create a distance matrix of all recorded viewing directions, regardless of participant and timestamp. The distance Δ between two viewing directions $v_{1,2} \in \mathbb{R}^3$ is determined using their cosine dissimilarity:

$$\Delta(v_1, v_2) = cos^{-1}\langle v_1, v_2 \rangle,$$

where $\langle \cdot, \cdot \rangle$ denotes the scalar product.

We then use nonmetric multidimensional scaling [6] to create a 1D embedding of all viewing directions, as this method preserves relative distances as well as possible. Depending on the temporal resolution of the head-tracker and the number of participants, this matrix may become arbitrarily large. The approach therefore strongly depends on the stability and performance of the underlying MDS algorithm. On the other hand, head-tracking data is not subject to as fine and rapid changes as eye-tracking data. Therefore, reducing the temporal resolution for the purpose of calculating the View Similiarity, will still yield adequate results.

Finally, we reintroduce time as a second dimension. By connecting all records for a participant over time, we obtain a line graph. In this resulting graph the proximity of lines at each frame is an approximate representation of the similarity between viewing directions in that frame.

To further highlight attentional synchrony, similarities between viewing directions that are above a user-defined threshold are additionally marked, by visually connecting the lines into clusters. Readability is further enhanced by additionally coloring these clusters using a simple quality metric and color gradient.

This quality metric q uses the sum of normalized distances from each participant's viewing direction $p \in P$ to it's k-nearest-neighbor $k_p \in P$:

$$q = 1 - \frac{1}{|P|} \sum_{p \in P} \left(\frac{\Delta(p, k_p)}{\pi} \right)^2, \quad q \in [0, 1]$$

The smaller the distances, the more clustered the viewing directions, and the higher the quality. We found that $k = 2$ worked well for our smaller test data sets. We also empirically found that a default threshold value of one third of the field of view of the display worked well, which for our setup was approximately 30°.

A simple seek slider at the bottom allows selecting a frame in the video, as well as zooming and panning the View Similarity visualization.

Analysts can also place annotations at key frames, in order to mark important audio or visual cues in the video, thus adding semantic context to the visualization. Each annotation is represented by a colored flag on the timeline.

(2) Participant View The Participant View shows the scene as it was experienced by a single selected participant. The limited perspective is intended to prevent analysts from erroneously assuming information that is provided to them by the more global views, but that would not have been visible to the participant during the trial. This view allows analysts to study the attention of an individual participant, and to investigate which elements in the scene might have influenced that participant to move their field of view.

Unfortunately, this limited view does not allow the analyst to differentiate between movement of the participant's head and movement of the camera in the video. While the camera is often fixed in immersive video—as simulated self motion has a tendency to cause discomfort for some participants [15, 26]—there are a large number of fast moving amateur sports and drone videos, as well as an increasing number of artistic videos that make use of slow, deliberate camera movements. Therefore, our framework supplies two additional spatial visualizations that provide global spatial context in relation to the video content.

(3) Master View The Master View shows the entire scene as an unwrapped video frame. In this equirectangular mapping—also known as latitude-longitude mapping—the azimuth is mapped to the horizontal coordinate, while the elevation is mapped to the vertical coordinate of the image. This mapping format is commonly used, since the sphere is flattened into a rectangular area, and thus traditional compression methods for rectangular images and videos can be applied. In this unwrapped view, the center of the view is the relative *front* of the scene (Fig. 2a), and the left and right edges of the view are the relative *back* of the scene (Fig. 2b). The top and bottom of each frame exhibit the most distortion, as these map to the *top* and *bottom* poles of the sphere (Fig. 2c, d). Additionally, the individual fields of view of each participant are marked by color-coded frames. This view is intended to give analysts global context, since all events that are occurring in a frame of the video can be observed at once.

As previously discussed, the distorted perspective and the fact that the image wraps around can make interpretation difficult. Therefore, our framework provides an additional more natural mapping.

(4) Sphere View The Sphere View maps the immersive video to the inside of a sphere, which can be rotated using the trackball metaphor [24]. As with the Master View, the fields of view of each participant are marked by color-coded frames. An arrow from the center of the sphere to the eye focus position of each selected participant additionally marks the gaze directions in 3D. This grants the analyst an intuitive spatial understanding of which direction each participant is facing in the immersive scene. For fulldome videos, this sphere view can also be used to obtain the original dome master mapping (Figs. 8 and 9).

4 Results

Our attention analysis framework relies on head movement and gaze data being recorded while the participant is immersed in the video. While HMDs with integrated eye tracking have been costly and therefore suited for professional use only, consumer-grade devices have been announced [9] and will allow a larger community to perform eye-tracking studies in virtual reality. In order to be able to develop suitable visualizations for such future studies, we have developed and built our own HMD with integrated eye tracking [28].

Our HMD provides binocular infrared-based eye tracking with low-latency and a sampling frequency of 60 Hz. In a calibrated state the gaze direction error of the eye tracker ranges from 0.5° to 3.5°, increasing at the edges of the screen. The head tracker provides 100 Hz for updating the head orientation with a viewing direction error of 0.5° to 1.0°. The display has a native resolution of 1280 × 800 pixels and a refresh rate of 60 Hz. The field of view is 110° vertically and 86° horizontally per eye.

We recorded data from 6 participants (5 males, 1 female) of which 4 had normal vision and 2 had corrected-to-normal vision. Our perceptual study was conducted as follows: First, we explained the HMD and the concept of 360° video to the participant. The participant was then seated on a rotatable chair and the HMD was mounted on their head, while still allowing for free head and body movement (Fig. 4). After calibrating the eye tracker, different monocular 360° panoramic videos were shown to the participant, while their head orientation and gaze data was being captured.

We showed a total of four immersive videos. Three videos consisted of equirectangular frames with a resolution of 1280 × 640, the fourth video consisted of dome master frames with a resolution of 2048 × 2048.

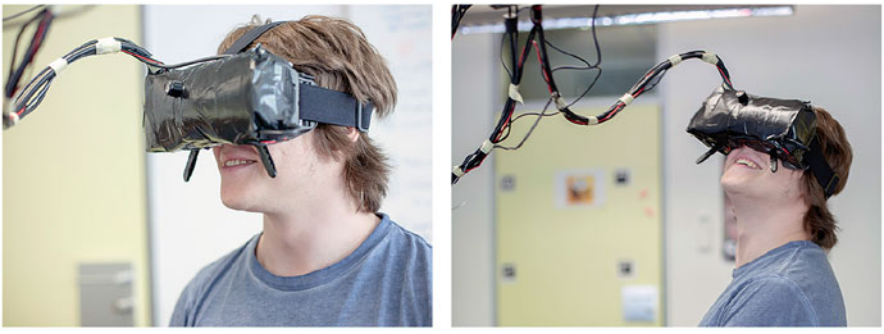


Fig. 4 Our experimental setup: A participant is watching an immersive video using our custom-built head-mounted display with integrated eye tracking. The participant is seated on a rotatable chair, in order to allow safe and free 360° body and head movement

In the following we take an in depth look at our results for two of these videos: The Creative Commons YouTube video "VIDEO 360: RAFTING", and a clip from the artistic fulldome short film "UM MENINO".

4.1 Video: RAFTING

"VIDEO 360: RAFTING" [1] is a short immersive video available under the Creative Commons CC BY license (Fig. 5). The clip is 42 s long and shows a rafting scene with a static camera centered in the raft. At 11 s into the video, the raft and the camera tilt, and two of the rafters fall into the water. The remainder of the video shows the crew reaching out and pulling the two back into the raft safely.

Figure 5 shows the View Similarity visualization with three frame annotations. The first annotation marks the moment the raft and the camera begin to tilt, the second marks an audible scream, and the third marks the moment the crew reaches out and starts helping their crewmates.

We observe that initially, all participants are individually exploring the immersive scene. The Master View in Fig. 3 shows that during this time, most attention is focused towards the travelling direction of the raft.

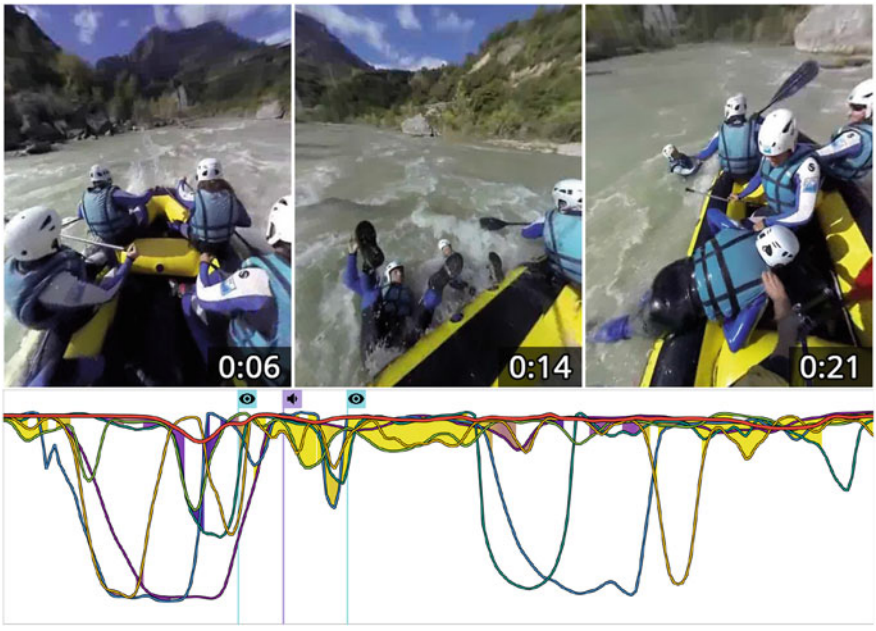


Fig. 5 *Top*: Scenes from the immersive 360° video clip "VIDEO 360: RAFTING" [1]. *Bottom*: View Similarity Visualization for the entire video

From the moment the raft and the camera tilt, all participants begin searching for what happened. We observe, that all participants' views follow the tilt of the raft and thus quickly converge around the two rafters in the water, even before the scream can be heard. Figure 6 shows the Master View during the rescuing efforts, with an overlaid attention map accumulated over all participants.

During the rescuing effort, the field of view of most participants remains centered on the events unfolding on the raft. It is particularly interesting that the gaze of most participants is focused on the helping crewmembers, rather than on the rafters in the water. After the first rafter has been saved, we observe that individual participants briefly turn their heads to check on the crewmember in the back of the raft.

After both rafters are safely back on the raft, we observe that most attention is again directed towards the traveling direction of the raft.

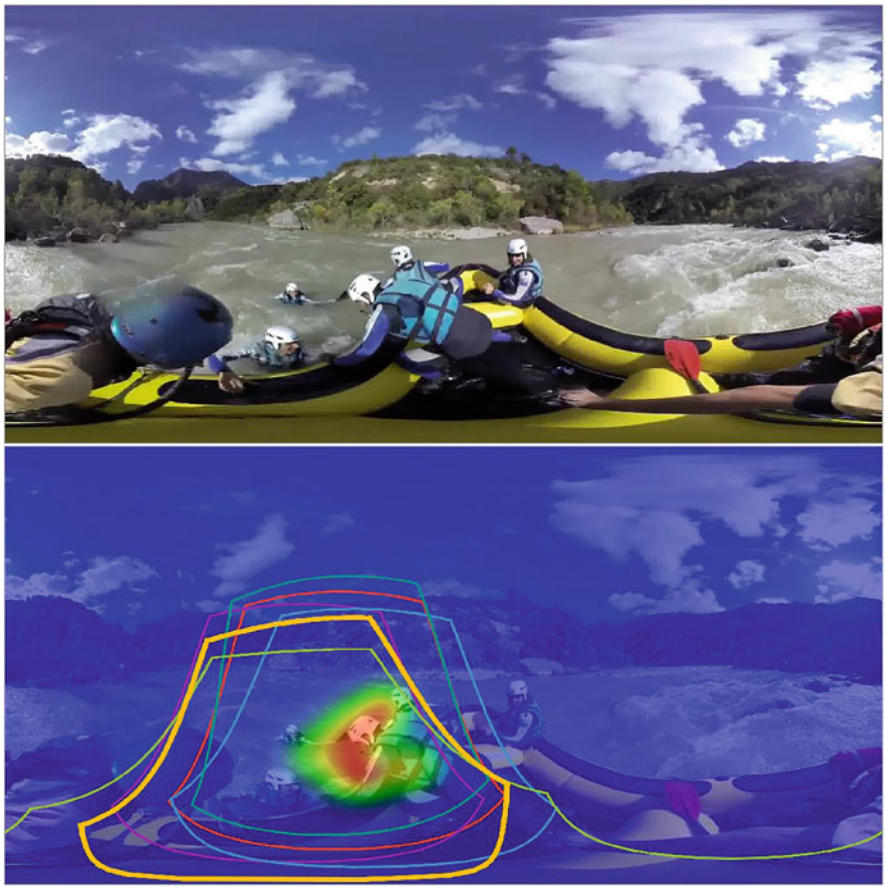


Fig. 6 *Top*: Master View of a frame from "VIDEO 360: RAFTING". Two rafters have fallen into the water, while the crew reaches out to help them. *Bottom*: The same frame overlaid with fields of view and an attention map. All participants are focused on the rescuing effort

In this use case our framework enabled us to quickly identify moments of attentional synchrony and to investigate the gaze behaviour leading up to that moment.

4.2 Video: UM MENINO

"UM MENINO" is an artistic 360° fulldome short film. The complete video is 5:46 min long and shows circus performers composited into a virtual environment. While the video is for the most part designed with a fixed forward direction, it has immersive elements. For our evaluation we selected a 45 s long sequence that starts at 2:23 min. The clip begins with a slow dolly shot moving backwards through a busy fairground. After 15 s the camera accelerates, simulating the viewer speeding away backwards in a roller coaster. After an additional 15 s the ride ends as the viewer emerges from the mouth of a giant clown, leading into an abstract kaleidoscopic sequence.

Figure 7 shows the View Similarity visualization with five frame annotations. The first annotation marks candy being thrown, the second annotation marks additional performers appearing, the third annotation marks a sound of the roller coaster

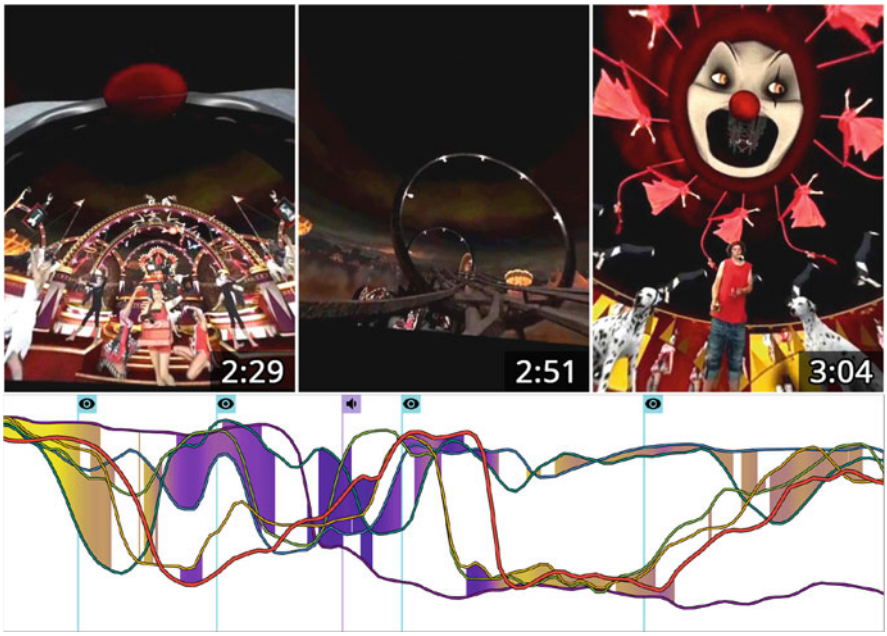


Fig. 7 *Top*: Scenes from the immersive fulldome short film "UM MENINO". *Bottom*: View Similarity Visualization for the entire video

charging up, the fourth annotation marks the moment the roller coaster accelerates, and the fifth annotation marks the beginning of the kaleidoscopic sequence.

The artists also created a regular video version of their short film, in which an observer is simulated [5]. This adaptation was intended to make the otherwise fulldome video available to a broader audience. We use this adaptation as a guide to understand the artist-intended viewing direction.

We observe that initially, each participant is individually exploring the scene. Using the Master View and the Participant View we additionally notice that most attention is indeed focused towards the artist-intended viewing direction.

At approximately 6 s into the video the performer at the center of the fairground reaches down and throws animated candy. This candy flies over the observer to the other side of the dome, and as can be seen in the regular video adaptation, the artists' intention was for the viewer to track it. In Fig. 8 (left) we observe that the participants recognized and initially followed the candy. However, they did not fully turn around, but rather quickly returned their attention to the busy fairground, which can be seen in Fig. 8 (right).

Shortly after the roller coaster sequence begins, the viewing directions form two distinct clusters. In Fig. 9 (left), the scan path visualization shows that most participants turn away from the artist-intended viewing direction, in order to instead face the travelling direction of the roller coaster. While this was not the case during the slow backward movement of the dolly shot of the previous sequence, the sudden acceleration appears to have caused a change in viewer behavior.

In the final kaleidoscopic sequence the camera slowly moves downwards, away from the giant clown at the top of the dome. This scene is largely symmetric, except for the artist-intended viewing direction, in which the protagonist of the video can be seen juggling. Figure 9 (right) shows that during this sequence most viewers

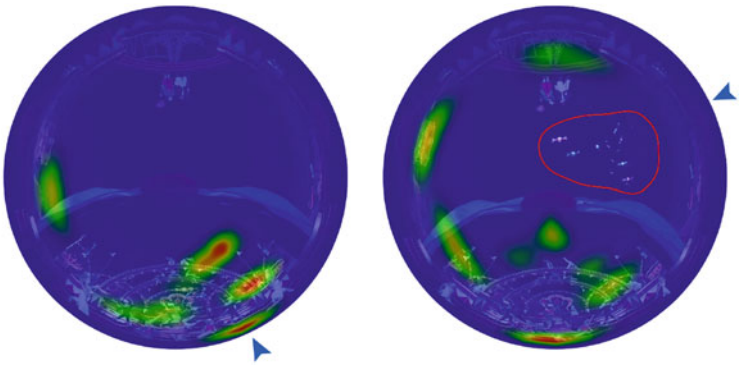


Fig. 8 Two Sphere Views of the candy throwing sequence from "UM MENINO". Both views are looking upward into the dome and are overlaid with attention maps. The *blue arrow* marks the artist-intended viewing direction. On *the left*: The performer at the center of the fairground begins to throw candy. On *the right*: The candy (*circled* in *red*) is floating across the dome along the artist-intended viewing direction, but none of the participants are tracking it

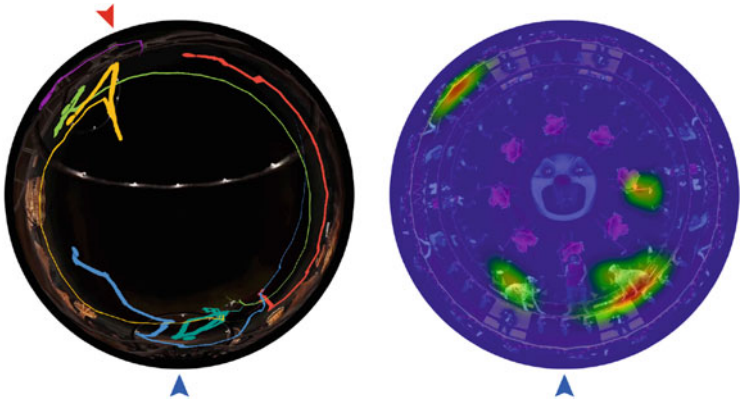


Fig. 9 Two Sphere Views from "UM MENINO" looking upward into the dome. The artist-intended viewing direction is marked by the *blue arrow*. On *the left*: The roller coaster sequence with scan paths. The *red arrow* marks the travelling direction of the roller coaster. On *the right*: The kaleidoscopic sequence with a superimposed attention map

have returned to the artist-intended viewing direction, focusing on the dogs next to the protagonist, with a slight tendency to look up.

In this use case, our framework allowed us to identify moments in which the observed viewing direction differed from the artist-intended viewing direction. Our additional visualizations further allowed us to investigate potential reasons for this difference in behavior.

5 Conclusion and Future Work

In this chapter, we have presented a novel visual analytics framework for jointly analyzing head movement and gaze data for immersive videos. Our design provides a specialized View Similarity visualization which allows analysts to quickly identify moments of spatiotemporal agreement between the viewing directions of individual participants. We also proposed three additional views (participant view, master view, and sphere view) that provide spatial context. These views can be combined with established gaze visualization techniques, in order to investigate viewing behavior in immersive video. We evaluated our approach within a small-scale perceptual study including amateur, choreographed and animated immersive video, and found that our framework can be used to detect whether the attention guidance of an immersive video works as intended.

As future work, we intend to further investigate how our method can be used to review and enhance artistic storytelling in immersive videos of different genres. We would like to extend our approach by supporting annotated areas of interest, in order to obtain an additional semantic context. We would also like to conduct a larger

perceptual study in order to gain further and more statistically significant insight into attentional synchrony and, by extension, storytelling in immersive videos.

Acknowledgements The authors thank Laura Saenger, Flávio Bezerra and Eduard Tucholke for permission to use the short film "UM MENINO". The authors gratefully acknowledge funding by the German Science Foundation from project DFG MA2555/6-2 within the strategic research initiative on Scalable Visual Analytics and funding from the European Union's Seventh Framework Programme FP7/2007-2013 under grant agreement no. 256941, Reality CG.

References

1. Ábaco Digital Zaragoza, VIDEO 360: RAFTING, 2015. youtube.com/watch?v=h0xo8QEPrk0, vis. 27 Jul 2015
2. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: State-of-the-art of visualization for eye tracking data. In: Proceedings of EuroVis, vol. 2014 (2014)
3. Burch, M., Kull, A., Weiskopf, D.: Aoi rivers for visualizing dynamic eye gaze frequencies. In: Computer Graphics Forum, vol. 32, pp. 281–290. Wiley Online Library, Chichester (2013)
4. Cheon, M., Lee, J.-S.: Temporal resolution vs. visual saliency in videos: analysis of gaze patterns and evaluation of saliency models. Signal Process. Image Commun. **39**, 405–417 (2015)
5. ChimpanZés de Gaveta, Um Menino Trilha: ChimpanZés de Gaveta, 2015. youtube.com/watch?v=q72AwheNYPk, vis. 18 Feb 2016
6. Cox, T., Cox, A.: Multidimensional Scaling, 2nd edn. Taylor & Francis, Boca Raton (2010)
7. Duchowski, A.T., Medlin, E., Cournia, N., Gramopadhye, A., Melloy, B., Nair, S.: 3d eye movement analysis for vr visual inspection training. In: Proceedings of the 2002 Symposium on Eye Tracking Research & Applications, pp. 103–110. ACM (2002)
8. Duchowski, A.T., Price, M.M., Meyer, M., Orero, P.: Aggregate gaze visualization with real-time heatmaps. In: Proceedings of the Symposium on Eye Tracking Research and Applications, pp. 13–20. ACM (2012)
9. FOVE: The world's first eye tracking virtual reality headset, 2015. getfove.com, vis. 29 Jul 2015
10. Google Jump, 2015. google.com/cardboard/jump, vis. 29 Jul 2015
11. Itoh, K., Hansen, J.P., Nielsen, F.: Cognitive modelling of a ship navigator based on protocol and eye-movement analysis. Le Travail Humain, pp. 99–127. Presses Universitaires de France, Paris (1998)
12. Itoh, K., Tanaka, H., Seki, M.: Eye-movement analysis of track monitoring patterns of night train operators: effects of geographic knowledge and fatigue. In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 44, pp. 360–363. SAGE Publications (2000)
13. Kurzhals, K., Weiskopf, D.: Space-time visual analytics of eye-tracking data for dynamic stimuli. IEEE Trans. Vis. Comput. Graph. **19**(12), 2129–2138 (2013)
14. Kurzhals, K., Weiskopf, D.: Aoi transition trees. In: Proceedings of the 41st Graphics Interface Conference, pp. 41–48. Canadian Information Processing Society (2015)
15. LaViola Jr, J.J.: A discussion of cybersickness in virtual environments. ACM SIGCHI Bull. **32**(1), 47–56 (2000)
16. Mackworth, J.F., Mackworth, N.: Eye fixations recorded on changing visual scenes by the television eye-marker. JOSA **48**(7), 439–444 (1958)
17. Noton, D., Stark, L.: Scanpaths in eye movements during pattern perception. Science **171**(3968), 308–311 (1971)

18. Nyström, M., Holmqvist, K.: Effect of compressed offline foveated video on viewing behavior and subjective quality. ACM Trans. Multimed. Comput. Commun. Appl. (TOMM) **6**(1), 4 (2010)
19. Oculus Story Studio, 2015. oculus.com/storystudio, vis. 29 Jul 2015
20. Perazzi, F., Sorkine-Hornung, A., Zimmer, H., Kaufmann, P., Wang, O., Watson, S., Gross, M.: Panoramic video from unstructured camera arrays. In: Computer Graphics Forum, vol. 34, pp. 57–68. Wiley Online Library, Chichester (2015)
21. Pfeiffer, T.: Measuring and visualizing attention in space with 3D attention volumes. In: Proceedings of the Symposium on Eye Tracking Research and Applications, pp. 29–36. ACM (2012)
22. Ramloll, R., Trepagnier, C., Sebrechts, M., Beedasy, J.: Gaze data visualization tools: opportunities and challenges. In: Proceedings of the Eighth International Conference on Information Visualisation, IV 2004, pp. 173–180. IEEE (2004)
23. Schulz, C., Schneider, E., Fritz, L., Vockeroth, J., Hapfelmeier, A., Brandt, T., Kochs, E., Schneider, G.: Visual attention of anaesthetists during simulated critical incidents. Br. J. Anaesth. **106**(6), 807–813 (2011)
24. Shoemake, K.: Arcball: a user interface for specifying three-dimensional orientation using a mouse. In: Graphics Interface, vol. 92, pp. 151–156. Morgan Kaufmann Publishers, San Francisco (1992)
25. Smith, T., Henderson, J.: Attentional synchrony in static and dynamic scenes. J. Vis. **8**(6), 773–773 (2008)
26. Soyka, F., Kokkinara, E., Leyrer, M., Buelthoff, H., Slater, M., Mohler, B.: Turbulent motions cannot shake vr. In: Virtual Reality (VR), pp. 33–40. IEEE (2015)
27. Stellmach, S., Nacke, L., Dachselt, R.: Advanced gaze visualizations for three-dimensional virtual environments. In: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, pp. 109–112. ACM (2010)
28. Stengel, M., Grogorick, S., Rogge, L., Magnor, M.: A nonobscuring eye tracking solution for wide field-of-view head-mounted displays. In: Eurographics 2014-Posters, pp. 7–8. The Eurographics Association (2014)
29. Tory, M., Atkins, M.S., Kirkpatrick, A.E., Nicolaou, M., Yang, G.-Z.: Eyegaze analysis of displays with combined 2D and 3D views. In: Visualization (VIS'05), pp. 519–526. IEEE (2005)
30. Youtube creator blog, 2015. youtubecreator.blogspot.de/2015/03/a-new-way-to-see-and-share-your-world.html, vis. 27 Jul 2015
31. Weibel, N., Fouse, A., Emmenegger, C., Kimmich, S., Hutchins, E.: Let's look at the cockpit: exploring mobile eye-tracking for observational research on the flight deck. In: Proceedings of the Symposium on Eye Tracking Research and Applications, pp. 107–114. ACM (2012)

Capturing You Watching You: Characterizing Visual-Motor Dynamics in Touchscreen Interactions

Leslie M. Blaha, Joseph W. Houpt, Mary E. Frame, and Jacob A. Kern

Abstract The relationship between where people look and where people reach has been studied since the dawn of experimental psychology. This relationship has implications for the designs of interactive visualizations, particularly for applications involving touchscreens. We present a new visual-motor analytics dashboard for the joint study of eye movement and hand/finger movement dynamics. Our modular approach combines real-time playback of gaze and finger-dragging behavior together with statistical models quantifying the dynamics of both modalities. To aid in visualization and inference with these data, we apply Gaussian process regression models which capture the similarities and differences between eye and finger movements, while providing a statistical model of the observed functional data. Smooth estimates of the dynamics are included in the dashboard to enable visual-analytic exploration of visual-motor behaviors on touchscreen interfaces.

1 Introduction

Development of effective interactive visualizations for touchscreens requires an understanding of the ways the visual features on the screen and the constraints of the touch interactions influence both eye movements and finger movements. For example, effective drop-down menus must use navigation paths large enough for fingers to drag through without dropping out of the menu. Evaluating the

L.M. Blaha (✉)
Air Force Research Laboratory, Dayton, Ohio
e-mail: leslie.blaha@gmail.com

J.W. Houpt
Wright State University, Dayton, Ohio
e-mail: joseph.houpt@wright.edu

M.E. Frame
Miami University, Oxford, Ohio
e-mail: frameme@miamioh.edu

J.A. Kern
Wright State University, Dayton, Ohio
e-mail: jacob.kern@outlook.com

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_5

efficacy of interactive visualization design choices requires an understanding of how those choices influence the cognitive processes supporting interactive visual analytics and how those processes manifest in measurable behaviors [22]. Here, we present a modular visual-motor visualization dashboard in which we can jointly examine hand/finger (motor) and eye movement dynamics, particularly as applied to touchscreen interactions.

Touchscreens offer unique advantages and challenges for research about user cognition and human-computer interaction because they couple the information displays with the input devices. This streamlines the device, eliminating the need for additional peripherals for input. The resulting systems are now ubiquitous as computer interfaces in public areas (e.g., ATMs, ticket kiosks) and portable in tablet or phone format. By integrating the input and display, however, touchscreens also couple the visual and motor sensory modalities such that changes in the display are often direct consequences of motor activity, unlike the indirect actions of peripheral computer input devices (mouse, keyboard, stylus).

Since the work of Woodworth [33], researchers have recognized that eye movements and motor activity should be studied together. In his early studies, Woodworth noted that the speed and accuracy of line tracing declined when participants closed their eyes compared to performance with open eyes. This established evidence for a strong role of visual feedback influencing motor actions. He also argued that both modalities introduced sources of error into overall task performance. Many researchers have subsequently used techniques that separate the visual and haptic modalities during task performance in order to study both the ways in which hand and motor behaviors can be used to understand underlying cognitive mechanisms and the nature of the relationship between the modalities [11, 24, 25].

A common finding in the literature is that the eyes generally precede the hand to a location in the environment or an object to act upon [2, 25]. Additionally, the eyes tend to linger on the object/location toward the conclusion of the motor action, though they may execute multiple saccades between the beginning and end of the task. For many years, this coordinated behavior in goal-directed aiming tasks was interpreted as evidence for the common command hypothesis [5, 6]. The common command hypothesis posits that eye and hand movements are planned and executed in parallel. Generally the correlations in the position of and timing lags between the eye and hand activity have supported this.

More recently, Hayhoe and colleagues have proposed an alternative explanation that eye-hand coordination in goal-directed actions is evidence for cognitive action planning [16, 19, 23]. In order to appropriately plan even a single event, or a series of events for a larger natural action (like making a sandwich), the eye movements reflect saccades and fixations to places containing task-relevant objects (e.g., location of bread or knife). They argue the eyes are planning ahead while the motor system is executing the planned actions. The convergence of the eyes on the same location of the hand toward the end of an action ensures proper completion of a step before proceeding to the next step. It is hypothesized that temporary eye-hand synergies served to simplify the task's cognitive demands by reducing the number of

control variables [23]. This is supported by the oculomotor control literature which finds evidence for a common control mechanism between the two modalities [1].

As noted above, popular methods for understanding the interplay of visual information (particularly visual feedback) and motor execution entail experimental separation of the two sensory modalities. That is, tasks are structured to have both open and closed control loop conditions [25]. In the former, visual feedback of the hand executing the goal-directed movement is blocked from view. In a closed loop, the observer can see her hand throughout the task. Alternatively, task conditions are structured so that visual processing is done with and without motor interactions [11]. In both cases, visual feedback improves speed and accuracy; the ways in which behavior changes between conditions are then interpreted in terms the cognitive mechanisms affected.

Manipulating the visual feedback resulting from an action has many advantages, but it is less practical in the study of interactions on touch screens and generally not ecological. Nonetheless, the eye and motor dynamics when interacting with touchscreens can produce rich data about the interplay between the two. For example, the display animation properties on touchscreens change a user's response dynamics: when icons are animated to follow the finger or objects are stretched and rotated, people slow their motions to match the responsiveness of the screen [3]. The effect is dependent on the refresh rate of the screen and the type of touch technology. A likely interpretation of this is that interaction modality is influencing the cognitive strategy so that the visual feedback is accurate with respect to the display content.

Fitts' Law [12] has long been the gold standard approach to capturing the trade-off in motor speed and accuracy resulting from changes in task difficulty or input modality. Fitts' Law models movement time as a function of distance between the start and end positions of a movement and the width of the target areas. In practice, Fitts' Law is fit as a least-square linear regression model. Thus, this provides a point-estimate and statistical summary of motor actions. But in many ways, although a standard approach, Fitts' Law oversimplifies movement behaviors and does not support the study of coupled eye-hand dynamics.

Recently there has been a surge in interest in using motor activity, particularly mouse movement behavior, to study continuous cognition [21, 29, 31]. Continuous cognition conceptualizes mental task performance as a non-discrete progression through mental states. Researchers have argued that the decision making process is continuously embodied in the trajectory of the mouse when a participant selects their response choice on a screen [31]. This argument has spurred multiple attempts to capture full trajectories and aspects of response dynamics from mouse movements or hand reaching as behavioral correlates of cognition [29].

Touchscreens lend themselves to examinations of continuous cognition, particularly the continuous processes supporting action planning, if the touchscreen movement trajectories and eye movements are collected simultaneously. Both the finger path and eye movement data are easily mapped onto the same 2-dimensional space defined by the touchscreen. Thus, the relationships between the two modalities are easily measured as a difference between pixel locations.

A new approach is needed for characterizing the coupling of visual and motor behavior on touchscreens that moves beyond correlating saccade and fixation times with movement start/end times that have been typical in the eye-hand coordination literature. In addition to position over time, collection of touchscreen motion further lends itself to capturing the functional dynamics of the movement responses. Several studies have noted that timing of the eye movements relates to various movement dynamics, such as peak velocity [4]. All moments of dynamics should be integrated into a single analytics platform. The goal of the present work is to introduce some approaches to creating integrated analytics tools for the study of visual-motor dynamics. We develop a modular visual analytics dashboard combining familiar visualization techniques for eye movements and mouse tracking data with functional data analysis. In particular, we highlight the use of Gaussian process (GP) regression for the simultaneous functional analysis of gaze location and finger positions, velocities, and accelerations (and higher order moments).

A dashboard provides a single platform for studying rich movement dynamics. The system can simultaneously show raw movement data, summary visualizations, and statistical models for hand and eye movements. The GP regression models provide functional estimates of position, velocity, and acceleration values as well as the statistical uncertainty of those estimates. We combine visualization of GP model estimates with modules depicting more well-known techniques for visualizing eye-hand coordination. Previous work established techniques plotting both eye fixation and mouse location x-y positions on a screen over time using space-time cubes [8, 10, 20] or traditional fixation heat maps. Both approaches have emphasized position and fixation duration over the movement dynamics. We included modular dashboard gauges to capture those familiar properties of movement trajectories and augment them with visualizations of velocity and acceleration estimates. Our approach models both eye and finger movements in a common statistical framework, enabling direct comparison of their behavioral profiles in order to address questions about the relationship between eye and finger trajectories and the movement dynamics of both modalities. We can also plot data in ways similar to ocular-motor coordination studies to highlight consistent and atypical patterns, which the statistical models can then better quantify.

This chapter is organized as follows. We first describe elements used to develop our analytics approach, including a sample data set taken from a larger study on touchscreen interactions (Sect. 2.1) and the hardware and software used (Sect. 2.2). Section 3 details some key modular gauges for visual-motor visualization, highlighting some of the types of insights they enable. We then delve into the details of the statistical modeling of movement dynamics in Sect. 4. We conclude with some next steps and future directions, illustrating an example of a full dashboard. Our goal here is to present visualization tools that will enable researchers in HCI, visualization, and cognitive science to utilize novel statistical models visualized in familiar ways for a more thorough study of movement dynamics.

2 Visual-Motor Analytics Approach

We take a modular approach to developing a visual analytics dashboard for capturing visual-motor behaviors on touchscreens. By taking this approach, we can leverage combinations of graphing and analysis techniques appropriate for the task under study. Dashboard components can be rearranged, added, or removed according to each researcher's preferences. The key characteristic of our visual analytics approach is that it enables a functional, rather than point-wise, analysis. Modular gauges can include animation and video of actual human performance, trial-by-trial plots of raw data together with statistical models of the dynamics. Additionally all gauges are synced to an interactive timeline of events to facilitate exploration, selection, and replay for closer examination.

2.1 Sample Visual-Motor Interaction Data

To demonstrate our analytic approaches, we selected a data set from a larger study of interactions and motor control on very large touchscreens [3]. These data were collected from a male volunteer (right handed, age 21) performing a touchscreen dragging task. The task required him to repeatedly drag his right index finger across the screen from a starting target to an ending target without losing contact with the touchscreen. This motion is similar to any interaction requiring the dragging of objects across a screen, such as dragging a file to the trash or rearranging the icons into meaningful patterns. The larger study by Blaha and colleagues utilizes the ISO standardized designs for assessment of computer input devices [17, 30]. For the dragging movements, this approach enables the study of input artifacts introduced by a device's display size and touch technology. By adding eye tracking, this approach grounds the study of how interacting with touchscreens influences hand-eye coordination against standardized tasks and measures. The resulting metrics can be leveraged for both understanding the cognitive mechanisms involved and making recommendations for interface designs on similar multi-touch systems.

Figure 1 illustrates two of the sixteen possible circular object configurations in the task. The white circles are the target positions. On a block of trials, the diameter of the configuration could change, requiring longer dragging motions between objects. Note that the diameter in Fig. 1a fits within the visual field of the observer, but the wider diameter in Fig. 1b is large enough that the observer must move his head to see targets on opposite sides of the circle configuration. This is a consequence of the observer standing only about 12 in. away from the large screen. Close proximity is necessary to reach the screen. The targets' sizes themselves could also change, which influences the precision of the motion needed to complete the dragging motion (see [3] for full details). In total, a single observer completed 400 dragging movements between targets with 16 target-diameter size combinations.

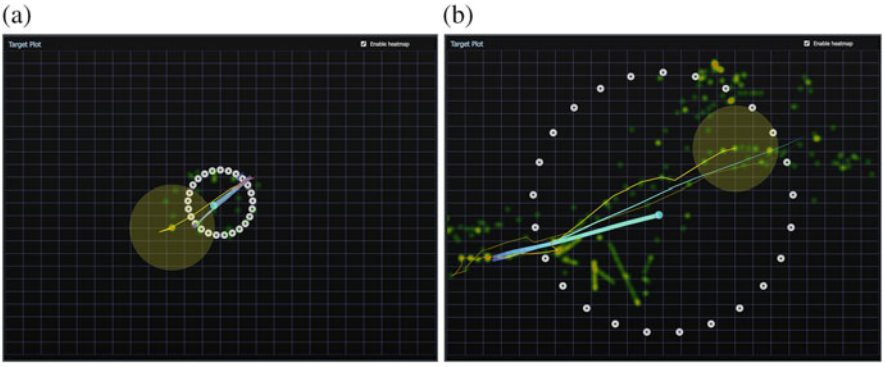


Fig. 1 Sample target heatmap visualizations showing both the finger (*blue line*) and eye movements (*yellow line*) overlayed on the target configuration for a block of dragging movements. The heatmap captures the eye fixation locations during the present block of movements. See text for full details. Note that (**a**) is the smallest configuration diameter and (**b**) is the largest, with 16 possible combinations of diameter plus variable target sizes

2.2 Equipment and Software

The dragging task was performed on an 82-inch, wall-mounted, rear-projection Perceptive Pixel-brand touchscreen (native resolution 4096 × 2400 horizontal by vertical, 75 Hz refresh rate). The Perceptive Pixel recorded touch inputs with Frustrated Total Internal Reflection [15]. Touch calibration was performed with the native software calibration routine. The task was programmed in Python with Kivy library version 1.8.0 and run with the Anaconda Scientific Python Distribution 1.9.1.

Eye-tracking data were collected with the Tobii Glasses 2 version 1.0.2. This system records a 160° horizontal and 70° vertical field of view at 50 Hz following a one-point calibration procedure. A wide-angle camera (1920 × 1080 pixels, 25 Hz) mounted between the lenses, centered on the observer's nose bridge, recorded the operator's scene view. The glasses record the eye gaze fixation point using 4 infrared cameras, two left and two right, mounted in the lens frame below the eyes. Gaze point data was mapped onto a 2-D image of the touch screen using Tobii Glasses Analysis Software version 1.0.

The dashboard visualizes and replays data stored in CSV files, which may be raw recordings or pre-processed outside the dashboard. Statistical modeling was completed in the R statistical computing language [26] and in MATLAB. Results were returned in CSV files to incorporate with raw hand movement. The eye-tracking data was pre-processed in the proprietary Tobii software and exported as a tab-delimited file. GP regression was done with the MATLAB GPstuff toolbox [32].

The dashboard interface is a web-browser application written in Node.js using common web languages, such as HTML, CSS, and JavaScript libraries (e.g., D3 [7]) to capture different data types within customizable, modular dynamic gauges. The use of standard web languages lends itself to integration with the growing number of

open source visual analytics tools programmed in web languages, such as PlotLy.[1] It is also compatible with web-browser based experimental software for recording human behaviors (e.g., the jsPsych library [9]). Our browser-based approach makes the dashboard platform-independent, which is important for touchscreens which may leverage custom or non-Windows operating systems. This is particularly the case for portable devices using iOS or Android operating systems. The operating system constraint prevents many touchscreens from using other recent open source tools for mouse movement tracing and analysis, like the Windows-specific MouseTracker software [14]. While our development has focused on analysis of already-collected data, the flexibility and compatibility of our approach will lend itself to online integration with interaction studies in future applications.

3 Modular Visualization Gauges

We developed dashboard gauges for replaying motor trajectories, eye fixations, and saccadic eye movements, together with model-based analyses explained in subsequent sections. The modular design of the gauges means they are easily exchanged for other models or desired analyses; additional gauges may be added, removed, or rearranged. All time series data are synced according to a timeline of events. This means that recordings of activity, including eye tracker recordings or screen captures, can be played synchronized to the data. Playback controls can be incorporated into their own gauge for researcher control.

The Target Plot gauge gives an animation of the task overlaid with both finger and eye movement data. This is a key gauge for presenting eye fixations and path tracing data in a format familiar to most researchers using these methods with visual enhancements emphasizing the performance dynamics. Examples of Target Plot gauges at two different time points are shown in Fig. 1. The targets for the task are shown as the circular configuration of white circles. These vary in both target width and configuration diameter between the blocks; the gauge specifications are scaled to the actual targets in the task. The goal is to represent in the gauge a view similar to the operator's view, which would also be captured by the eye tracker scene camera. Note that in this particular gauge, all targets for a block are shown simultaneously; an operator would only see the start and end targets for any given movement. Finger dragging movement is shown in a blue trace line. This depiction is consistent with the suggested x-y coordinate path plots in [14]. The green-blue circle at the head of the line shows the current finger position. We enhanced the path tracing by grading the line color according to velocity of the movement (range determined according to the raw data, not a predetermined range). In Fig. 1, brighter greens correspond to faster speeds, and darker blues correspond to slower speeds. Saccadic eye movements are similarly shown in the red-yellow trace lines. The area

[1] http://plot.ly

of view visible by the Tobii scene camera is indicated by the yellow circle, centered at the head of the eye trace line. Again, the line color is graded according to velocity, with darker red being slower speeds and brighter orange-yellow being faster speeds. Thus, for both time points captured in Fig. 1, we see the eye ahead of the finger. Given the large diameter in Fig. 1b, the finger is far outside the scene camera range, so we can infer it is outside the foveal region of the eyes.

The Target Plot gauge also contains an optional eye fixation heat map overlay, which can be toggled on/off. The fixation heat map is colored according to the total accumulated time spent on foveating any position on a green-to-red scale, with green being short fixations and red being longer. The heat values reflect total time fixated in a given location during the depicted block; these values are not transformed or subject to a threshold. When viewed dynamically during data playback, the fixation heat map accumulates over the course of the operator's motor activity, and it is reset for each block of trials during playback. This allows the researcher to watch the progression of the eyes relative to the finger. Locations revisited will accumulate more time (become more red). In this way, dynamic changes in the visual-motor coordination strategy may emerge as fixation pattern anomalies.

Figure 2 gives examples of gauges developed to illustrate the relative positions and raw dynamics of the eye and finger on the touchscreen. Figures 2a, b are sample Position Delta gauges. In all the plots, the vertical gray line indicates the current time in the time series, and the horizontal axis represents time from past (left) to future (right). During playback, the data slide from right to left over time, crossing the vertical line at "now". The vertical axis is the position difference in pixels, with the horizontal gray line at the zero or threshold value for the data presented. The left-hand plots are examples of Eye-Finger Delta gauges. Eye-finger delta quantifies, in pixels, the difference between the touchscreen positions of the finger and the eye. The lines are disjoint, because the finger is not always in contact with the screen (lifted in between dragging movements). The line only has a given position while the finger is actively touching the screen. The horizontal threshold is set to 64 pixels (approx. 1 in.), and lines dropping below threshold occur when the eye and finger are within 64 pixels of each other.

Eye-finger delta data often show a "v" shape, where the finger and eye move apart, then closer, and then apart during a single target-dragging movement. The steepness of the "v" gives an indication of the difference in speed at which the eye and finger are moving, with steeper slopes indicating a faster convergence or divergence of the positions. It has been observed that eye and finger latency are poorly correlated when examined with standard linear model techniques [25]. The eye-finger delta plots suggest that a simple correlation is not enough to capture the dynamic relationship between the position or speed of the two movements, because the degree and direction of the correlation likely varies over time.

The right-hand plots in Fig. 2a, b give examples of the Target Delta for Completed Trials gauge. This gauge shows two lines tracking the distance between the current target and each of the finger (blue) and eye fixation (orange) positions. The horizontal threshold is the location of the target, such that zero is "on target". Over time, these data generally show a larger delta at the beginning of the trial, which

(a) Single trial; eye and finger move separately

(b) Single trial; eye and finger move together

(c) Scarf plot of all eye-finger position differences

Fig. 2 Illustrations of the relationship between eye and finger position during dragging movements. Time is on the x-axis in all plots. In (**a**) and (**b**) the *left-hand plots* show the eye-finger delta with difference on the y-axis, such that zero means they are in the same place. The *right-hand plots* show the difference of position of the finger (*blue*) and fovea (*orange*) relative to the end target location; zero here means the finger or eye is on the target. (**c**) Gives a scarf plot summary of all times when the eye and finger are in the same place (*green*), the eye is leading the finger (*orange*), and the finger is leading of the eye (*blue*)

is expected as each a new target appears in a position opposite the end of the last movement.[2] Delta decreases as the eye and finger move toward the target location. Two things are highlighted well with this gauge: (1) the pattern of the eye moving toward the target before the finger, and (2) the eye movement slope often steeper (faster saccadic movement) than the finger (Fig. 2a). This pattern was reported as typical for eye-hand coordination by Abrams and colleagues [2]. Figure 2b captures a deviation from this typical pattern, where the orange and blue lines are coincident for a period of time; the eye is tracking the finger movements before looking ahead to the target. Simple computation of mean movement time or mean velocity would not detect this difference in pattern.

[2]Note that the eye data is continuously recorded, and the sharp rising slopes occur as an artifact of the sudden shift in target position. The finger data are disjoint, reflecting the finger being lifted from the screen between movements.

Independent eye tracking and path tracing studies are often interested in areas of interest (AOIs) within the stimulus or display. For the study of visual-motor coordination, we are interested in coordinated actions of interest (Co-AOIs). As the position of the eyes and hands are moving, we want to emphasize modeling the relative position of the finger and the eye fixations. We define three key Co-AOIs: (1) eye-finger-colocated, which is defined as fixation position and finger position being within 64 pixels of each other, (2) eye-leading-finger, defined as the eye fixation position being closer to the target than the finger position, and (3) finger-leading-eye, defined as the finger position being closer to the target than the eye fixation position. The scarf plot gauge in Fig. 2c illustrates the frequencies of the Co-AOIs over the course of the data collection. These are similar to plots used in [28] for capturing AOIs over time. In fact AOI and Co-AOI scarf plots could be stacked in order to align AOIs in the display with the Co-AOI activities to further study display content for action planning. In this Co-AOI scarf, eye-finger-colocated is coded in green, eye-leading-finger is coded in orange, and finger-leading-eye is coded in blue. Times when no dragging movements occurred (i.e., no finger data is available) are colored with the gray background color. The pattern of events are dominated by orange, demonstrating that the eye led the finger most of the time. Second, we see periods of green, indicating both are in the same position, often occurring at the end of the trial when the operator may be verifying he correctly reached the end target location. There are few instances of the finger leading the eye in this data set.

All together, the plots in Fig. 2 confirm previous findings that eye-hand coordination usually exhibit an eyes-before-hand pattern. Generally, this is interpreted as forward planning by the eyes to enact efficient motor behaviors [19]. For touchscreen interactions herein, forward planning likely consists of eyes looking ahead to locate the next target, checking the finger to ensure the target was reached, and continuing to plan for the next target. The advantage of the combination of gauges is that we can then compute the frequency (scarf plot) at which this predictable sequence did not occur (e.g., periods with the finger and eye together) together with the dynamics of that behavioral change such as in Fig. 2b. Cross-referencing the dynamic heatmap in the Target Plot gauge, we can observer that this particular example of the eyes staying on the finger occurred following a motor error, in which the observer lost contact with the screen, interrupting the dragging motion. It would appear that part of his error correction entailed longer-than-usual visual feedback confirming a correct motion on the second try.

While all the above visualizations capture some aspect of eye-hand movement dynamics, they do not provide a strong quantitative characterization of the data. In order to visualize full statistical characterization of movement dynamics, we develop a Target-Referenced Dynamics gauge for both finger and eye data. Two examples are depicted in Fig. 3. In both examples, the finger data is modeled in the left column of plots and the eye data is shown in the right column. From top to bottom, these gauges illustrate GP regression-derived estimates. The purpose of the Target-Referenced Dynamics gauge is to capture the full position, velocity, and acceleration dynamics in path-to-target coordinates based on GP regression of the finger and eye movement data (the analysis technique is described in Sect. 4).

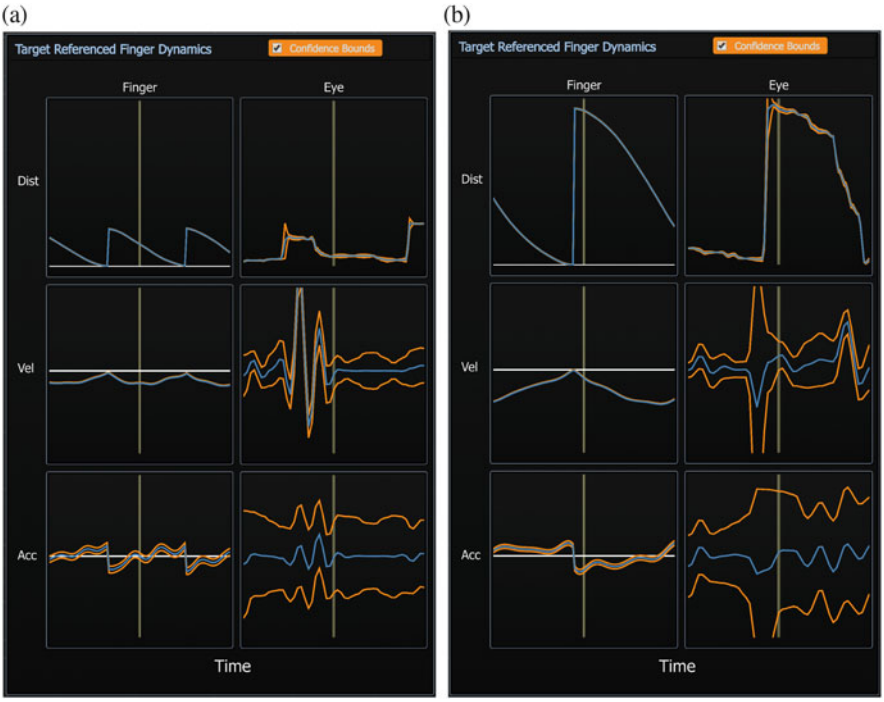


Fig. 3 Example Target Referenced Dynamics gauges for (**a**) a short-diameter configuration and (**b**) wide-diameter configuration. The data are derived from the GP model fit to finger position data (*left columns*) and eye fixation position data (*right columns*). From top to bottom, the *blue curves* give the smooth estimates of the position, velocity, and acceleration, respectively. The *orange curves* indicate the amount of uncertainty in each estimate. In all plots, the x-axis is time, where the vertical bar indicates "now" in the time series. When in playback mode, the data slide from right to left over time

Using target-referenced coordinates, the movements are illustrated as starting far and closing the distance to zero between the eye/hand and target positions. The Target-Referenced Finger (Eye) Dynamics gauges illustrate the profiles over time by scrolling the data from right to left, synced to the speed of the replay timing. By scrolling this way, the earliest events are always furthest to the left while later-occurring events are to the right. A vertical gray line again indicates "now" in the time series. The mean GP estimate is given in the blue curves, and the orange curves indicate the estimated region of uncertainty for each curve. We will return to the patterns of these plots in Sect. 4 after describing the GP models.

4 Modeling Visual-Motor Dynamics

To visualize the full dynamic profiles of the visual and motor movement time series data, particularly the position, velocity, and acceleration, our analysis approach uses Gaussian process (GP) regression [27]. We derived GP statistical models for both the finger and eye movement trajectories. This allows for both an estimate of the trajectory at *any* point in time, not just those at which an observation occurred, as well as the estimated variance (uncertainty) of the estimate at any point. Under a Bayesian interpretation of the GP model, the estimate is a posterior distribution over possible functions that pass through (or near) the observed points while simultaneously reflecting uncertainty in interpolated time points. This degree of uncertainty increases with increased distance to the nearest observed values (sparse sampling). We chose the GP modeling framework because it allows for the alignment and display of multimodal information collected from different apparati with unequal sampling rates using the same fundamental framework. Furthermore, GP regression also provides statistical models for derivatives of the trajectory. That is, it estimates the velocity, acceleration, jerk, etc. along with indication of the variability of those estimates. When the derivatives exist, the joint distribution over position, velocity, and acceleration is simultaneously estimated.

In general, a GP is uniquely defined by its mean and covariance function. In practice, the choice of covariance function dictates properties of the possible estimated paths through the observed data. The most commonly used covariance function appropriate for trajectory data is the radial basis function (RBF), examples of which are shown in Fig. 4a:

$$cov(s,t) = \tau^2 \exp\left(-\frac{(s-t)^2}{2l^2}\right). \tag{1}$$

Here, s and t are two different time points, and τ and l are the scaling parameters for the y-axis and x-axis, respectively. We used the RBF for finger path data. The RBF constrains the trajectory path data to a relatively smooth curve because this covariance function implies that the paths are infinitely differentiable. Finger movement is fairly constrained by the task itself due to response limitations using the touchscreen apparatus. The friction of the finger against the touchscreen led to smooth trajectories with virtually no small scale variability in the path. This made the RBF appropriate for characterizing finger movement data.

Within the class of Gaussian processes with a RBF covariance, the parameters must be chosen, or fit (e.g., with maximum likelihood methods), for the data. The fit parameters are τ and l mentioned above as well as the regression error standard deviation. The two parameters τ and l generally can both be fit together, although they can trade-off if there is not additional data constraining the velocity and acceleration profiles. For these data, we did not have a separate measurement of velocity or acceleration, so we used a maximum a posteriori approach with priors

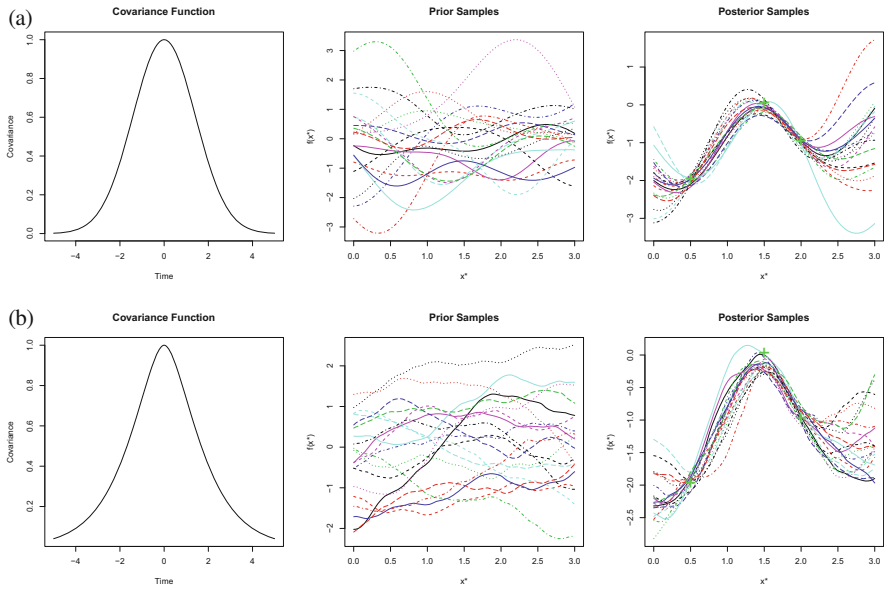


Fig. 4 Illustration of the GP regression process. The top row illustrates the radial basis covariance function, and the lower row shows the Matérn 5/2 covariance function. The *left-most plots* illustrate the shape of the kernels. The *center plots* are prior samples of the possible function shapes. Finally, the *right-hand plots* show samples of the posterior distribution, which constrain the possible prior functions to pass through the empirical data points, which are indicated by *green plus symbols*. (**a**) Gaussian process regression with radial basis covariance function. (**b**) Gaussian process regression with Matérn 5/2 covariance function

constraining the range of τ and l.[3] We chose the priors on τ and l based on known physical constraints of finger movement. Although the space of possible predicted velocity and acceleration profiles is quite large for any given trajectory, constraining the parameters based on maximum likelihood or maximum a posteriori values leads to relatively specific estimates.

Although the RBF was a sensible choice for modeling the smooth motor trajectories of finger path data, it was found to be less reasonable for eye trajectory data, which is more irregular due to the inherent jerkiness of large saccades as well as small-scale microsaccadic movements. To accommodate for both small-scale movements (e.g., microsaccades, drift) and large-scale movements (e.g., saccades) in eye-movement data, we used a Matérn 5/2 function. This is more robust to noisy trajectories than the RBF as it does not presume smoothness in the time series data.

[3]This approach is effectively the same as a constrained maximum likelihood fit with soft-constraints.

The Matérn 5/2 function, shown in Fig. 4b, is defined as:

$$cov(s,t) = \tau^2 \left(1 + \frac{\sqrt{5(s-t)^2}}{l} + \frac{5(s-t)^2}{3l^2}\right) \exp\left(-\frac{\sqrt{5(s-t)^2}}{l}\right), \tag{2}$$

where s and t are two different time points, and τ^2 and l are the scaling factors for the y-axis and x-axis, respectively.

Regardless of the kernel chosen, the derivative of a GP is also a GP. We leverage this to also estimate the velocity and acceleration of the finger and gaze trajectories, derived from the model fit to the position times series data. Just as we are able to compare eye to finger location differences by modeling distance to target using the same modeling framework, we are also able to compare the velocity and acceleration profiles of both the eye and the finger during the task. We visualize these differences side-by-side, as in Fig. 3. The covariance function between a GP at time s and the derivative of the GP at time t' is given by the derivative of the covariance function with respect to its second term. For example, the correlation between the location at s and the velocity at time t' assuming a RBF covariance is:

$$cov(s,t') = \frac{\tau^2(s-t')}{l^2} \exp\left(-\frac{(s-t')^2}{2l^2}\right). \tag{3}$$

The main advantages of the GP-regression approach are: (1) it is a statistical model of the trajectories; (2) it allows for straightforward statistical modeling of the derivatives of the trajectories; (3) it can be leveraged to compare across trajectories even if those trajectories have drastically different characteristics and/or are measured on different time scales. Because it is a statistical model, the GP yields estimates based on the data of the entire trajectory, not just location at observed time-points, as well as the uncertainty associated with those estimates. Similarly, because the derivatives of GPs are again GPs (as long as the GP is differentiable), the regression also gives estimates and uncertainties for velocity, acceleration, jerk, and higher order moments, each of which inform visual-motor behavior theories.

The GP illustrations in the Target-Referenced Dynamics gauges depict target-relative position, velocity, and acceleration of the mean of the posterior from the GP models fit individually to each trial and with τ and l fixed within blocks. There is an option to toggle on/off the point-wise uncertainty regions intervals around the mean posterior fit lines. Figure 3 highlights that finger and eye dynamics in this task have very different profiles. First, consider the finger GP model data. The smooth estimate of position for each dragging event shows a single movement from start to end position. The monotonic decreasing curve illustrates that the operator acquired the target and smoothly dragged toward the end target without deviation from a direct path. This motion is the same for short (Fig. 3a) and long (Fig. 3b) movements. The velocity and acceleration profiles indicate the classic initial speed up, followed by a slowing to land on the target. In the short movement (Fig. 3a), the acceleration is roughly symmetric around zero at the mid-point of the movement. In the long

movement (Fig. 3b), the acceleration is initially higher magnitude toward the target relative to the later deceleration. Particularly in the long movement, there is potential evidence of corrective sub-movements indicated by the non-monotonicities. This is consistent with multiple models of motor control theory (see, e.g., [2, 18]).

The eye data morphology appears to capture multiple large scale saccadic movements within a single trial. This is clear in the position plot that shows two jumps between 0 and 2400 pixels. The velocity and acceleration plots show sharp transitions between positive and negative speeds, a pattern indicating fast, saccade movements. Interestingly, these multiple saccades occurred while a single continuous motor path was being traced between the two targets. This confirms our intuition from the "v"–shaped plots in the Eye-Finger Delta gauge that the observer was looking away from the finger, then back at the finger. This seems to imply that the operator did not necessarily follow his finger with his eyes, nor did the operator simply look at the end target and wait to bring the finger to meet with the gaze point. Rather, the operator seems to simply check in on the finger's progress periodically, likely with the goals of ensuring the target wasn't dropped (error trial) and that the end target was reached correctly before lifting the finger to complete the dragging motion.

As the task or display constraints are varied, GP models will flexibly capture the systematic variations in movement dynamics. For example, suppose we add constraints that the operator must drag a physical target and the operator must keep the target inside a path or set of guidelines (e.g., menu navigation). We hypothesize that the eye movements would show more evidence of smooth pursuit of the finger and object as they move along the screen, rather than the saccades between the finger and the target locations ahead of the finger observed in this data. The GP framework can also be used to model dynamic changes in distance between the point of visual fixation and the location of the finger on the screen. Additional regression modeling on time, velocity, and acceleration parameters estimated from the GP models can be leveraged to produce a Fitts' Law-like analysis of the impact of task manipulations on the different movement dynamics. This additional analysis is left to future work.

5 Future Directions

Figure 5 shows one example of the modular gauges combined into a full visual-motor analytics dashboard. Playback controls are at the top; the right side includes scene camera video and plots of Fitts' Law analyses. Other experimenter-preferred gauges or analyses could be included. The interchangeability of plots and analyses is one strength of the modular dashboard framework. For example, it has is the capacity to display GP models under various covariance assumptions, such as RBF, rational quadratic, or Matérn class, independently or simultaneously for comparison purposes. Thus, we can leverage the dashboard playback for visual comparison of real and model dynamics to determine best models, instead of relying on fit statistics alone. An additional set of Eye Dynamics plots might be added to highlight the

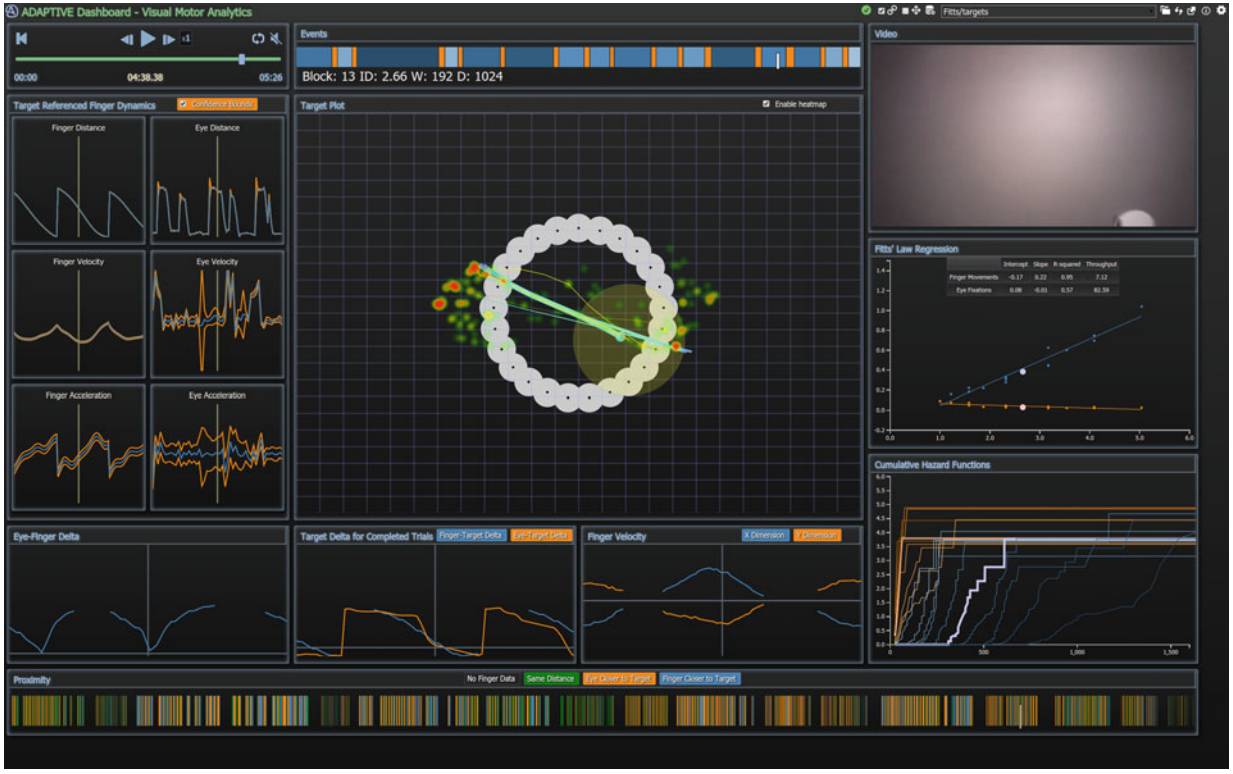


Fig. 5 Visual-motor dashboard for combined eye and finger movement data. The central gauge illustrates the target configuration for a given block overlaid with finger paths (*blue-green lines*), eye scanpaths (*yellow-red lines*), and a fixation heat map. The remaining plots capture various aspects of movement dynamics which are summarized in the text. For a full video illustrating the time-synced dynamic gauges see http://sai.mindmodeling.org/fitts

microsaccadic movements at a different scale than is currently shown. Phase space plots may be used to illustrate the relationships between position, velocity, and acceleration, or other higher order moments of the movement distributions.

Additionally, the GP models implemented in the dashboard provide parameter estimates needed to populate computational human cognition and motor control models. For example, the velocity and acceleration from the GP model are needed in the bang-bang model of human control theory. This model posits that motor actions follow a trajectory of maximum acceleration in the direction of the target up to the midpoint, then maximum deceleration to come to a complete stop at the target [18]. By populating this model with estimates derived in the dashboard, it can simulate realistic behavior for novel visual-motor interaction situations. These model predictions can be incorporated into the modular dashboard for streamlined evaluation of the model predictions in the same gauges as real human data.

A further key advantage of the GP approach is that, much like univariate and multivariate Gaussian random variables, linear combinations of GPs are also GPs. This means a GP model of the difference between the finger location time series and the gaze location time series is implied by the difference between GP models of the individual time series data. Consequently, we can estimate coupled dynamics directly from the statistical models of the individual modality dynamics. For the particular data demonstrated herein, this can be done by deriving the difference between the Matérn 5/2 and RBF functions. Not only, then, can we use full statistical models of the eye and motor dynamics, but we obtain a full statistical model of the dynamics of the difference between the two. In future efforts, once this derivation is completed, we will have a statistical model for coupled ocular-motor coordination.

An important facet of the modular approach is that using the Co-AOIs and dynamics we can confirm hypotheses about visual-motor coordination derived from previous literature. In the present data, consistent with action planning, the operator looks where he wants his finger to go, not at his finger during most of the task. This is likely not the case in all visual-motor tasks. But utilization of our visual-motor visualization dashboard approach will help characterize tasks that encourage one type of behavior or the other. We can further leverage the combination of eye-motor dynamics models to support future studies of continuous cognition in both modalities [13]. Remaining modular in design and platform independent in implementation keeps this flexible for applications across touchscreen technologies and data collection approaches, which may support direct integration into devices for real-time analysis in future applications.

Acknowledgements The authors thank three anonymous reviewers for their feedback on this chapter. The views expressed in this paper are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government. This work was supported in part by AFOSR LRIR to L.M.B. and AFOSR grant FA9550-13-1-0087 to J.W.H. Distribution A: Approved for public release; distribution unlimited. 88ABW Cleared 08/26/2015; 88ABW-2015-4098.

References

1. Abrams, R.A., Meyer, D.E., Kornblum, S.: Speed and accuracy of saccadic eye movements: characteristics of impulse variability in the oculomotor system. J. Exp. Psychol. Hum. Percept. Perform. **15**(3), 529–543 (1989)
2. Abrams, R.A., Meyer, D.E., Kornblum, S.: Eye-hand coordination: oculomotor control in rapid aimed limb movements. J. Exp. Psychol. Hum. Percept. Perform. **16**(2):248–267 (1990)
3. Blaha, L., Schill, M.T.: Modeling touch interactions on very large touchscreens. In: Proceedings of the 36th Annual Meeting of the Cognitive Science Society, Quebec City (2014)
4. Binsted, G., Chua, R., Helsen, W., Elliott, D.: Eye-hand coordination in goal-directed aiming. Hum. Mov. Sci. **20**(4), 563–585 (2001)
5. Bizzi, E., Kalil, R.E., Tagliasco, V.: Eye-head coordination in monkeys: evidence for centrally patterned organization. Science **173**(3995), 452–454 (1971)
6. Bock, O.: Contribution of retinal versus extraretinal signals towards visual localization in goal-directed movements. Exp. Brain Res. **64**(3), 476–482 (1986)
7. Bostock, M., Ogievetsky, V., Heer, J.: D3: data driven documents. IEEE Trans. Visual. Comput. Graph. **17**(12), 2301–2309 (2011)
8. Çöltekin, A., Demsar, U., Brychtová, A., Vandrol, J.: Eye-hand coordination during visual search on geographic displays. In: Proceedings of the 2nd International Workshop on Eye Tracking for Spatial Research (ET4S 2014). ACM, New York (2014)
9. De Leeuw, J.R.: jsPsych: a JavaScript library for creating behavioral experiments in a web browser. Behav. Res. Methods **47**(1), 1–12 (2015)
10. Demšar, U., Çöltekin, A.: Quantifying the interactions between eye and mouse movements on spatial visual interfaces through trajectory visualisations. In: Workshop on analysis of movement data at GIScience, Vienna, pp. 23–26 (2014)
11. Epelboim, J., Steinman, R.M., Kowler, E., Edwards, M., Pizlo, Z., Erkelens, C.J., Collewijn, H.: The function of visual search and memory in sequential looking tasks. Vis. Res. **35**(23), 3401–3422 (1995)
12. Fitts, P.M.: The information capacity of the human motor system in controlling amplitude of movement. J. Exp. Psycholo. **47**, 381–391 (1954)
13. Franco-Watkins, A.M., Johnson, J.G.: Applying the *decision moving window* to risky choice: comparison of eye-tracking and mouse-tracing methods. Judgm. Decis. Making **6**(8), 740–749 (2011)
14. Freeman, J.B., Ambady, N.: MouseTracker: software for studying real-time mental processing using a computer mouse-tracking method. Behav. Res. Methods **42**(1), 226–241 (2010)
15. Han, J.Y.: Low-cost multi-touch sensing through frustrated total internal reflection. In: Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, pp. 115–118. ACM, Seattle (2005)
16. Hayhoe, M., Ballard, D.: Eye movements in natural behavior. Trends Cogn. Sci. **9**(4), 188–194 (2005)
17. ISO9241-400: Ergonomics of Human-System Interaction – Part 400: Principles and Requirements for Physical Input Devices, Geneva (2007)
18. Jagacinski, R.J., Flach, J.M.: Control Theory for Humans: Quantitative Approaches to Modeling Performance. CRC Press, Mahwah (2003)
19. Land, M.F., Hayhoe, M.: In what ways do eye movements contribute to everyday activities? Vis. Res. **41**(25), 3559–3565 (2001)
20. Li, X., Çöltekin, A., Kraak, M.-J.: Visual exploration of eye movement data using the space-time-cube. In: Geographic Information Science, pp. 295–309. Springer, Berlin (2010)
21. McKinstry, C., Dale, R., Spivey, M.J.: Action dynamics reveal parallel competition in decision making. Psychol. Sci. **19**(1), 22–24 (2008)
22. Patterson, R.E., Blaha, L.M., Grinstein, G.G., Liggett, K.K., Kaveney, D.E., Sheldon, K.C., Havig, P.R., Moore, J.A.: A human cognition framework for information visualization. Comput. Graph. **42**, 42–58 (2014)

23. Pelz, J., Hayhoe, M., Loeber, R.: The coordination of eye, head, and hand movements in a natural task. Exp. Brain Res. **139**(3), 266–277 (2001)
24. Prablanc, C., Desmurget, M., Gréa, H.: Neural control of on-line guidance of hand reaching movements. Prog. Brain Res. **142**, 155–170 (2003)
25. Prablanc, C., Echallier, J., Jeannerod, M., Komilis, E.: Optimal response of eye and hand motor systems in pointing at a visual target. I. Spatio-temporal characteristics of eye and hand movements and their relationship when varying the amount of visual information. Biol. Cybern. **35**(3), 113–124 (1978)
26. R Development Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna (2011)
27. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. The MIT Press, Cambridge (2006)
28. Richardson, D.C., Dale, R.: Looking to understand: the coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. Cogn. Sci. **29**(6), 1045–1060 (2005)
29. Song, J.-H., Nakayama, K.: Hidden cognitive states revealed in choice reaching tasks. Trends Cogn. Sci. **13**(8), 360–366 (2009)
30. Soukoreff, R.W., MacKenzie, I.S.: Towards a standard for pointing device evaluation: perspectives on 27 years of Fitts' law research in HCI. Int. J. Hum. Comput. Stud. **61**(6), 751–789 (2004)
31. Spivey, M.J., Dale, R.: Continuous dynamics in real-time cognition. Curr. Dir. Psychol. Sci. **15**(5), 207–211 (2006)
32. Vanhatalo, J., Riihimäki, J., Hartikainen, J., Jylänki, P., Tolvanen, V., Vehtari, A.: GPstuff: Bayesian modeling with gaussian processes. J. Mach. Learn. Res. **14**(1), 1175–1179 (2013)
33. Woodworth, R.S.: Accuracy of voluntary movement. Psychol. Rev. Monogr. Suppl. **3**(3), i (1899)

Visualizing Eye Movements in Formal Cognitive Models

J. Timothy Balint, Brad Reynolds, Leslie M. Blaha, and Tim Halverson

Abstract We present two visualization approaches illustrating the value of formal cognitive models for predicting, capturing, and understanding eye tracking as a manifestation of underlying cognitive processes and strategies. Computational cognitive models are formal theories of cognition which can provide predictions for human eye movements in visual decision-making tasks. Visualizing the internal dynamics of a model provides insights into how the interplay of cognitive mechanisms influences the observable eye movements. Animation of those model behaviors in virtual human agents gives explicit, high fidelity visualizations of model behavior, providing the analyst with an understanding of the simulated human's behavior. Both can be compared to human data for insight about cognitive mechanisms engaged in visual tasks and how eye movements are affected by changes in internal cognitive strategies, external interface properties, and task demands. We illustrate the visualizations on two models of visual multitasking and juxtapose model performance against a human operator performing the same task.

1 Introduction

Eye-tracking technology provides a critical data source for the design and evaluation of visual analytics tools. The efficacy of information visualizations for human discovery, insight, and decision making is driven by a visualization's ability to

J.T. Balint (✉)
George Mason University, Fairfax, Virginia
e-mail: jbalint2@gmu.edu

B. Reynolds
Wright State University, Dayton, Ohio
e-mail: reynolds.157@wright.edu

L.M. Blaha
Air Force Research Laboratory, Wright-Patterson AFB, Ohio
e-mail: leslie.blaha@gmail.com

T. Halverson
Oregon Research in Cognitive Applications, LLC, Springfield, Oregon
e-mail: thalverson@gmail.com

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_6

successfully leverage perceptual and cognitive mechanisms [15]. Eye movements provide invaluable non-invasive measures of attention and visual information processing for assessing visualization interface efficacy. However, teasing apart the mechanisms supporting the observed behavior can be difficult based on eye tracking alone. Computational cognitive models provide powerful tools for understanding visual processes in complex tasks. Cognitive models are formal instantiations of theories about how the mind functions and operates in the physical world [5]. They capture perceptual, cognitive, and motor behaviors to produce predictions about human behavior, such as response choice and accuracy, response speed, manual motor activity, and eye movements. Critically, because of their computational nature, models can be extensively explored at a low cost to researchers in ways that provide specific predictions of human behavior.

Computational cognitive models can perform tasks with the same visual interface environments as human users [10, 20] and can be designed to produce eye movements similar to humans. Thus, we can use basic units of analysis like fixation locations, dwell times, and scanpaths, to study both human and model eye movements. Additionally, if we can visualize the internal dynamics of the model, we can begin to gain insights into the underlying processes producing the observable eye behaviors. In particular, the order of operations within the model highlight whether mental processes are causing eye movements or external events are diverting eye movements and triggering mental processes. For example, if an item in the memory process is activated before an eye movement process is activated, then we can infer that memory was causing the agent to shift his/her focus of attention in the display. In this way, we gain some understanding about the cognitive processes underlying visual task performance and how they are affected by interface attributes that would be difficult to derive by other methods.

Visualization plays a critical role in elucidating the eye movement behavior from formal models of human cognition and comparing it to observed human behaviors, as seen across Fig. 1. The goal of the present work is to leverage two very different types of visualization for an initial exploration of the complex

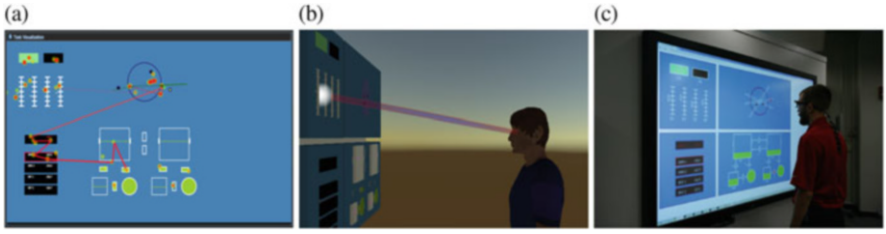


Fig. 1 Visualizing eye-tracking data generated by cognitive models of visual multitasking. (**a**) The model's internal representation of the experiment with its visual scanpath drawn in *red*. (**b**) Virtual simulation of a person whose eye movements are powered by a model of visual attention. (**a**) and (**b**) simulate (**c**) a typical human operator performing the multitasking experiment wearing eye-tracking equipment

interplay of cognitive activity and observable eye movements. We hypothesize that a combined approach will enable three types of insight. Insights about the cognitive mechanisms are gained by visualizing internal model dynamics during task performance together with the model's simulated eye movement behavior. Insights about model predictions for the physical human eye movements themselves are gained by embodying model behavior in virtual human agents. Finally, insights about cognitive strategies, model validity, and realism are gained by simultaneously visualizing and embodying model eye movements, comparing those to human eye movement behaviors, or directly comparing the dynamics and predicted movements of multiple candidate models. To enable this exploration, we present two qualitative visualization tools: a Model Visualization Dashboard system for capturing internal model dynamics supporting eye movement behavior, and an approach to virtual embodiment of model eye movements in human agents. This paper is structured in the following manner: Sect. 2 describes the task that is used to gather both human and model behavior data for our visualizations. Section 3 describes the cognitive modeling architecture, and Sects. 4 and 5 describe our visualization strategies. Finally, Sect. 6 provides an example of our visualization strategies when used on both human and model data.

2 Our Visualization Task

Both visualization techniques leverage the client-server software Simplified Interfacing for Modeling Cognition–JavaScript (SIMCog-JS) [10]. SIMCog-JS currently connects Java ACT-R (see next section) with the JavaScript-based Modified Multi-Attribute Task Battery (mMAT-B) [6]. This multitasking environment, shown in Fig. 2 and on a large touchscreen in Fig. 1c, contains four tasks, presented simultaneously, that require the operator to respond to different alerts and system states. Clockwise from the upper left, these are a monitoring task, tracking task, resource management task, and communications task.[1] Human observers are instructed to make the tracking task their primary focus. This task requires the operator to continuously track one of three circles moving along an elliptical path by keeping the mouse cursor over the moving target. Periodically, the to-be-tracked target switches; the operator is alerted to this switch by one of the circles turning red. The operator is expected to select the cued target and continues tracking.

Simultaneously, the operator must watch and respond to alerts from all other quadrants. In the monitoring task, the green/black rectangles might change color to black/red. Any of the sliders might move out the normal range (± 1 tick mark

[1] The four tasks used herein constitute the standard implementation of the mMAT-B for multitasking research [6]. The number of tasks can be flexibly increased or decreased and the nature of the tasks can be changed to accommodate the research questions of interest. For more on the mMAT-B software, see http://sai.mindmodeling.org/mmatb.

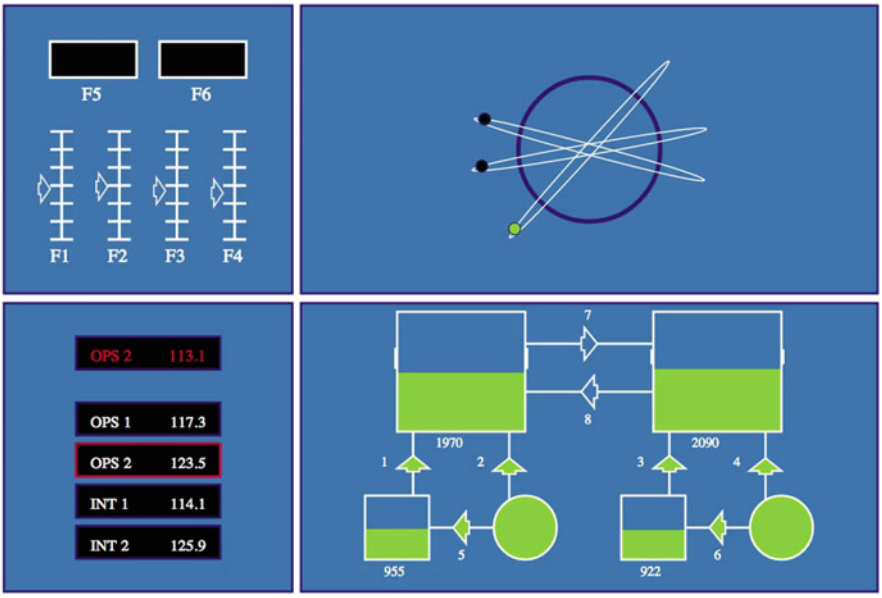


Fig. 2 A screenshot of the JavaScript implementation modified Multi-Attribute Task Battery (mMAT-B). Clockwise from *top left*, the quadrants contain the monitoring task (note that F5 is in the alert state), the tracking task, the resource management task, and the communications task (showing a target cue in *red text*)

from center). The operator responds to each out of state alerts with a key press, as indicated on the screen. In the communications task, a cue is given to set a specified channel to a new frequency value. The operator must select the target channel and then reach the target frequency value using arrow key presses. Finally, in the resource management task, the operator must keep fuel levels within the target range on the main tanks. Flow valves are toggled on/off with key presses, moving fuel between the fuel source, main tanks, and reservoir tanks. Valve switches periodically break and turn off, which could prompt the operator to toggle other valve switches in order to maintain the target fuel range.

All four quadrants entail visual alerts that can grab visual attention, and correct responses to the tasks require an observer (human or model) to periodically scan between tasks for those alerts. In the present work, to demonstrate the visual analytics and virtual embodiment of model behaviors, we utilize models implementing two possible task strategies and the mMAT-B environment to simulate eye movements during multitasking. Using two models allows us to illustrate the use of visualizations to compare model behaviors. In order to do this, we captured the SIMCog message streams which can be saved to a file for post-experiment review or sent directly to visualization software. The result of this is that while an ACT-R model is multitasking, we can simultaneously visualize its internal activity and

visualize the predictions for human eye movement behaviors. Before we introduce our visualizations, we review the characteristics of ACT-R.

3 Adaptive Control of Thought-Rational (ACT-R)

One way to model human activity is with cognitive architectures. The cognitive architecture used in the current research is ACT-R [1]. Other cognitive architectures, like EPIC [12] and Soar [13], or other modeling formalisms, like Guided Search [27], could also be utilized for research on cognitive strategies and behaviors. The methods presented herein for visualizing model behavior generalize across choice of models and can serve as a tool suite to compare candidate eye movement models.

Adaptive Control of Thought–Rational (ACT-R) is a general theory of human cognition, including cognitive, perceptual, and motor processes. The ACT-R cognitive architecture is a computational instantiation of that theory. Figure 3 illustrates a box-and-arrow representation of ACT-R. The cognitive architecture is used to build models that simulate how people perform tasks given the cognitive, perceptual, and motor constraints provided by ACT-R and the dynamics of the task with which ACT-R interacts, much as a human would. For the current discussion, it is critical that ACT-R produce eye movements. ACT-R includes an implementation of Eye

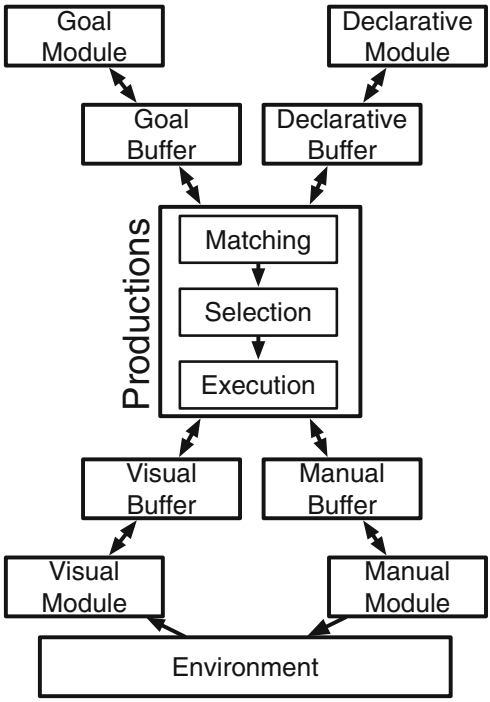


Fig. 3 Adaptive Control of Thought-Rational (ACT-R) [1] diagram

Movements and Movement of Attention (EMMA) [22], one theory that links eye movements to the covert shifts of visual attention produced by ACT-R. Other theories of eye movements, like PAAV [14], can also be utilized within ACT-R.

In ACT-R, covert shifts of attention are required for visual object encoding. In EMMA, object eccentricity, with respect to the fixation location of the eyes, and how frequently the object has been attended affect the time to encode the object. Eye movements tend to follow covert attention. However, an eye movement to an attended object might not occur if the covert attention to that object is brief. Extra eye movements (i.e., corrective saccades) can occur when the eyes overshoot or undershoot the center of covert attention.

The strategies that ACT-R models bring to bear on a task are encoded in production rules. Production rules are fine-grained representations of procedural knowledge that "fire" when the rule's conditions are met. This firing can change the model's internal state and initiate external actions. For example, a rule might specify that whenever a red object appears and visual attention is not busy encoding something else, then shift attention (and perhaps the eyes) to that red object. Only one rule may fire at a time, and which rule fires is primarily determined by the contents of the buffers (internal storage units). ACT-R includes a number of modules and associated buffers that instantiate theory related to specific cognitive processes (e.g., declarative memory, procedural memory). The modules make predictions about the timing and accuracy associated with the specific cognitive process. It is the interaction among the modules, buffers, production rules, and external environment that generate predictions of human behavior.

The output of model simulations includes behavioral data similar to that produced in human observations, such as response time, error rate, and eye movements. Over the course of a mMAT-B session, we track the time series data for eye position on the screen, mouse movement on the screen, mouse clicks, and button presses. Additionally, the cognitive architecture provides a detailed trace of the perceptual, motoric, and cognitive processes recruited by the simulated human to produce that behavior. The processes required for mMAT-B performance are the visual, manual, and temporal modules.

The use of formal cognitive models, such as ACT-R, allows one to explore how visual behavior is determined through the interaction of perceptual processes, manual motor interaction, and other cognitive processes like associative memory. An analyst encodes his/her hypotheses of how people accomplish a task in computational models, runs numerous simulations with those models, and then compares the results of the simulation with human data from the same task to help support or reject those hypotheses. While the use of such models does not fully confirm the underlying hypotheses, previous research has shown the utility of this methodology for understanding, among other things, how people visually scan graphs [17, 20], visually interleave complex dual-tasks [28], and visually search menus [9]. Visualizing the model's simulated eye movements, along with the detailed trace of the processes that produced those scanpaths, can provide additional insight into how people accomplish visual tasks.

4 Visual Analytics Dashboard

We first present a Model Visualization Dashboard to visualize the internal and eye movement activities of cognitive models. Deeper insights about the cognitive activity driving the external eye movements are gleaned from examination of internal cognitive model dynamics. Internal behaviors of ACT-R include the goals activated by changes in the task environment, information utilized by the cognitive mechanisms, and current strategies (i.e., production rules) being executed. The Dashboard has been designed to illuminate the internal activities and to help connect them to simulated eye movement behaviors. Multiple models can be displayed for direct comparison. Human eye-tracking data can also be included in a comparable format, although we do not have direct metrics for the internal activity of the humans like we do for the models.

To populate the Dashboard with model data, model states are stored using SIMCog in a time series data set, written to a CSV file. The CSV entries consist of time stamps, x/y pixel positions, and string data. The string data contains information about the internal states of the model, requests for interaction with the interface, and changes in interface elements "perceived" by ACT-R. Simple counts of the occurrences of different states/string values are computed for the visualizations. By reading this CSV, the Dashboard enables playback of ACT-R states and behaviors over the course of the task, and provides a human analyst the ability to examine the data using different performance metrics.

Human data from any eye tracker can be included in the Dashboard by similarly storing the data in a CSV file. For the human data herein, we recorded eye movements with a head-mounted Tobii Pro Glasses 2 eye tracker. The Tobii was worn by a participant during a 20 min session performing the mMAT-B task. Once the task is completed, the data is run through Tobii's Analysis software to map the gaze data to points on the screen. When that is completed, a perspective transform is done in Python to account for the angle at which participants viewed the task. Data is exported to a final CSV for visualization.

The Dashboard visualizes the raw data file in a web-browser application. It uses common web languages, such as HTML, CSS, and JavaScript libraries (e.g., D3 [4]) to capture different data types within customizable dynamic gauges. Each gauge visualizes a specific aspect of the model's current state. The different Dashboard gauges show a variety of data, such as recreations of what the model/human was visually presented, textual data giving details about that observer's state at specific times, and animated graphs that show trends occurring in the model.

The Model Visualization Dashboard, configured for model comparisons, is illustrated in Fig. 4, using 90 s of recording from the model performing the mMAT-B task; two model visualizations are shown side-by-side, each containing a set of gauges capturing model behaviors. The Dashboard can replay a model performing tasks, and playback controls are given at the top of the screen. Thus, we get real-time and summary visualizations of the model behaviors. Total task execution time elapsed is given in the radial Task Time gauge.

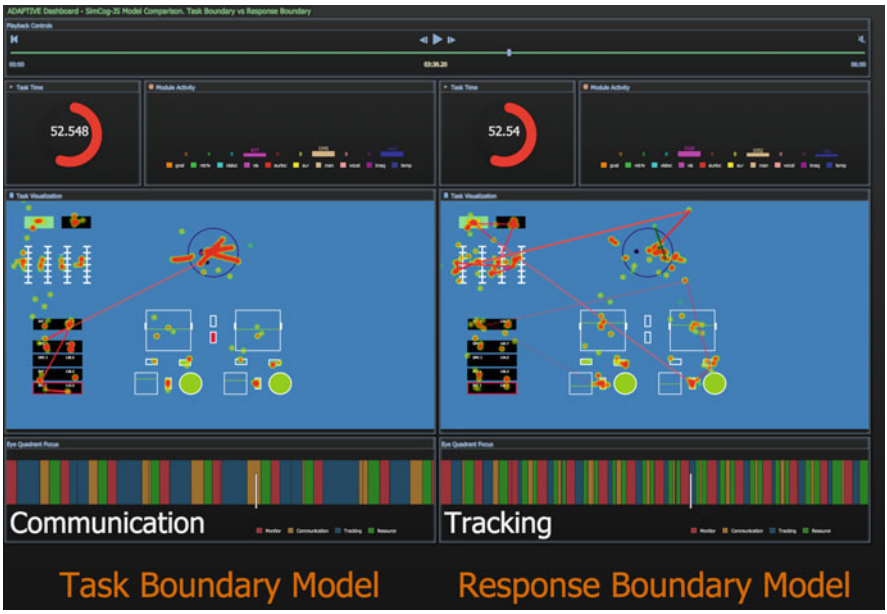


Fig. 4 Model Visualization Dashboard, set up for comparing two models, displaying simulated eye movement data from ACT-R models implementing different multitasking strategies. Component gauges are described in Sect. 4, and the two models are described in Sect. 6

The central Task Visualization gauge shows the model's visicon, its internal visual representation. The scanpath of recent eye movements is drawn as a red line over the task. Mouse movements are similarly shown with a green line; given the constraints of the task, the green line appears only over the tracking task. These paths are transient and show a recent 10 s of activity. A cumulative fixation heatmap, also called an attention heatmap [3], is also overlaid to show all fixations from the start of the task. The heatmap color values give the total accumulated time at which the model has looked at each fixation point. The green end of the spectrum gives short total times, and the points grow red and more saturated as time accumulates. The values are not subjected to a threshold. Unlike the scanpaths, the cumulative heatmap values do not degrade over time. This allows the researcher to see the task from the model's perspective with the additional insight given by the mouse and eye locations. Note that the colors selected here are for strong hue and luminescence contrast with the mMAT-B's blue background; these parameters can be flexibly changed according to analyst preference.

Below the Task Visualizer is the Eye Quadrant Focus gauge, which shows a scarf plot visualization of the time spent in each area of interest (AOI) in the task [21]. The AOIs correspond to the four task quadrants. Fixations anywhere within a quadrant are coded as in that AOI; we do not segment the AOIs into task subcomponents at this time. If a model switches tasks more frequently, the scarf colors will change

more frequently. During playback, the current quadrant in which the model is looking is named below the plot, and the white slider corresponds to the current time point in the model replay.

The Module Activity Gauge above the Task Visualizer gives a bar graph showing the level of activity in each of ACT-R's modules from the start of the task. The modules utilized in the two models herein are the visual module (bright pink), manual model (light orange), and temporal module (blue). This gauge can reveal when specific resources are a bottleneck across all goals. For example, if all task performance declines over an interval in which visual resource usage is maximized, then we can infer that the visual demands of the tasks are too high for successful completion of all goals. As shown in Fig. 4, different strategies may utilize resources differently, resulting in different heights of the Module Activity Bar.

The Model Visualization Dashboard provides multiple ways to view model strategies in action, going well beyond a simple video playback of the model performing the task. Additional gauges might be added to illustrate the sequence of production rules firing as the task is performed, as well as illustrations of the model's motor activity, to further capture the underlying cognitive processes. However, any additional gauges are currently outside the scope of this work. Our Model Visualization Dashboard allows an analyst to see which tasks or actions are slowing performance for a given model. This empowers a researcher to draw conclusions about human performance or make modifications to a model or the task interface. Insights about internal processes lead to hypotheses about observable human behaviors which can be tested through both animation and human experiments.

5 Virtual Embodiment of Model Eye Movements

Beyond examining the internal model dynamics, we visualize eye movement by embodying model activity in a virtual character operating in a 3-D environment through animation. This gives us a concrete way to examine model predictions as they would play out in real people. It further empowers an analyst to quickly observe a cognitive model's head and eye movements, determining unrealistic behavior at a glance. Furthermore, 3-D virtual environments allow for situations where eye-tracking tasks exist in a 3-dimensional world. Mobile eye-tracking systems such as the Tobii Glasses allow for tracking activities without the need to focus on a screen or to be constrained to a laboratory environment, such as driving in a real car as opposed to a driving simulator. Real-world 3-D environments will need to be modeled differently, and although un-examined in this work, the ability to compare the behavior of cognitive models in these more complex environments will ultimately allow for novel analyses of this kind to take place.

Virtual humans further allow a researcher to examine additional data, such as head movements, available from mobile eye trackers, complimenting the 2-D view of the Dashboard. The 3-D environment allows the analyst to move the camera and lighting for different views of the agent in action. Figure 5 illustrates this concept

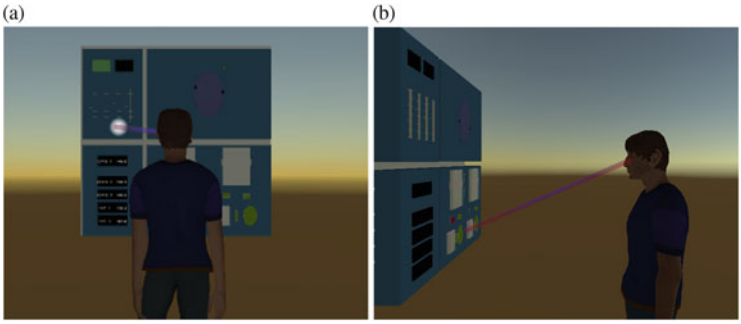


Fig. 5 Front view (**a**) and side view (**b**) showing the explicit fixation beams illustrating the gaze angle and focus of attention within the virtual 3-D mMAT-B environment. The beams are the *red lines* between the virtual human and screen, and the focus of attention is the *white halo* on the virtual mMAT-B screen

with a character multitasking. Typically, eye-tracking software records the center of fixation, measured in pixels relative to the captured image. This is often interpreted as the location of visual attention and is similar to the information produced from ACT-R's vision module. Using a virtual character to display eye movement patterns can provide an analyst with a high fidelity visualization method to examine the realism of a model's saccadic and smooth pursuit behaviors. Again, in the present work, the data driving this virtual agent is derived from SIMCog-JS JSON messages passed via WebSockets.

The utility of virtually modeling attention in 3-D space has been shown in work eliciting joint attention between agents and people [7]. However, these methods only model the position of the eyes and do not model the scanpath. Furthermore, unlike the models used by Itti et al. [11], which implemented eye movements based on low-level salience maps to control virtual agents, we used cognitive architecture-based models. These architecture-based models capture higher-level cognitive processes supporting eye movements with parameters that can be tailored to emulate eye-tracking data in specific tasks.

To calculate eye movements from either a model-generated or human-measured center of attention, we first must determine the world position of the fixation region in 3-D space. This is accomplished by assuming all of our data is modeled from a task displayed on a screen in front of the virtual character. By using this assumption, the area of attention for a given task can be converted into world coordinates by a simple linear transformation, treating the center of the screen in the virtual environment as the origin. We attach a halo or orb to the world position, allowing an analyst the ability to track the movement of a character's eyes either by viewing the character's face or by watching the orb position in the virtual task environment. Essentially, we view in a virtual simulation the same information that is seen in the Dashboard's Task Visualization gauge, like in Figs. 1a and 4.

Once the fixation location is determined in world coordinates, the rotations of the eye and head are then determined through cyclic coordinate descent inverse

kinematics [26]. This provides an easy method to switch from one eye position to another. Inverse kinematics only provide the end rotations of a series of selected joints. This will cause the center gaze of the eye to snap to a given rotation. This appears as a jump, which is acceptable for ACT-R models, as they currently only produce saccadic eye movements. Yet, for data from human or other models, a virtual character needs to perform other forms of eye movements, such as a smooth pursuit. For smooth pursuit eye movements, using cyclic coordinate descent can cause undesirable and unrealistic behavior. Therefore, we linearly interpolate between gaze points, which exist as 3-D points and provide the movement trajectory between known points. We examined two other common interpolation methods, SLERP and SQUAD [8], but have found that these methods create strange eye movement patterns between points. The inherent spherical nature of these techniques cause gaze patterns to move in an arc, which does not produce smooth pursuits for linearly moving objects. More complex interpolation techniques such as Gaussian process interpolation have not been examined using virtual human animations and are left for future work.

Our 3-D modeling system allows a screen generated at any size on which a virtual human can operate. Screen size differentiation allows us to model experiments that are performed on larger and larger screens, such as the screen seen in Fig. 1c, which are growing in popularity in some applied domains. As screen size grows larger, the eye movements become compounded with head rotations. Eye-tracking systems that track head rotation, and provide head rotation separately, become invaluable to building a realistic simulation from both human and model data. Too large a screen without head rotation data forces the model eyes to rotate in an unrealistic manner, which is easily observed by a quick examination of the virtual character displays. As this is examined from the perspective of the eyes, a 2-D examination of the screen and focal points is not sufficient for this kind of information, as scale and head movements are lost.

After examining the character operating on virtual tasks using our generated eye movement models, we noticed that it is difficult to watch his eye and head movement while simultaneously examining the areas of attention that eye-tracking data and cognitive models produced. The size of the eyes, specifically of the pupil and iris, is quite small relative to the rest of the body and to the distance between the character and screen. Therefore, we also provide a method to exaggerate the eye movement and track the connection between the area of attention and center point of the eye socket. The connection is modeled as fixation beams, one for each eye, seen in Fig. 5. This exaggeration can also be combined with the halo over the area of attention, which can be seen in Fig. 5a. To model the fixation beams connecting a character's eye to the area of attention, we construct a rod from the area of attention (transformed into world coordinates) to the center-point of the virtual human's eye. Pitch and yaw rotations are calculated between the two points, providing a new transformation matrix to the rod. Using a rod between eyes allows for more exaggerated movements of the eyes and head, exaggerating the differences in behavior patterns that might go unnoticed using other visualization tools.

6 Visualizing Eye Movements Strategies

We illustrate the utility of this multi-pronged approach to visualizing formal model eye movements in the comparison of two candidate multitasking models. Using ACT-R [1] with EMMA [22], we developed multiple, hypothetical task-interleaving strategies. These models are a priori predictions of human behavior based on task analyses and constraints provided by the cognitive architecture. Leveraging the visualization tools herein, we can compare these model predictions to each other and to human behavior in order to highlight similarities between them and to identify components in the models needing further theoretical development and refinement.

One theory about how people multitask is that they interleave multiple "threads of thought", switching between these threads as cognitive processes and task demands allow [23]. Two simplified multitasking strategies along these lines are Task-Boundary Interleaving (TBI) and Response-Boundary Interleaving (RBI). The TBI strategy only shifts visual attention to another task when either no response is needed to the currently attended task or after all key presses to resolve that task have been initiated. That is, only when one task is completed will the system switch to a different task. There is minimal task overlap, as each task is executed sequentially.

The RBI strategy attempts to maximize the overlap of visual attention shifts and keyboard responses by shifting visual attention to another task if consecutive key presses are needed for a response to the currently attended task *and* shifts of visual attention are not required to determine the correct key response. This is based on the assumption that if people only need to finish a determined sequence of motor actions to complete the current task, then visual attention can be reallocated to the next task while the manual response is completed. The communication task in the lower left corner is the only task in the mMAT-B environment that meets this criterion of repeated key presses (multiple arrow key presses are often needed to change the channel value). So by definition of the RBI strategy, the model (and people using this strategy) should switch attention to the next goal while completing a response to the communication task.

Each strategy elicits a different set of scanpaths and visual fixation patterns. Based on our strategy definitions, we predict that TBI will result in the model spending more time fixated on the lower left quadrant than RBI, and that RBI may scan between the different quadrants more frequently than TBI, reflecting more shifts in covert attention. The scarf plots in Fig. 4 illustrate exactly this predicted pattern, with RBI switching AOIs more frequently than the TBI model. Thus, the Model Visualization Dashboard can quickly capture some key differences in the cognitive models' performances on the mMAT-B task.

A key step in model development and validation is comparing the model predictions to empirical observations from human operators. Figure 6 shows 90 s of human performance data in the same Model Visualization Dashboard as the RBI model. Note that the difference in the Task Visualization panel is that the RBI model data shows the ACT-R visicon (as discussed above) while the human data shows the screenshot from the actual display used by the human operator. Note that the

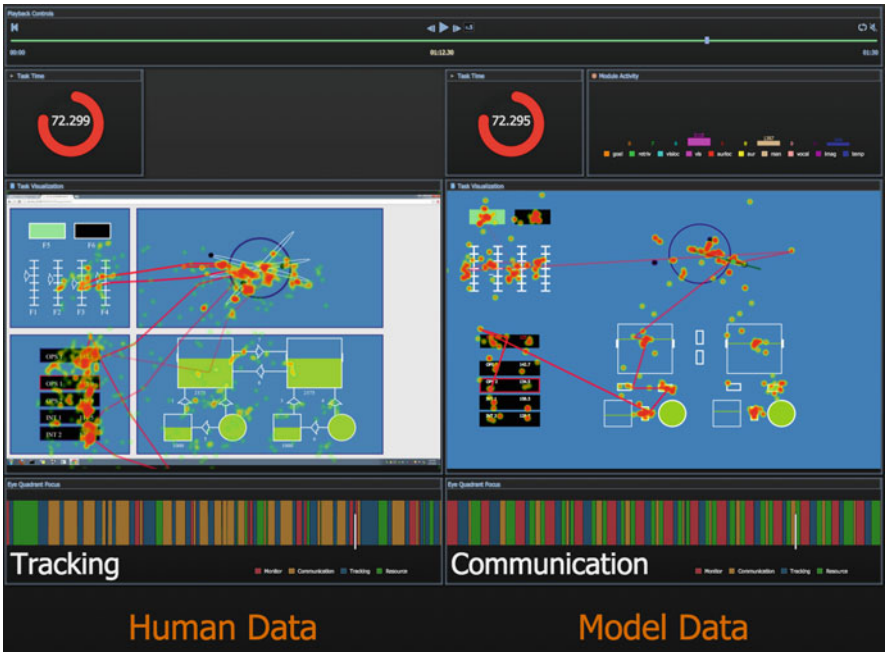


Fig. 6 Dashboard configured to visualize a human eye tracker recording (*left*) together with a cognitive model (*right*). The Response Boundary Interleaving (RBI) model is visualized in this figure

human data does not include the Module Activity gauge, because we do not have a direct way to make a similar assessment from human behavioral data. All other characteristics of the eye movement visualizations are the same as for the model data.

There is strong evidence from the Dashboard in Figs. 4 and 6 that the RBI bears a stronger similarity to the human data than the TBI data. This is particularly evident in the scarf plots, where both the human and RBI data show more task switching than the TBI data. We illustrate this further with recurrence plots in Fig. 7, which plot the sequence of fixation AOIs for the human (Fig. 7a), TBI model (Fig. 7b), and RBI model (Fig. 7c). Notably, the TBI model shows perfect recurrence, as the TBI strategy sequentially completes each task without switching the order. Both the human and RBI model show more frequent AOI switches, and the recurrence plots highlight that there are changes in the patterns of those switches, which are likely driven by cuing events in the mMAT-B task. Thus, this visualization approach provides some immediate evidence that the RBI model is a better candidate to explain the strategy used by the human than the TBI model.

However, both model predictions are incorrect in many ways. For one, the human data shows more time spent on the tracking and communication tasks. While the model predictions do reflect a greater dwell time on the tracking task, perhaps

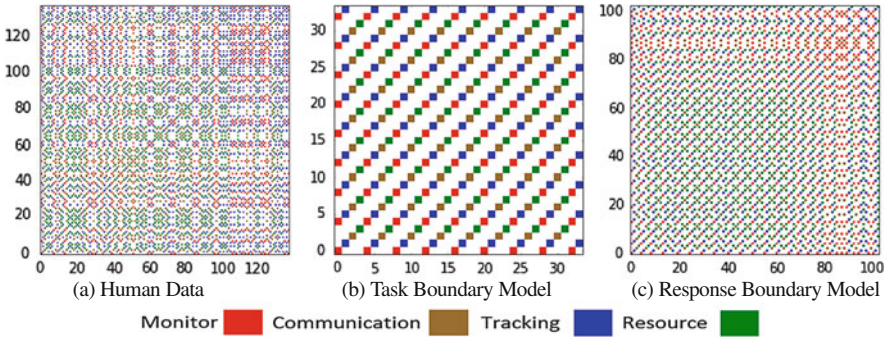


Fig. 7 Recurrence plots for 90 s of the quadrant fixation sequences from the (**a**) human, (**b**) task boundary model, and (**c**) response boundary model. Each point in the time series represents a fixation on the indicated quadrants, regardless of the fixation length. The colors for the quadrants are consistent with the scarf plots: *red* is monitoring task; *orange-brown* is communication task; *blue* is tracking task; *green* is resource management task

due to the difficulty or emphasis on the task, both models spend roughly the same amount of time on the other three tasks. This difference suggests that the strategy, perhaps related to the task prioritization, may be incorrect in the models. A second difference can be seen in the upper-left quadrant monitoring task. The attention heatmap shows that both models tend to fixate equally on all elements in the monitoring task. In contrast, the human tends to fixate on some elements longer than others, evident by the more variable total time in any given position. One possible explanation is that the model does not utilize peripheral vision to the extent that the human might be able to. For example, the status of some of the simple visual elements may be peripherally discernible, but the models strategy chooses to fixate on all elements individually. A third difference can be seen in the lower-left quadrant communication task. The human fixates on the channel frequency values (the intense total time heat pattern on the right column of the quadrant) more than the models do. Perhaps the human is taking longer to verify the frequency values than the model. The models seem to emphasize the channel names (left column) more than the frequency values. The model could be improved to include additional decision making steps to verify the frequency values, perhaps with time-consuming retrievals from declarative memory to compare with the fixated values.

The models' data visualizations do reflect some patterns observed in the human data visualizations. As stated previously, the greater similarity between the visualizations of the human data and the RBI model provides more support for that model and suggests that the RBI model may be the better candidate model on which to improve. Differences between the visualizations suggest directions for improving the models, which may include changes in model strategy (e.g., improving task prioritization) or details in the underlying theories (e.g., utilizing a newer theory of eye movements, like [14]).

Animation provides an overview and high-level understanding of the differences between the embodied eye movements produced by the two models. Very soon into the playback, it becomes clear that the TBI model fixates for long periods on the continuous object tracking task (upper right quadrant, reflected in the greater total time (red intensity) in the attention heatmap over that task for TBI in Fig. 4) and fixates on the communication task less frequently, which is contrary to our original predictions. This might be a result of the continuously moving target shifting the tracking goal state. Consequently, the model acts like the task is not complete, and it continues to execute the tracking while ignoring activity in the other tasks. On the other hand, the RBI model shows more saccades between quadrants, but often appears to overshoot the target locations across quadrants and has to make extra corrective movements.

When the human data is brought into the mix with the virtual humans, a different shortcoming of the models is highlighted. The visual world representation in which the models operate is restricted to the mMAT-B visicon representation. Thus, aside from some occasional movement errors overshooting the quadrant edges, all eye movements occur within the mMAT-B boundaries. However, the human data contains points in time when the operator looks off the task screen down to the key board (probably to ensure hitting the correct keys). This is not well captured in the Dashboard. But the fixation beams in the virtual world exaggerate these movements, as they swing below the task screen in ways that only make sense if the human had moved his/her head downward, looking below the screen. Thus, virtual embodiment of cognitive models can allow for behaviors that match the natural human patterns but that might extend beyond the explicit characteristics of the task encoding, to reflect the performance of tasks in the real world.

Our understanding of the eye movement behaviors is augmented by the Module Activity gauge in the Dashboard, which reveals additional differences in the underlying processes brought to bear on the task by the two models. It shows the TBI model recruits more temporal planning (blue bar) and motor (light orange bar) resources, while the RBI model utilizes more visual resources (bright pink bar). This information is only observable in the Module Activity gauge, demonstrating that the deepest insights about model behavior will be derived by utilizing this combination of visualization techniques.

7 Conclusion

Visual analytics user interfaces present complex task environments and often requiring multitasking for human decision making. Cognitive modelers have begun developing models for many aspects of the visual analytics process, including network visualization interpretation [24] and graph reading and comprehension [16, 18]. We show that multiple visualization approaches can both capture the model predictions for eye movements and elucidate key underlying cognitive mechanisms supporting those eye movements.

The animation further shows the model eye movements next to the actual eye movements, validating or refuting the predictions made by the model and demonstrating the realism of model behaviors. Thus, visualizing formal cognitive models provides the capability to make the complete set of desired inferences from eye-tracking data about the efficacy of visual information processing.

The focus of this chapter is not on how to best use cognitive models to predict eye movements. Rather, we wished to show how comparing visualizations of eye movements can be useful for deciding which potential model of visual interaction has better support in the empirical data. Gathering a full understanding of the model's behaviors requires a combination of visualization techniques. We also showed that comparing visualizations suggests where predictions of eye movements can be improved. Our understanding about underlying cognitive processes and the strategies people employ in visual multitasking can be better understood by comparing predictions derived from theoretical assumptions about the visual processes employed in visual tasks with each other and with human data.

7.1 Limitations

Our system is constrained by many limitations, some of which appear to arise when examining cognitive models. Of course, good inferences about cognitive mechanisms rely on working with strong, validated models of the task under evaluation. In many systems, the model is compared quantitatively to human data, which our qualitative toolkit does not do. We mitigate this limitation with the belief that the visualizations presented herein are an important part of a tool suite meant to augment the model evaluation and validation process. Because we have leveraged popular web-based libraries for the Dashboard, additional visualization techniques could easily be incorporated for further visual comparison. In particular, we might leverage parallel scan path visualizations which have previously shown utility in correlating scan paths with cognitive processes [19] or other similar space-time visualization techniques [2]. These techniques are currently useful for static visualization environments or videos playing the exact same sequence of events. However, incorporating this approach with our present dynamic multitasking environment is challenging, because the sequence of events, which are governed by a series of random variables and the operator's stochastic inputs, will diverge after the first 30 s for every run through the task. We must develop an appropriate way to align the sequences in order to leverage parallel scan path visualizations to their full potential in this application. This alignment would additionally facilitate the application of time series metrics, like cross correlation, to the comparison process.

Because all the components are fluidly interconnected by SIMCog-JS, leveraging websockets and common communication protocols, they can be integrated smoothly into the experimental process. Also, all the analyses presented herein were completed post experiment. SIMCog-JS, however, has the potential to run a model concurrently with a human operator (or another model) on the same task interface,

and it can stream the resulting model data to the Dashboard and/or the animations in near real time. While there is no explicit reason that experimenters and model developers would not have the ability to study the models as they are running using our visualizations, we have not thoroughly examined this ability.

7.2 Future Work

The work presented in this chapter has focused on qualitative methods to examine the similarities and differences between predicted eye movements from two cognitive models, or between predicted and observed eye movements. Quantitative methods, including descriptive statistics of eye movements (e.g., quantifying scanpath differences, or analyzing the number and duration of fixations), should also be used to compare predicted and human data. Extending the Model Visualization Dashboard to include such analyses is left to future work.

In addition, we plan to explore utilizing our system with other frameworks that facilitate the prediction of eye movements. While the system presented in this chapter provides novel facilities for evaluating the visual behavior of cognitive models, it does not facilitate the development of said models. Frameworks exist to support the development of cognitive models that interact with real visual interfaces [25] and graph visualizations of data [20]. Combining SIMCog-JS's ability to translate screen elements into the model visicon combined with Raschke and colleagues' tools for generating ACT-R code based on visualization ontologies [19, 20] may help speed the development of additional candidate strategies or generalize these approaches to other visual task/experiment environments. Integrating frameworks that facilitate the development of cognitive models with frameworks that facilitate the evaluation of cognitive models may lead to more robust predictions of eye movements and other measurable behaviors with novel visualizations.

Finally, we plan to examine the utility of our visualizations when comparing model-to-model and human-to-model data. Formal user studies will help determine which Dashboard and virtual human visualizations provide the most insight for analysts. In addition, we plan on examining whether the real-time visualization of cognitive model eye movements (as opposed to post-hoc analysis from recorded data) improves and accelerates the development and evaluation of such models.

Acknowledgements The authors thank Dr. Megan Morris and Mr. Jacob Kern for their assistance in processing the Tobii Glasses data, and Dr. Dustin Arendt for recurrence plot inspiration. The views expressed in this paper are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government. This work was supported by AFOSR LRIR to L.M.B. Distribution A: Approved for public release; distribution unlimited. 88ABW Cleared 08/26/2015; 88ABW-2015-4021.

References

1. Anderson, J.R., Bothell, D., Byrne, M.D., Douglass, S., Lebiere, C., Qin, Y.: An integrated theory of the mind. Psychol. Rev. **111**(4), 1036–1060 (2004)
2. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl. T.: State-of-the-art of visualization for eye tracking data. In: Proceedings of EuroVis, Swansea, vol. 2014 (2014)
3. Bojko, A.A.: Informative or misleading? Heatmaps deconstructed. In: Jacko, J.A. (eds.) Human-Computer Interaction. New Trends, pp. 30–39. Springer, Berlin/Heidelberg (2009)
4. Bostock, M., Ogievetsky, V., Heer, J.: D3: Data driven documents. IEEE Trans. Vis. Comput. Gr. **17**(12), 2301–2309 (2011)
5. Busemeyer, J.R., Diederich, A.: Cognitive Modeling. Sage, Los Angeles (2010)
6. Cline, J., Arendt, D.L., Geiselman, E.E., Blaha, L.M.: Web-based implementation of the modified multi-attribute task battery. In: 4th Annual Midwestern Cognitive Science Conference, Dayton (2014)
7. Courgeon, M., Rautureau, G., Martin, J.-C., Grynszpan, O.: Joint attention simulation using eye-tracking and virtual humans. IEEE Trans. Affect. Comput. **5**(3), 238–250 (2014)
8. Dam, E.B., Koch, M., Lillholm, M.: Quaternions, interpolation and animation. Technical report DIKU-TR-98/5, University of Copenhagen, Universitetsparken 1, DK-2100 Kbh (1998)
9. Halverson, T., Hornof, A.J.: A computational model of "Active Vision" for visual search in human-computer interaction. Hum. Comput. Interact. **26**(4), 285–314 (2011)
10. Halverson, T., Reynolds, B., Blaha, L.M.: SIMCog-JS: simplified interfacing for modeling cognition – JavaScript. In: Proceedings of the International Conference on Cognitive Modeling, Groningen, pp. 39–44 (2015)
11. Itti, L., Dhavale, N., Pighin, F.: Photorealistic attention-based gaze animation. In: 2006 IEEE International Conference on Multimedia and Expo, Toronto, pp. 521–524 (2006)
12. Kieras, D.E., Meyer, D.E.: An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. Hum. Comput. Interact. **12**(4), 391–438 (1997)
13. Laird, J.E.: The Soar Cognitive Architecture. MIT Press, Cambridge (2012)
14. Nyamsuren, E., Taatgen, N.A.: Pre-attentive and Attentive Vision Module. In: Proceedings of the 2012 International Conference on Cognitive Modeling, Berlin, pp. 211–216 (2012)
15. Patterson, R.E., Blaha, L.M., Grinstein, G.G., Liggett, K.K., Kaveney, D.E., Sheldon, K.C., Havig, P.R., Moore, J.A.: A human cognition framework for information visualization. Comput. Gr. **42**, 42–58 (2014)
16. Peebles, D.: A cognitive architecture-based model of graph comprehension. In: Rußwinkel, N., Drewitz, U., van Rijn, H. (eds.) 11th International Conference on Cognitive Modeling, Berlin, pp. 37–42 (2012)
17. Peebles, D.: Strategy and pattern recognition in expert comprehension of 2×2 interaction graphs. Cognit. Syst. Res. **24**, 43–51 (2013)
18. Peebles, D., Cheng, P.C.-H.: Modeling the effect of task and graphical representation on response latency in a graph reading task. Human Factors: J. Hum. Fact. Ergon. Soc. **45**(1), 28–46 (2003)
19. Raschke, M., Blascheck, T., Richter, M., Agapkin, T., Ertl, T.: Visual analysis of perceptual and cognitive processes. In: 2014 International Conference on Information Visualization Theory and Applications (IVAPP), Lisbon, pp. 284–291 (2014)
20. Raschke, M., Engelhardt, S., Ertl, T.: A framework for simulating visual search strategies. In: Proceedings of the 11th International Conference on Cognitive Modeling, Ottawa, pp. 221–226 (2013)
21. Richardson, D.C., Dale, R.: Looking to understand: the coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. Cogn. Sci. **29**(6), 1045–1060 (2005)
22. Salvucci, D.D.: An integrated model of eye movements and visual encoding. Cogn. Syst. Res. **1**(4), 201–220 (2001)

23. Salvucci, D.D., Taatgen, N.A.: Threaded cognition: an integrated theory of concurrent multitasking. Psychol. Rev. **115**(1), 101–130 (2008)
24. Schoelles, M., Gray, W.D.: Speculations on model tracing for visual analytics. In: Proceedings of the 12th International Conference on Cognitive Modeling, Ottawa, pp. 406–407 (2013)
25. Teo, L.-H., John, B.E., Blackmon, M.H.: CogTool-Explorer: a model of goal-directed user exploration that considers information layout. In: Proceedings of the Conference on Human Factors in Computing Systems, Texas, pp. 2479–2488 (2012)
26. Wang, L.-C., Chen, C.: A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. IEEE Trans. Robot. Autom. **7**(4), 489–499 (1991)
27. Wolfe, J.M.: Guided Search 4.0. In: Gray, W.D. (ed.) Integrated Models of Cognitive Systems, pp. 99–119. Oxford University Press, Oxford/New York(2007)
28. Zhang, Y., Hornof, A.J.: Understanding multitasking through parallelized strategy exploration and individualized cognitive modeling. In: Conference on Human Factors in Computing Systems, New York, pp. 3885–3894 (2014)

Word-Sized Eye-Tracking Visualizations

Fabian Beck, Tanja Blascheck, Thomas Ertl, and Daniel Weiskopf

Abstract In user studies, eye tracking is often used in combination with other recordings, such as think-aloud protocols. However, it is difficult to analyze the eye-tracking data and transcribed recordings together because of missing data alignment and integration. We suggest the use of word-sized eye-tracking visualizations to augment the transcript with important events that occurred concurrently to the transcribed activities. We explore the design space of such graphics by discussing how existing eye-tracking visualizations can be scaled down to word size. The suggested visualizations can optionally be combined with other event-based data such as interaction logs. We demonstrate our concept by a prototypical analysis tool.

1 Introduction

Eye-tracking data recorded in user studies is commonly analyzed using statistical methods. Visualizations depicting the data complement these methods by supporting more exploratory analysis and providing deeper insights into the data. Visualization research nowadays provides a body of techniques to visually represent the spatial and temporal dimensions of the recorded eye movements [7]. Eye-tracking data, however, is only one of many data streams—such as video, audio, and user interactions—that are usually recorded during an experiment. For instance, when applying a think-aloud protocol, a transcript of the oral statements is a particularly rich source that could explain the behavior of the participant on a higher level. To support an analyst to leverage the full potential of the recordings, it is important to integrate all streams of information within a single approach.

In this work, we focus on the integration of transcribed statements of individual participants and eye-tracking data into a visually augmented user interface (Fig. 1). Unlike other visualization approaches (Sect. 2), we handle the transcribed text as a first class entity which we complement with *word-sized eye-tracking visualizations*

F. Beck (✉) • T. Blascheck • T. Ertl • D. Weiskopf
University of Stuttgart, Stuttgart, Germany
e-mail: fabian.beck@visus.uni-stuttgart.de; tanja.blascheck@vis.uni-stuttgart.de; thomas.ertl@vis.uni-stuttgart.de; daniel.weiskopf@visus.uni-stuttgart.de

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_7

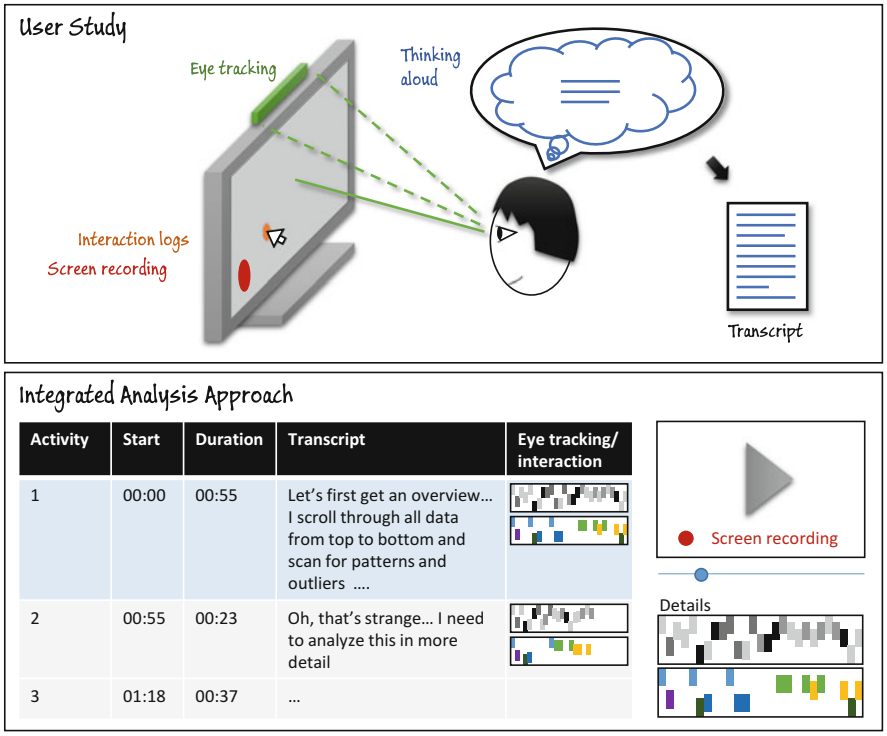


Fig. 1 Illustration of our approach to analyze transcribed recordings of user studies (e.g., based on think-aloud protocols): we integrate word-sized eye tracking and interaction visualizations into a tabular representation of the transcript and provide screen recordings and enlarged visualizations on demand

in a tabular chronological representation (Sect. 3). We systematically explore the design space of these word-sized visualizations, also known as *sparklines* [49], for eye-tracking data by discussing how existing eye-tracking visualizations can be scaled down to word size (Sect. 4). Similar visualizations can be used for representing interaction logs; the small size of visualizations allows us to combine multiple eye-tracking and interaction visualizations within a user interface. We implemented a prototype of the suggested user interface (Fig. 1, bottom) as a details-on-demand view for a visual analytics framework for eye-tracking data [6] (Sect. 5). To illustrate the applicability of our approach, we used the implementation to reanalyze and detail the results of an eye-tracking study (Sect. 6). A discussion sheds light on strengths, shortcomings, and other areas of application of our approach (Sect. 7). We see our main contributions in designing novel word-sized variants of established eye-tracking visualizations and demonstrating how these can be leveraged as part of an interactive transcript-focused analysis tool.

2 Related Work

There are various approaches to visualize eye-tracking data as Blascheck et al. [7] surveyed. Those focusing the analysis to an individual participant are closely related to our work, for instance, approaches that represent the spatial coordinates of fixations and saccades [13, 21, 35] or approaches that abstract this data to fixations on *areas of interest* (AOIs) and transitions between those [12, 15, 22, 25, 28, 41]. Also, a number of visualizations of interaction logs are available, for instance, for interactions of software developers in IDEs [36], interactions with visualization systems [17, 34, 43], or provenance information in scientific workflows [23]. However, only few approaches integrate eye-tracking or log visualizations with transcribed experiment recordings: Holsanova [26] connects transcribed picture descriptions with picture viewing data on a simple timeline showing both text and events. Franchak et al. [19] extend such a timeline with other events, in their case, interactions of infants with their environment. ChronoViz [50] includes a transcript view complementing a separate timeline view of eye-tracking data and other event-based data. Blascheck et al. [6] combine eye-tracking and interaction data in an extended timeline; the transcript is retrievable on demand only for individual time spans. Our approach, in contrast to these, puts a greater focus on text and handles eye-tracking and interaction data only as context of the transcript.

A common method for integrating text and visualization—in particular when the text should not only be a supplement to the visualization—are word-sized graphics, also called *sparklines* [49]. They can be integrated in all textual representations, such as natural-language text [20, 49], tables [49], source code [2, 4], visualizations [9, 33], or user interfaces [3]. In this paper, we integrate them into columns of a tabular representation as additional information for transcribed experiment recordings. Being a kind of scaled-down information visualization, sparklines might represent any kind of abstract data, however, only under restricted space constraints. To the best of our knowledge, sparklines have not been used so far for representing eye-tracking or interaction log data.

There are annotation and coding tools for transcribed experiment recordings. In the context of psycholinguistics, ELAN [11, 46] supports the analysis of orthographic and phonetic transcriptions. Another tool for linguistic analysis of spoken text is ANVIL [29]. It allows the integration of multimodal audiovisual material and was later extended to include spatiotemporal information of videos [30] and motion capturing [31]. None of these tools, however, supports the analysis of eye-tracking and interaction data along with the text.

3 Setting

Our goal is to provide an analysis tool that enriches a transcribed experiment recording (e.g., from a think-aloud protocol) with eye-tracking information. We focus on analyzing a single participant at a time, for instance, as part of a data

exploration step or a systematic coding of performed activities. The integrated visualization, in addition to text, should enable the analyst to make informed data analysis and coding decisions without having to switch between multiple tools or visualizations.

We assume that a transcript is divided into *activities* having a precise start and end time. The stimulus used in an experiment can either be static or dynamic. In the dynamic case, we want to be flexible enough to support video stimuli as well as interactively changeable stimuli such as user interfaces. A visual encoding of interaction logs is a secondary goal for our approach. Interaction events typically carry a timestamp when a participant triggered them, a spatial position that describes their location, and can be classified into different abstract categories such as *selection*, *encoding*, *navigation*, etc.

We assume that the eye-tracking data consists of a sequence of *fixations* with spatial coordinates as well as start and end times; *saccades* describe quick eye movements between individual fixations. Some of the visualizations discussed in the following require that a stimulus has been annotated with *areas of interest* (AOIs), summarizing sets of fixations into spatial groups. Individual transitions between AOIs can be considered as a graph, either aggregated over time as a static graph or reflecting the temporal order of transitions [12].

Formally, we define a sequence of fixations as $F = (f_1, \ldots, f_n)$ where $f_i = (x_i, y_i, t_{1_i}, t_{2_i}, c_i)$ describes an individual fixation as a 5-tupel: x_i and y_i denote the coordinates of the fixation, t_{1_i} and t_{2_i} are the start and end times, and c_i refers to a categorial attribute, such as an AOI. A derived attribute is the duration $t_{d_i} := t_{2_i} - t_{1_i}$. Moreover, we define an aggregated duration Δ as a function of different arguments summing up all durations t_{d_i} for f_i in F under certain conditions:

- for $\Delta(x, y)$, in an area around (x, y) (with constants a_x, a_y)

$$\Delta(x, y) = \sum_{f_i \in F_{x,y}} t_{d_i}, \quad F_{x,y} = \{f_i \text{ in } F | x \leq x_i < x + a_x \wedge y \leq y_i < y + a_y\},$$

- for $\Delta(x)$, in an area around x (with constant a_x)

$$\Delta(x) = \sum_{f_i \in F_x} t_{d_i}, \quad F_x = \{f_i \text{ in } F | x \leq x_i < x + a_x\},$$

- for $\Delta(t, x)$, in a spatio-temporal area around (t, x) (with constants a_x, b; fixations first need to be split into separate fixations at $t+b$ as long as there exists a fixation that spans across one of these interval limits)

$$\Delta(t, x) = \sum_{f_i \in F_{t,x}} t_{d_i}, \quad F_{t,x} = \{f_i \text{ in } F | t \leq t_{1_i} \wedge t_{2_i} < t + b \wedge x \leq x_i < x + a_x\}, \text{ and}$$

- for $\Delta(c)$, with a specific category c

$$\Delta(c) = \sum_{f_i \in F_c} t_{d_i}, \quad F_c = \{f_i \text{ in } F | c = c_i\}.$$

Analogously, function σ counts the number of fixations according to the same arguments and conditions. All functions depending on x can be formulated analogously for y.

To define a temporal transition graph $G = (V, E)$, we use the categories as nodes ($V = \{c_1, \ldots, c_n\}$) and insert edges $e = (c_i, c_{i+1}; t_{2_i}, t_{1_{i+1}})$ to E for every pair of subsequent fixations f_i, f_{i+1} in F. Hence, the graph represents subsequent fixations in transition edges between categories, also encoding the time that the transtion takes (the two timestamps t_{2_i} and $t_{1_{i+1}}$ mark the end of the first fixation and the beginning of the following). We further derive the time-aggregated graph $G' = (V, E')$ by adding edges $e' = (c_j, c_k; w_{j,k})$ to E' where weight $w_{j,k}$ is the number of edges connecting the two categories c_j and c_k in E, if $w_{j,k} > 0$.

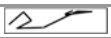
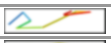
Our solution as outlined in Fig. 1 (bottom) is based on representing the transcript in a table, showing one activity per line in chronological order. Besides a column containing the actual transcript text, additional columns provide context about timing, eye tracking, and interaction events that happened during the respective activity. Since the tabular representation does not allow us to integrate large visualizations, we use *word-sized eye-tracking visualizations*. Due to the division of time into short activities, each sparkline only needs to show a small amount of data. As an additional help to make the visualizations more readable, a larger version of each word-sized graphics is retrievable on demand as part of a sidebar. The sidebar also allows us to show the recorded video stream of a specific activity, with eye-tracking and interaction data potentially overlaid.

4 The Design Space of Word-Sized Eye-Tracking Visualizations

A central element of our approach is the representation of eye-tracking data as word-sized visualizations. Since many approaches already exist for visualizing this data in normal-sized graphics [7], we take these as a starting point for developing word-sized variants showing similar data. This transformation usually requires one to simplify the visualization approach: in particular, one cannot, or at least should not, label visual objects with text, use thin lines or border lines for objects, waste space by separating objects using white space, or show 3D graphics. Moreover, a sparkline—like a word—usually has a *panorama format*, being limited to the line height of the text but having some space on the horizontal axis.

To explore the design space of those visualizations in a systematic way, we analyze all eye-tracking visualization techniques Blascheck et al. [7, Table 1] surveyed and try to transfer each approach to a word-sized visualization. Since we only target at visualizing the data recorded for a single participant, we exclude all visualizations focusing on comparing or aggregating multiple participants. Further, we are not able to suggest meaningful word-sized variants of some techniques, in particular, because of the use of 3D views [1, 18, 32, 39], the original stimulus [16, 27, 42] (the stimulus usually is too complex to be represented within a sparkline), circular

Table 1 Design space of *word-sized eye-tracking visualizations*: data dimensions are mapped to visual attributes of the graphical representations; some visualizations are only defined for coordinate x, but there always exist equivalent ones for coordinate y; variables with index i refer to a specific fixation f_i, whereas variables without an index or with a different index are defined by the interval sizes and categories used for the visualization

Visualization	Data	Encoding	X-axis	Y-axis	Color	Ref.
Point-based visualizations						
P1 	Space	Lines	x_i	y_i	–	[24, 38]
P2 	Space	Cells	x	y	Dur. $\Delta(x, y)$	[24, 35]
P3 	Space	Bars	x	Freq. $\sigma(x)$	Dur. $\Delta(x)$	
P4 	Space-time	Cells	Time t	x	Dur. $\Delta(t, x)$	[21, 51]
P5 	Space-time	Lines	x_i	y_i	Time t_{1_i}	
P6 	Space-time	Arcs	x_i	Direct. $x_{i+1} - x_i$	Time t_{1_i}	[13]
AOI-based visualizations						
A1 	AOI statistics	Bars	Freq. $\sigma(c)$ or dur. $\Delta(c)$	AOI c	AOI c	
A2 	AOI seq.	Columns	Events i	–	AOI c_i	[24]
A3 	AOI seq.	Boxes	Events i	AOI c_i	AOI c_i	
A4 	AOI seq.	Boxes	Events i	AOI c_i	Dur. t_{d_i}	[41]
A5 	AOI seq.	Boxes	Time t_{1_i}, t_{2_i}	AOI c_i	AOI c_i	[15, 28]
A6 	AOI trans.	Arcs	Trans. e'	Direct. of e'	Weight $w_{j,k}$	[37]
A7 	AOI trans.	Lines	Trans. e'	Direct. of e'	Weight $w_{j,k}$	[12]
A8 	AOI trans.	Cells	AOI c_j	AOI c_k	Weight $w_{j,k}$	[22]

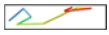
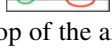
Legend: seq.–sequence; trans.–transition; freq.–frequency; dur.–duration; direct.–direction; ref.–references

layouts [8, 27, 40, 45] (advanced circular layouts are hard to fit to the elongated format of a sparkline), or a specialization to particular kinds of stimuli [5, 48]. As a result, we come up with a list of visualization techniques that can be adequately transferred to miniaturized graphics. Below, we discuss all these miniaturized visualization techniques by showing an example embedded in the text and defining the specific visual encoding in Table 1 based on the formalism introduced in Sect. 3. We furthermore describe the modifications needed when using the visualizations as word-sized graphics. All visualizations shown in this section are manually created drafts encoding artificial data. Some of them are implemented as examples in our prototypical analysis tool (Sect. 5).

4.1 Point-Based Visualizations

Each fixation f_i is assigned a pair of coordinates (x_i, y_i) on the stimulus that represents the estimated location a participant looked at. This information is a rich data source for interpreting eye movement data, together with durations and saccades between fixations.

Space. Focusing on the spatial part of the data, the standard representations of eye-tracking data are *scan paths* and *heat maps*. Scan path visualizations simply overlay the trajectory of the gaze $((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n))$ onto the stimulus [38], often encoding fixations as circles scaled according to their duration t_{d_i} [24]. For the word-sized variant, we do not show the stimulus or fixations, but just plot the trajectory as a line (P1 ). In contrast, heat maps, also called *attention maps*, aggregate fixation durations for spatial coordinates (x_i, y_i), which are color-coded and overlaid onto the stimulus [24, 35]. For a word-sized attention map, we suggest plotting a coarsely gridded map [24] into the sparkline representation and encoding the duration $\Delta(x, y)$ in the darkness of the grid cells (P2 ). As an alternative, we could focus on only one spatial axis (either x or y), again encode duration (here, either $\Delta(x)$ or $\Delta(y)$) in the color, and use bar charts to encode another metric, such as the frequency of fixations $\sigma(x)$ or $\sigma(y)$ within the respective area (P3 ). Spatial information can also be restricted otherwise to make them representable at small scale, for instance, encoding angles of the trajectory in radial diagrams [21].

Space and Time. The temporal sequence of fixations is also important for some analysis scenarios. Mapping time t to a spatial dimension, however, requires the encoding of spatial information to be limited [21, 51]. For instance, using the longer x-axis as a timeline, the y-axis could encode one of the spatial coordinates of the fixations (either x or y) while darkness indicates the distribution of fixation durations, here, either $\Delta(t, x)$ or $\Delta(t, y)$ (P4 ). We can also extend scan paths with temporal information by using the edge color for encoding time (P5 ). This is similar to *Saccade Plots* [13], which show saccades (i.e., the jumps between fixations) at the side of a stimulus. Leaving out the stimulus, we could use a similar approach within a sparkline plotting a spatial coordinate (either x_i or y_i) on the x-axis and connecting points with arcs according to observed saccades (P6 )—like in the original approach, arcs are directed from left to right on top of the axis, whereas arcs in the opposite direction are below.

4.2 AOI-Based Visualizations

AOIs abstract from the exact location of fixations to semantic regions on a stimulus, which an analyst usually defines manually. AOIs also allow us to build a transition graph connecting the AOIs according to the sequence they were looked at. We assume for the following visualizations that we have to handle five to ten different

AOIs. Due to the limited size of our visualizations, most of the suggested approaches do not scale to more AOIs, but according to our experience, ten AOIs suffice for the majority of application scenarios.

AOI Statistics. One of the simplest AOI-based visualizations is to depict the frequency $\sigma(c)$ or total duration $\Delta(c)$ each AOI c was fixated, for instance, in a line or bar chart. Such diagrams can be directly transferred into word-sized graphics. We decide to use bar charts because lines are harder to perceive if only little space is available. We use the y-axis to distinguish AOIs to have more spatial resolution for reading the value from the x-axis and redundantly color-code the AOIs to improve the discernibility of the bars (A1 ).

AOI Sequences. The temporal sequence of viewed AOIs reveals, on the one hand, what a participant saw and, on the other hand, in which order. This sequence of AOIs $(c_1, c_2, \ldots, c_n)$ might be visually encoded in any list representation showing, for instance, the logical temporal sequence of events from left to right. This has been done in various eye-tracking visualizations, for instance, connecting subsequent AOI fixations by lines and encoding the AOI fixation durations in node sizes [41] or in the horizontal length of a line [28]. In a sparkline, the sequence is easily visualized as a sequence of blocks each representing an AOI event. The different AOIs might be discerned by color (A2 ) or redundantly as a combination of position and color (A3 ). When the duration of each AOI fixation t_{d_i} is important, it can be encoded in the darkness of the boxes if the position encoding is used for discerning AOIs (A4 ) or a linear timeline can be employed scaling the width of the boxes according to the time span from t_{1_i} to t_{2_i} (A5 ) [15, 28].

AOI Transitions. Transitions between AOIs might also be depicted as a graph G' with AOIs as nodes V, and aggregated transition frequencies as weighted links E' [28]. Considering the temporal dimension of the data as well, the aggregated static graph becomes dynamic and might be visualized by animation- or timeline-based dynamic graph visualization approaches [12]. Graphs are, however, difficult to represent as a sparkline because nodes and links require a certain amount of 2D space to be discernible. Arranging the nodes in only one dimension simplifies the problem: like in *ArcTrees* [37], we draw nodes V on a vertical axis connected by arcs according to transitions $e' = (c_j, c_k; w_{j,k})$, having the weight $w_{j,k}$ encoded in the link darkness (A6 ). A more scalable variant is the *Parallel Edge Splatting* approach [14], which was already applied to AOI transitions graphs [12]: the graph is interpreted as a bipartite graph duplicating the nodes V to two horizontal axes; all transitions E' are drawn as straight lines connecting a source AOI c_j at the top to a target AOI c_k at the bottom (A7 ). Furthermore, matrix representations of graphs are space-efficient and have already been employed to represent eye-tracking data [22]. A transformation into a sparkline is straightforward, for instance, color-coding the AOIs (first row and column) in addition to the transition weights

$w_{j,k}$ within the matrix cells (A8 ). A limitation, however, is that they are inherently quadratic—although they can be stretched to fill an arbitrary rectangle, additional vertical space does not necessarily improve their readability.

4.3 Combination and Extension

The suggested visualizations provide a flexible framework for encoding eye-tracking data. To decide between the different encodings is not an either–or decision because visualizations can be combined with each other to build an even more expressive analysis tool. Moreover, the framework of visualizations might be extended with only little adaption to also depict interaction data.

Juxtaposing Visualizations. Since word-sized visualizations are space-efficient, they can easily be juxtaposed within one line, each graphic providing a different perspective onto the data. For instance, it could be useful to combine a point-based and an AOI-based visualization:  . If the application scenario allows the use of several lines en bloc, a vertical stacking of the sparklines (i.e., placing them on top of each other) is possible. To align both visualizations, the x-axes should have the same encoding, for example, a color-coded sequence of AOIs combined with a duration encoding:

Interaction Data. Interaction data shares characteristics to eye-tracking data: Much like fixations, interactions are temporal events on the same experiment time dimension. They can be classified according to their type into categories or assigned to AOIs based on their location. Also, transitions between interactions might be derived from the sequence of logged events. One difference, however, is that typical interaction events do not have a duration; they only get a duration if they are abstracted to longer sequences of semantically linked interactions. Hence, an interaction can be represented as a 5-tupel like a fixation $f_i = (x_i, y_i, t_{1_i}, t_{2_i}, c_i)$, often with $t_{1_i} = t_{2_i}$. The general similarity between the two data streams now allows us to reuse most of the suggested *word-sized eye-tracking visualizations* for interaction data; even those representing durations are applicable if we just assume constant durations $t_{d_i} > 0$. Furthermore, the discussed horizontal and vertical juxtaposition of these sparklines provides an easy way of integrating both data sources within one user interface.

5 Prototype Implementation

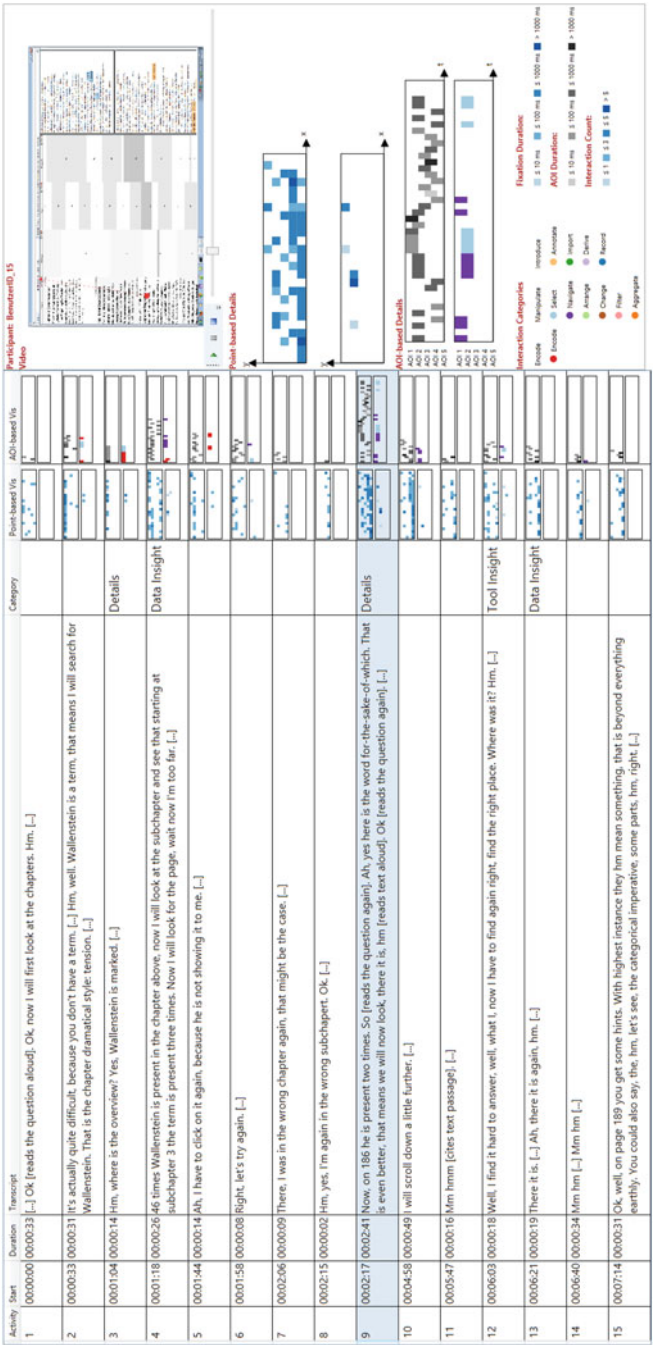
We implemented the approach as a detail view of a larger visual analysis framework for eye-tracking studies [6]. The visual analysis framework is intended to support the joint analysis of eye-tracking and interaction data. In the original implementation, think-aloud data was added to enrich the other two data sources. In the new detail view, in contrast, we intend to present the think-aloud protocol in detail and enrich it with eye-tracking and interaction data. This prototype is a proof of concept implementing two AOI-based and two point-based versions of word-sized visualizations.

Figure 2 shows a screenshot of our prototype, depicting data of one participant in a temporal order. A tabular view represents the main part of the prototype. For each verbal statement, word-sized visualizations are shown, in one column the two point-based visualizations, in another the two AOI-based ones. In both columns, the visualizations for eye movements and interactions are juxtaposed vertically, showing the eye-tracking visualization above the interaction visualization.

The point-based visualizations are gridded attention maps (Table 1, P2 ) or, respectively, maps showing the spatial distribution of interactions. We divided the stimulus into 25 columns and five rows. For each cell, we counted the fixation durations and the count of interactions and color-coded the cells accordingly. The color coding was obtained from ColorBrewer [10], using a sequential, single-hue blue color and a gradation of four (fixation duration $\leq$10, $\leq$100, $\leq$1000, and $>$1000 ms; interaction count $\leq$1, $\leq$3, $\leq$5, and $>$5).

Our AOI-based visualizations (Table 1, A4 ) represent each AOI as a row of rectangles. Since only one AOI is active at a time, we assign a height to each rectangle greater than the row height to increase the size of the rectangles (which improves color perception). In the eye-tracking visualization at the top, for each individual AOI fixation, the duration is calculated and the AOI rectangle is colored based on the duration. We chose a sequential gray scale and a logarithmic gradation of four (AOI fixation duration $\leq$10, $\leq$100, $\leq$1000, and $>$1000 ms). For the visualization of interaction data below, interactions are assigned to AOIs and the color is determined by the categorical interaction category. For example, an interaction from the category *encode* is shown in red, a *select* interaction in light blue, and a *navigate* interaction in purple. The interactions are temporally aligned with the AOI fixations, thus, representing an interaction at the point in time of its corresponding AOI fixation.

Based on the eye movement and interaction data depicted in the word-sized visualizations, an analyst adds categories to the activities. Additionally, rows and columns might be reordered. On the right side of the prototype, a video playback is shown for further reference. The playback might be combined with an animated representation of the eye-tracking or interaction data, in our case, a dynamic scan path overlay obtained from Tobii Studio. Below the video, the visualizations of a selected row are shown enlarged and annotated with labels.

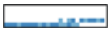


Activity	Start	Duration	Transcript	Category
1	0000000	0000033	[...] Ok [reads the question aloud]. Ok, now I will first look at the chapters. Hm. [...]	
2	0000033	0000031	It's actually quite difficult, because you don't have a term. [...] Hm, well. Wallenstein is a term, that means I will search for Wallenstein. That is the chapter dramatical style: tension. [...]	
3	0001:04	0000:14	Hm, where is the overview? Yes, Wallenstein is marked. [...]	Details
4	0001:18	0000:26	46 times Wallenstein is present in the chapter above, now I will look at the subchapter and see that starting at subchapter 3 the term is present three times. Now I will look for the page. wait now I'm too far. [...]	Data Insight
5	0001:44	0000:14	Ah, I have to click on it again, because he is not showing it to me. [...]	
6	0001:58	0000:08	Right, let's try again. [...]	
7	0002:06	0000:09	There, I was in the wrong chapter again, that might be the case. [...]	
8	0002:15	0000:02	Hm, yes, I'm again in the wrong subchapert. Ok. [...]	
9	0002:17	0002:41	Now, on 186 he is present two times. So [reads the question again]. Ah, yes here is the word for-the-sake-of-which. That is even better, that means we will now look, there it is. hm [reads text aloud]. Ok [reads the question again]. [...]	Details
10	0004:58	0000:49	I will scroll down a little further. [...]	
11	0005:47	0000:16	Mm hmm [cites text passage]. [...]	
12	0006:03	0000:18	Well, I find it hard to answer, well, what I, now I have to find again right, find the right place. Where was it? Hm. [...]	Tool Insight
13	0006:21	0000:19	There it is. [...] Ah, there it is again. hm. [...]	Data Insight
14	0006:40	0000:34	Mm hm [...] Mm hm [...]	
15	0007:14	0000:31	Ok, well, on page 189 you get some hints. With highest instance they hm mean something, that is beyond everything earthly. You could also say, the, hm, let's see, the categorical imperative, some parts. hm, right. [...]	

Fig. 2 Screenshot of the prototype implementation of our approach that shows the think-aloud protocol in a tabular fashion, containing an activity ID, start time and duration, a transcript of the audio recording, a category, and point-based as well as AOI-based word-sized visualizations for eye movements (top) and interactions (bottom). The sidebar provides a video replay of an enriched stimulus and enlarged word-sized visualizations of a selected activity

6 Application Example

For a small use case example, we re-analyzed data from a user study testing a visual text analysis tool [6] (Participants 13 and 15, transcript partly translated from German to English). For Participant 15 (Fig. 2), we first explore the data trying to get an overview. We find that in the point-based visualization at the beginning most of the fixations are in the upper part of the stimulus (Activities 1  , 2, 3, 5), whereas later, most of the fixations are in the lower part of the stimulus (Activities 13, 14, 15 ). In the AOI-based sparkline, it becomes apparent that, at the beginning, the participant used mostly *encode* and *select* interactions in the first two AOIs (Activities 1, 2  , 3, 4, 5, 6) while focusing mostly on AOIs 1 and 2  . At the end, the participant used more *navigate* interactions (Activities 8, 9  , 10, 12) and was looking at AOIs 4 and 5 more often  .

We can observe a similar behavior when looking at another participant (Participant 13). This participant also focused on the upper part at the beginning (Activities 1  , 2, 3, 4) and at the lower part at the end (Activitites 12, 13, 14, 15, 16  , 17). However, this participant has not interacted with the system as intensely as Participant 15 did, only at the beginning (Activities 1, 2  , 4, 5) and once more at the end (Activity 16 ). In the meantime, this participant was switching focus from top to bottom and looking at AOIs 1, 2, and 5 mainly (Activities 5  , 6, 7, 9, 10).

These kinds of analyses allow us to classify the participants' activities and manually assign categories in the respective table column (Fig. 2). To categorize some of the data, we use categories Saraiya et al. [44] define for visualization systems, namely *overview*, *patterns*, *groups*, and *details*. Smuc et al. [47] add the categories *data insights* and *tool insights*. For example, if we want to categorize Activity 9, highlighted in Fig. 2, we can look at the transcript and read that the participant found a specific text passage with the word the participant was searching for and some related terms after a while. Looking at the point-based word-sized graphics of the fixations  shows that the participant was looking at large parts and most AOIs  of the stimulus while using many navigate interactions and at the end the participant selected some items  . This behavior could be classified as *details*, because the participant was inspecting the system in order to find details about the analysis task the participant was solving. Thus, we can add this category to Activity 9 in the transcript.

7 Discussion

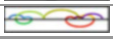
We explored the design space of *word-sized eye-tracking visualizations* systematically in the sense that we miniaturized existing eye-tracking visualizations following a taxonomy [7]. This strategy provides a variety of meaningful solutions. However, we cannot guarantee that we covered the full available design space: First, we

only focused on single-participant visualizations and might have excluded other visualizations that would be well suitable for word-sized representation. Second, we did not analyze the design space for word-sized representations independently of previous normal-sized visualizations—there might be visualizations that make sense as word-sized graphics but would not be used as normal-sized visualizations.

Word-sized visualizations, by design, have only limited space available and hence scale worse to larger data sets. By splitting up the data into short activities and present multiple word-sized visualizations, we circumvent this problem to some extent. Our application example in Sect. 6 already shows that at least the implemented visualizations scale well to a meaningful amount of data. However, if the activities become longer or if we have to handle more AOIs, visualizations that do not aggregate this information would become cluttered; word-sized graphics only leave few opportunities for improving visual scalability. According to our experience with the current implementation, we estimate that the suggested visualizations scale sufficiently up to 30 events (i.e., fixations and interactions) or 30 time intervals respectively and 10 AOIs per activity. To substantiate this estimation, Table 2 provides some mockup examples showing increasing numbers events for visualizations that do not aggregate time and examples for increasing numbers of AOIs. For the number of events, the number of crossing lines and the width of the visualization restrict their scalability. Limiting factors for the number of AOIs are the amount of discernible colors and the height or width of the visualizations.

We studied in detail the embedding of *word-sized eye-tracking visualizations* within transcribed experiment recordings of single participants. But beyond that, we see broader opportunities for application of our approach: First, the visualizations can be easily used to compare similar activities of multiple participants by juxtaposing the visualizations as demonstrated in the application example (Sect. 6). Those visualizations that aggregate time to duration or transitions (Table 1, P2, P3, A1, A6, A7, A8) can even be directly used to summarize data of several participants within a single representation. A second application is the use of our visualizations within scientific texts, for instance, to report the results of a study like demonstrated in Sect. 6. Since publications are a static medium, however, it is a restriction in this scenario that interactions are not available to view a larger version of the visualization. In general, an open question is still how well the suggested word-sized visualizations would be received by experimenters analyzing their eye-tracking data and readers seeing the visualizations within publications.

Table 2 Examples of *word-sized eye-tracking visualizations* to investigate scalability by number of events/intervals for point-based visualizations and number of AOIs for AOI-based visualizations

	10 events/intervals	30 events/intervals		5 AOIs	10 AOIs
P1			A1		
P4			A2		
P6			A7		

8 Conclusion and Future Work

With a focus on analyzing the transcribed experiment recording of a single participant, we suggested a novel approach to visually enrich the textual representation of a transcript with eye-tracking and interaction data. This data is represented in word-sized visualizations that provide different perspectives onto the data. We systematically explored the design space of *word-sized eye-tracking visualizations* and prototypically implemented the approach as a detail view of a larger visual analysis framework for eye-tracking studies.

Since our implementation is work in progress, it only partly covers the suggested visualizations yet, still lacks important interaction techniques, and only provides rudimentary support for coding. We will extend the implementation toward a full-fledged visual analysis and coding system. Moreover, we want to explore which of the suggested visualizations is most effective and efficient for analyzing the data and at the same time easy to understand for potential users. Beyond that, we are interested in exploring other application scenarios for the suggested visualizations, for instance, their use to communicate results of eye-tracking studies in scientific publications.

Acknowledgements Fabian Beck is indebted to the Baden-Württemberg Stiftung for the financial support of this research project within the Postdoctoral Fellowship for Leading Early Career Researchers.

References

1. Baldauf, M., Fröhlich, P., Hutter, S.: KIBITZER: a wearable system for eye-gaze-based mobile urban exploration. In: Proceedings of the 1st Augmented Human International Conference, AH, pp. 9:1–9:5 (2010)
2. Beck, F., Dit, B., Velasco-Madden, J., Weiskopf, D., Poshyvanyk, D.: Rethinking user interfaces for feature location. In: Proceedings of the 23rd IEEE International Conference on Program Comprehension, ICPC, pp. 151–162. IEEE (2015)
3. Beck, F., Koch, S., Weiskopf, D.: Visual analysis and dissemination of scientific literature collections with SurVis. IEEE Trans. Vis. Comput. Graph. **22**(1), 180–189 (2016)
4. Beck, F., Moseler, O., Diehl, S., Rey, G.D.: In situ understanding of performance bottlenecks through visually augmented code. In: Proceedings of the 21st IEEE International Conference on Program Comprehension, ICPC, pp. 63–72 (2013)
5. Beymer, D., Russell, D.M.: WebGazeAnalyzer: a system for capturing and analyzing web reading behavior using eye gaze. In: CHI '05 Extended Abstracts on Human Factors in Computing Systems, CHI EA, pp. 1913–1916 (2005)
6. Blascheck, T., John, M., Kurzhals, K., Koch, S., Ertl, T.: VA^2: a visual analytics approach for evaluating visual analytics applications. IEEE Trans. Vis. Comput. Graph. **22**(1), 61–70 (2016)
7. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: State-of-the-art of visualization for eye tracking data. In: EuroVis – STARs, pp. 63–82 (2014)
8. Blascheck, T., Raschke, M., Ertl, T.: Circular heat map transition diagram. In: Proceedings of the 2013 Conference on Eye Tracking South Africa, ETSA, pp. 58–61 (2013)

9. Brandes, U., Nick, B.: Asymmetric relations in longitudinal social networks. IEEE Trans. Vis. Comput. Graph. **17**(12), 2283–2290 (2011)
10. Brewer, C.A., Harrower, M.: ColorBrewer 2.0. http://www.colorbrewer.org
11. Brugman, H., Russel, A.: Annotating multimedia/multi-modal resources with ELAN. In: Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC, pp. 2065–2068 (2004)
12. Burch, M., Beck, F., Raschke, M., Blascheck, T., Weiskopf, D.: A dynamic graph visualization perspective on eye movement data. In: Proceedings of the 2014 Symposium on Eye Tracking Research & Applications, ETRA, pp. 151–158 (2014)
13. Burch, M., Schmauder, H., Raschke, M., Weiskopf, D.: Saccade plots. In: Proceedings of the 2014 Symposium on Eye Tracking Research & Applications, ETRA, pp. 307–310 (2014)
14. Burch, M., Vehlow, C., Beck, F., Diehl, S., Weiskopf, D.: Parallel Edge Splatting for scalable dynamic graph visualization. IEEE Trans. Vis. Comput. Graph. **17**(12), 2344–2353 (2011)
15. Crowe, E.C., Narayanan, N.H.: Comparing interfaces based on what users watch and do. In: Proceedings of the 2000 Symposium on Eye Tracking Research & Applications, ETRA, pp. 29–36 (2000)
16. Dorr, M., Jarodzka, H., Barth, E.: Space-variant spatio-temporal filtering of video for gaze visualization and perceptual learning. In: Proceedings of the 2010 Symposium on Eye Tracking Research & Applications, ETRA, pp. 307–314 (2010)
17. Dou, W., Jeong, D.H., Stukes, F., Ribarsky, W., Lipford, H., Chang, R.: Recovering reasoning processes from user interactions. IEEE Comput. Graph. Appl. **29**(3), 52–61 (2009)
18. Duchowski, A., Medlin, E., Courina, N., Gramopadhye, A., Melloy, B., Nair, S.: 3D eye movement analysis for VR visual inspection training. In: Proceedings of the 2002 Symposium on Eye Tracking Research & Applications, ETRA, pp. 103–155 (2002)
19. Franchak, J.M., Kretch, K.S., Soska, K.C., Babcock, J.S., Adolph, K.E.: Head-mounted eye-tracking of infants' natural interactions: a new method. In: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, ETRA, pp. 21–27 (2010)
20. Goffin, P., Willett, W., Fekete, J.D., Isenberg, P.: Exploring the placement and design of word-scale visualizations. IEEE Trans. Vis. Comput. Graph. **20**(12), 2291–2300 (2014)
21. Goldberg, J.H., Helfman, J.I.: Visual scanpath representation. In: Proceedings of the 2010 Symposium on Eye Tracking Research & Applications, ETRA, pp. 203–210 (2010)
22. Goldberg, J.H., Kotval, X.P.: Computer interface evaluation using eye movements: methods and constructs. Int. J. Ind. Ergon. **24**, 631–645 (1999)
23. Hlawatsch, M., Burch, M., Beck, F., Freire, J., Silva, C.T., Weiskopf, D.: Visualizing the evolution of module workflows. In: Proceedings of the International Conference on Information Visualisation, IV, pp. 40–49 (2015)
24. Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., Van de Weijer, J.: Eye Tracking: A Comprehensive Guide to Methods and Measures, 1st edn. Oxford University Press, Oxford (2011)
25. Holsanova, J.: Picture viewing and picture description: two windows on the mind. Ph.D. thesis, Lund University (2001)
26. Holsanova, J.: Dynamics of picture viewing and picture description. Adv. Conscious. Res. **67**, 235–256 (2006)
27. Hurter, C., Ersoy, O., Fabrikant, S., Klein, T., Telea, A.: Bundled visualization of dynamic graph and trail data. IEEE Trans. Vis. Comput. Graph. **20**(8), 1141–1157 (2014)
28. Itoh, K., Tanaka, H., Seki, M.: Eye-movement analysis of track monitoring patterns of night train operators: effects of geographic knowledge and fatigue. In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 44, pp. 360–363 (2000)
29. Kipp, M.: ANVIL – a generic annotation tool for multimodal dialogue. In: Proceedings of the 7th European Conference on Speech Communication and Technology, Eurospeech, pp. 1367–1370 (2001)
30. Kipp, M.: Spatiotemporal coding in ANVIL. In: Proceedings of the 6th International Conference on Language Resources and Evaluation, LREC, pp. 2042–2045 (2008)

31. Kipp, M.: ANVIL: The video annotation research tool. In: Jacques Durand, U.G., Kristoffersen, G. (eds.) The Oxford Handbook of Corpus Phonology, chap. 21, pp. 420–436. Oxford University Press, Oxford (2014)
32. Lankford, C.: Gazetracker: software designed to facilitate eye movement analysis. In: Proceedings of the 2000 Symposium on Eye Tracking Research & Applications, ETRA, pp. 51–55 (2000)
33. Lee, B., Henry Riche, N., Karlson, A.K., Carpendale, S.: SparkClouds: visualizing trends in tag clouds. IEEE Trans. Vis. Comput. Graph. **16**(6), 1182–1189 (2010)
34. Lipford, H.R., Stukes, F., Dou, W., Hawkins, M.E., Chang, R.: Helping users recall their reasoning process. In: Proceedings of the 2010 IEEE Symposium on Visual Analytics Science and Technology, VAST, pp. 187–194 (2010)
35. Mackworth, J.F., Mackworth, N.H.: Eye fixations recorded on changing visual scenes by the television eye-marker. J. Opt. Soc. Am. **48**(7), 439–444 (1958)
36. Minelli, R., Mocci, A., Lanza, M., Baracchi, L.: Visualizing developer interactions. In: Proceedings of the Second IEEE Working Conference on Software Visualization, VISSOFT, pp. 147–156 (2014)
37. Neumann, P., Schlechtweg, S., Carpendale, S.: ArcTrees: Visualizing relations in hierarchical data. In: Proceedings of the 7th Joint Eurographics/IEEE VGTC Conference on Visualization, EuroVis, pp. 53–60 (2005)
38. Noton, D., Stark, L.: Scanpaths in saccadic eye movements while viewing and recognizing patterns. Vis. Res. **11**, 929942 (1971)
39. Paletta, L., Santner, K., Fritz, G., Mayer, H., Schrammel, J.: 3D attention: measurement of visual saliency using eye tracking glasses. In: CHI'13 Extended Abstracts on Human Factors in Computing Systems, CHI EA, pp. 199–204 (2013)
40. Pellacini, F., Lorigo, L., Gay, G.: Visualizing Paths in Context. Technical report #TR2006-580, Department of Computer Science, Dartmouth College (2006)
41. Räihä, K.J., Aula, A., Majaranta, P., Rantala, H., Koivunen, K.: Static visualization of temporal eye-tracking data. In: Costabile, M.F., Paternò, F. (eds.) Human-Computer Interaction-INTERACT 2005. LNCS, vol. 3585, pp. 946–949. Springer Berlin Heidelberg (2005)
42. Ramloll, R., Trepagnier, C., Sebrechts, M., Beedasy, J.: Gaze data visualization tools: opportunities and challenges. In: Proceedings of the 8th International Conference on Information Visualization, IV, pp. 173–180 (2004)
43. Reda, K., Johnson, A., Leigh, J., Papke, M.: Evaluating user behavior and strategy during visual exploration. In: Proceedings of the 2014 BELIV Workshop, pp. 70–77 (2014)
44. Saraiya, P., North, C., Duca, K.: An insight-based methodology for evaluating bioinformatics visualizations. IEEE Trans. Vis. Comput. Graph. **11**(4), 443–456 (2005)
45. Schulz, C.M., Schneider, E., Fritz, L., Vockeroth, J., Hapfelmeier, A., Brandt, T., Kochs, E.F., Schneider, G.: Visual attention of anaesthetists during simulated critical incidents. Br. J. Anaesth. **106**(6), 807–813 (2011)
46. Sloetjes, H., Wittenburg, P.: Annotation by category – ELAN and ISO DCR. In: Proceedings of the 6th International Conference on Language Resources and Evaluation, LREC, pp. 816–820 (2008)
47. Smuc, M., Mayr, E., Lammarsch, T., Aigner, W., Miksch, S., Gartner, J.: To score or not to score? Tripling insights for participatory design. IEEE Comput. Graph. Appl. **29**, 29–38 (2009)
48. Špakov, O., Räihä, K.J.: KiEV: a tool for visualization of reading and writing processes in translation of text. In: Proceedings of the 2008 Symposium on Eye Tracking Research & Applications, ETRA, pp. 107–110 (2008)
49. Tufte, E.R.: Beautiful Evidence, 1st edn. Graphics Press, Cheshire (2006)
50. Weibel, N., Fouse, A., Emmenegger, C., Kimmich, S., Hutchins, E.: Let's look at the cockpit: exploring mobile eye-tracking for observational research on the flight deck. In: Proceedings of the 2012 Symposium on Eye Tracking Research & Applications, ETRA, pp. 107–114 (2012)
51. Yarbus, A.L.: Eye Movements and Vision. Plenum Press, New York (1967)

GazeGIS: A Gaze-Based Reading and Dynamic Geographic Information System

Laura G. Tateosian, Michelle Glatz, Makiko Shukunobe, and Pankaj Chopra

Abstract Location is an important component of a narrative. Mapped place names provide vital geographical, economic, historical, political, and cultural context for the text. Online sources such as news articles, travel logs, and blogs frequently refer to geographic locations, but often these are not mapped. When a map is provided, the reader is still responsible for matching references in the text with map positions. As they read a place name within the text, readers must locate its map position, then find their place again in the text to resume reading, and repeat this for each toponym. We propose a gaze-based reading and dynamic geographic information system (GazeGIS) which uses eye tracking and geoparsing to enable a more cohesive reading experience by dynamically mapping locations just as they are encountered within the text. We developed a prototype GazeGIS application and demonstrated its application to several narrative passages. We conducted a study in which participants read text passages using the system and evaluated their experience. We also explored an application for intelligence analysis and discuss how experts in this domain envision its use. Layman and intelligence expert evaluations indicate a positive reception for this new reading paradigm. This could change the way we read online news and e-books, the way school children study political science and geography, the way officers study military history, the way intelligence analysts consume reports, and the way we plan our next vacation.

1 Introduction

We propose an idea to change the way we read narratives. Most narratives we read on a daily basis have something in common. They reference toponyms (place names), the names of cities, countries, rivers, and islands, and the names of these places themselves carry their own connotations. The name of a Swiss village invokes

L.G. Tateosian (✉) • M. Glatz • M. Shukunobe
Center for Geospatial Analytics, North Carolina State University, Raleigh, NC, USA
e-mail: lgtateos@ncsu.edu; mlglatz@ncsu.edu; mshukun@ncsu.edu

P. Chopra
Department of Human Genetics, Emory University, Atlanta, GA, USA
e-mail: chopra.pankaj@gmail.com

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_8

a mountainous scene, a reference to Siberia implies the frigid temperatures there, the story taking place on a Balinese island suggests a backdrop of Hinduism, and an event in a Canadian state occurred in a first-world economic environment. Familiar place names offer underlying context for the story we are reading. Still, many place names are unfamiliar. If we do not take the time to interrupt the flow of reading to look at these places on a map, we will miss this locational context. If we do stop to seek a map, by the time we get back to the reading, we have lost track of where we were reading. In fact, we may have even forgotten the flow of the story.

Our idea to change the way we read narratives combines reading with a dynamic geographic information system. This application would provide a map and corresponding geographically derived information on the fly as the reader encounters toponyms in the text. A device would sense when a person is reading a place name in the text and display a map of the place. Pictures of the site would also appear along with related information, like population, current weather conditions and currency valuation, political climate and dominant religion, or even proximity and route to the nearest airport. The map and geo-information would be introduced in a way that allows the reader to access the supplementary material without losing their place in the text. The map and images would help us to retain and comprehend the story arc. Studies have shown improvement in recall and comprehension of text content when accompanying maps are provided [2, 24, 34]. This could change the way we read online news and e-books, the way school children study political science and geography, the way officers study military history, the way intelligence analysts consume reports, and the way we plan our next vacation.

We demonstrate this idea with a prototype called GazeGIS, a gaze-based reading and dynamic geographic information system. GazeGIS couples geoparsing and Web mapping with an eye-tracking device. Eye-tracking devices have begun to be used for interactive reading systems in other contexts. Our innovation is combining this approach with geoparsing and GIS. To contextualize our contribution, we discuss related work in geoparsing and eye tracking (Sect. 2). Then, we explain the system design and implementation in Sect. 3. To see how our users interact with our system and how they perceive it, we conducted a study, which we describe in Sect. 4. Short passages were used in the study so that reading would take only a few minutes. The content in these passages could be considered leisure reading. We were also interested to learn how domain experts would use our system for reading or writing reports to rapidly analyze complex scenarios. For a demonstration, we obtained a report from an analyst in the intelligence community and applied GazeGIS to the report. Section 5 discusses insights from this experience. In Sect. 6, we describe the extensions we implemented to incorporate feedback and lessons from our studies. This is followed in Sect. 7 by conclusions and plans for future work.

2 Related Work

Geoparsing algorithms enable place names to be extracted from text and geolocated. One approach for implementing our idea is to combine this geoparsing cabability with eye-tracking technology for eye-based interaction. When discussed in the context of related work, this idea appears to be a natural extension of previous work conducted in these areas.

2.1 Geoparsing

Parsing place names within structured data files such as tables is simply a matter of cleaning the data and discerning the pattern (e.g., it is easy to parse county names, if you know they are in the third column of a table). A different approach is needed for unstructured text like news articles or novels. Advances in natural language processing enable place names within unstructured text to be automatically detected and geolocated. The process of identifying toponym instances within a text and subsequently assigning a coordinate to each name is referred to as *geoparsing*. The first part of this process is implemented with a named-entity identifier and a *gazetteer*, a database of place names and related information. Once something is identified as the name of a place, the toponymic homonym problem may need to be resolved. Toponymic homonyms are different places that share the same names, as in Springfield, Ohio and Springfield, Rhode Island. The geoparsing algorithms usually make some assumption such as proximate locations being clustered within text, the occurrence of other small town names in the text, or if another place name within the same border is mentioned. Failing other cues, place names may be ranked by population on the assumption that larger places are mentioned more frequently. Leidner and Lieberman [25] review the workflow and challenges involved in geoparsing.

In the past, geoparsers were primarily proprietary software. Recent open source projects, such as geodict, Unlock Text, and CLAVIN, have made this technology more widely available. As a result, researchers have begun adapting geoparsing algorithms to handle specific unstructured media, such as news articles and micro-text [14, 18] and specific content domains such as historical texts and classic literature [20, 33]. Though our work does not target a particular media or content domain, the system could be modified to be tailored for specific content.

Geoparsing along with mapping tools has also begun to be conceived of as a means for visualizing collections of documents and multilingual texts [1, 30]. Geoparsing underlies the digital maps project, DIGMAP, which uses a map as part of an interactive user interface for searching digital resources [26]. Our prototype is currently designed to display a single document at a time, though it can be extended to explore databases of documents.

2.2 *Eye Tracking*

Eye tracking has a long history of application in both diagnostics and interaction [15]. From the 1970s, eye trackers have been used in controlled experiments to record eye movements and focus as users perform a cognitive task or inspect a stimulus, such as a work of art, an advertisement, or a Web page. Data visualization and cartographic design choices are also being evaluated in this way [9–11]. In these experiments, the data is recorded to be analyzed afterward. Whereas, applications for real-time eye movement consumption appeared, from the early 1990s onward, with eye trackers acting as devices for interacting with applications [8, 21]. For example, gaze focus has been used as a pointer to make selections in virtual environments [36] and to act as a typing device for people with a loss of movement [22]. These tools require the user to actively adjust eye movements to control their environment. Our application, which captures the user's natural eye movements to trigger timely responses, is termed gaze-contingent.

An early gaze-contingent reading application used the reader's gaze path to drive zoom and voice narration for items of interest in the novella, "The Little Prince" [8]. Meanwhile, diagnostic eye-tracking systems have been used extensively to study how people read in relation to cognition [32]. This work provided the foundation for several interactive eye-tracking applications that react, in real-time, to what the user is reading and how they are reading it. The GWGazer Reading Assistant is a remedial reading application that uses gaze to determine when the user hesitates over a word and provides assistance by speaking the word [35]. SUITOR harnesses gaze information to infer the reader's interest and automatically finds and displays relevant information in a scrolling display at the bottom of the screen [27]. The iDict application provides automatic translation help when users appear to be having difficulty reading in a foreign language [19]. In Text 2.0, words or phrases are associated with sound effects that are played or images that are displayed when this selection is read. Additional information like translations or explanations are presented when the user's attention indicates difficulty. Also, if skimming is detected, the document display is altered such that words that are likely to be skipped are faded [5–7]. Like our system, these systems also use figures to supplement the text, though they do not focus on geographic content. Recent work by Bektas et al. applied gaze-contingent behavior to improve performance with large geographic imagery and digital elevation model datasets by varying level of detail based on eye movements and models of human visual perception [3].

3 GazeGIS Design and Implementation

We implemented our idea in an application we call GazeGIS. GazeGIS consists of a gaze-contingent Web page application for reading text documents while an eye-tracking device computes the reader's point of gaze. We use a layout widely

employed within military history books where the story often hinges on the location of commanders and troops [13, 17]. In books such as this, passages of text are accompanied by a corresponding map on each facing page, so that the text can refer to important locations pictured opposite. Analogously, we split the main part of a web page with reading on the left and a map on the right. Of course, in the digital medium, we can use a single Web map and move or mark the map to reflect place names being referenced. To supplement the maps, we also display photographs to provide some information about the appearance of the named places. We display the images in a panel that runs across the bottom of the screen, beneath both the text and map. This approach follows a popular design for Web pages such as Esri Story Maps, which display text, maps, and images [16]. The GazeGIS web page is divided into three regions, with a box for text on the left, a box for a map on the right, and an image panel along the bottom (Fig. 1). The system augments the reading content with geographic information in the form of maps and images to orient readers to the local scenery and geographic location of a place just as their eyes encounter the location's name within the text. When the system detects that the user's gaze has reached a place name within the text, the map pans to display and tag that location, and photos taken near that location appear in the image panel.

Naturally, the reader may want to look at the map or images, and then come back to the text and find the position where they left off to resume reading there. Kern et al. observed that readers performing tasks necessitating switching back and forth between two printed documents tended to use some kind of pointer (e.g., a pencil or a finger) to keep their place [23]. They tested an analogous idea for digital environments that uses eye tracking to determine gaze location and adds graphical markers, "gazemarks", to mark recent gaze locations. In our application the user is likely to switch to the map or images when these frames are updated, that is, when the reader's gaze reaches a place name. As a gazemark, our system highlights the name of the most recently gazed place name, so that readers may inspect the map and images and easily re-orient themselves to their position in the reading. In our initial prototype the reader must use the mouse to pan and zoom the map or browse the images. Hardware, software, and gaze interaction details are described next.

3.1 Hardware

The primary hardware component of GazeGIS is an X2-60 Compact Eye Tracker from Tobii Technologies. This 60 HZ eye-tracking system is mounted below a 27-inch desktop monitor with 1920×1080 resolution. The tracker comes with adhesive for mounting on the monitor frame. We choose not to use this method of attachment in order to preserve portability. Our solution is to hold the tracker in place with a flexible cell phone tripod, a Gorillapod Grip. Mirroring the screen on a secondary overhead 65-inch Samsung wall-mounted monitor facilitates demonstrations for groups and allows researchers to observe usage unobtrusively (Fig. 1).

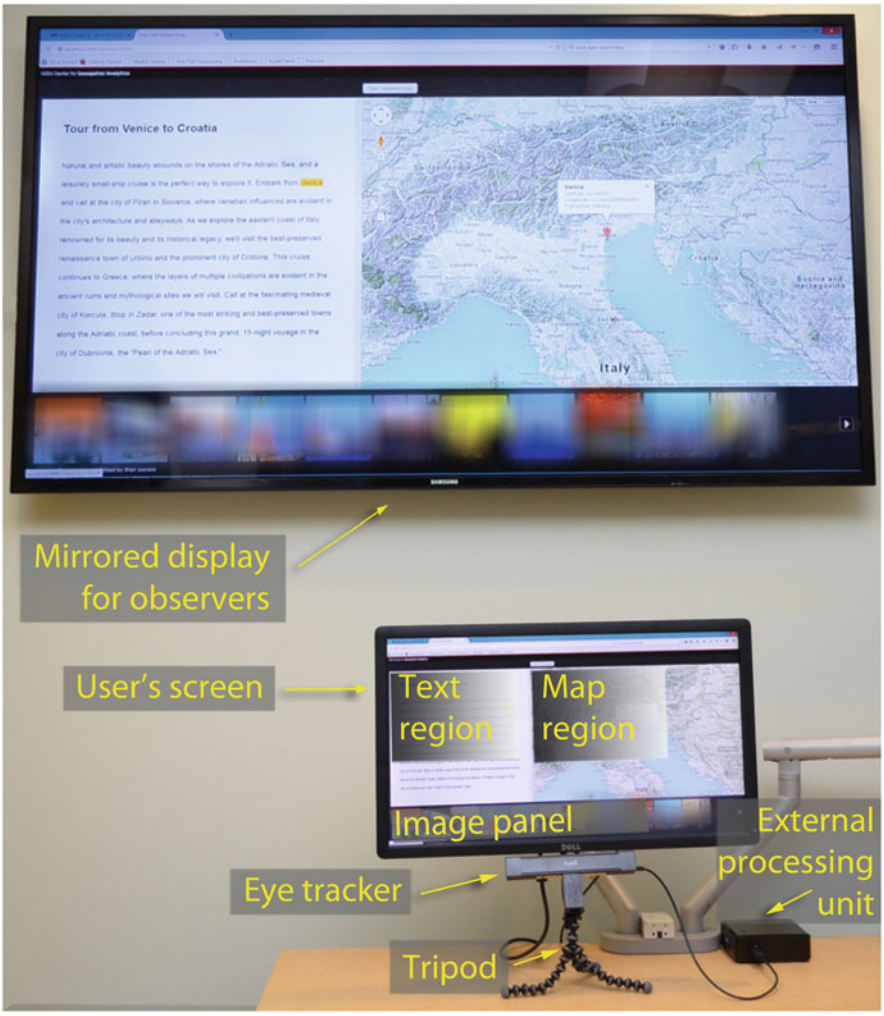


Fig. 1 The physical setup of GazeGIS in the lab: A seated reader views the application on a desktop monitor with an eye-tracking device centered below the screen. The wall-mounted display mirrors the primary display, allowing others to observe the interactions (The geotagged images were blurred for this reproduction to avoid copyright infringements)

3.2 Software

We combined several open source software libraries and proprietary APIs to create GazeGIS. For geoparsing, we selected the open-source package, CLAVIN-REST, for its modifiability and ease of use with Web mapping packages and other services (CLAVIN stands for Cartographic Location And Vicinity INdexer).

CLAVIN-REST extracts place names from unstructured text using an entity extractor and gazeteer (by default the GPL licensed Stanford Named Entity Recognizer and the GeoNames[1] world gazeteer). CLAVIN returns place names with latitude/longitude coordinates along with related information, such as population, alternative names by different languages, and so forth. Fuzzy matching is used to find misspelled place names and context-based heuristics are used to disambiguate topographic homonyms. The heuristics rely on the population of a location, the geographic proximity to other place names in the text, the text-based distance to other place names within the text, and political boundaries [4].

To visualize the geographic information extracted from the text, GazeGIS uses a Web map and an online geotagged photo database. There are a number of Web mapping APIs, both proprietary and open source. GazeGIS is built using a Google API which has abundant documentation. For imagery, we compared Flickr[2] and Panoramio.[3] Though Flickr hosts a greater number of images than Panoramio, Flickr also contains many personal images that are not fitting for our purposes. Hence, GazeGIS subscribes to Google's Panoramio, a geolocation-oriented photo-sharing site hosting millions of geotagged images. To receive related image feeds for selected toponyms, we filter images with a bounding box centered on a toponym's coordinates.

To communicate between eye tracker and browser, GazeGIS uses an open source Text 2.0 Framework package [7]. This interface enables event notification based on the user's gaze position. The package provides several functions to obtain gaze data from the eye tracker. We invoke a function to report an event as soon as the reader encounters a place name (the reader is not required to dwell on the location name). A small trigger radius around the element is specified to avoid simultaneous gaze-over event triggering for place names that are proximate within the text layout.

3.3 Gaze Interaction

The GazeGIS display updates are driven by user gaze feedback. Our javascript/HTML Web page application first allows the user to select a text document to read. The system then geoparses the selected document to identify toponyms and creates an HTML version of the document contents, inserting HTML tags within the document to mark the toponyms. Then, the document is loaded into the Web page application and the preprocessing is complete. Next, the feedback loop commences. The eye-tracking system analyzes the infrared video image of the eye and computes the coordinates of the gaze-point (the screen position of the viewer's gaze) and sends this to our GazeGIS application. The application tracks the reader's gaze as it passes

[1] http://www.geonames.org

[2] https://www.flickr.com

[3] http://www.panoramio.com

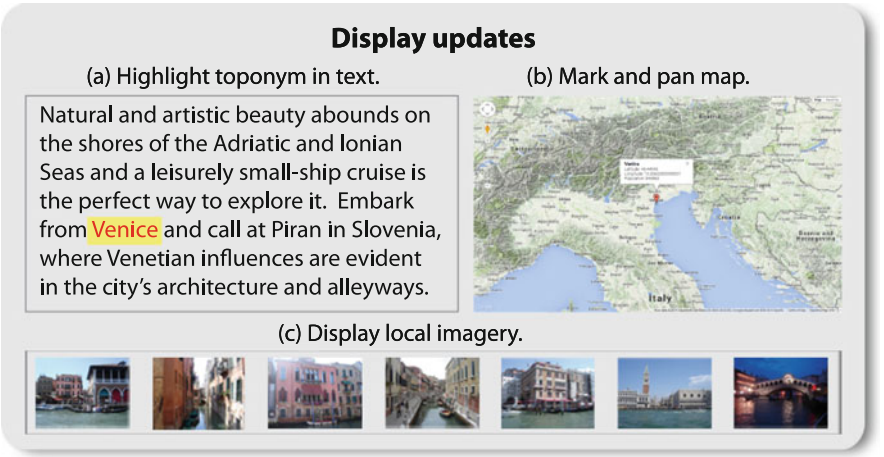


Fig. 2 When a viewer's gaze falls on a toponym, the GazeGIS display updates. A video demonstration is available at http://go.ncsu.edu/gazegis (Photos courtesy of Laura Tateosian)

over the text. When the user's gaze reaches a toponym identified by the geoparser, the display is updated (Fig. 2). The place is marked on the map, local imagery is displayed in an image panel in the browser, and the place name is highlighted within the text. This toponym remains highlighted until the reader returns to the text and then encounters another toponym within the text. This behavior is designed to enable users to inspect the map and images without losing their place in the reading.

4 User Study

We wondered how this idea would be received by users and how they would use the system. Here we explore some preliminary questions. Would readers enjoy the experience of reading with GazeGIS or would they find the map and images distracting? Would they spend time looking at the maps and images or just focus on the text? To answer these questions, we conducted a study in which we invited participants to try the system.

For our study, we selected two text passages for participants to read within our application. The first passage, S1, describes a historical event and the second one, S2, discusses travel. We selected topics to be accessible for a general audience and portray real-world usage scenarios. S1 and S2 are also suitable for the study due to their brevity and because they contain a few place names that might not be familiar to southeastern U.S. undergraduate students. S1 is a six sentence, 144 word, excerpt from a Wikipedia page, describing Paul Revere's midnight ride to warn of the approach of the British Army. This refers to six place names in the greater Boston area (Boston, Charles River, Lexington, Somerville, Medford,

and Middlesex County). S2 is a seven sentence, 145 word, passage from a travel brochure, touting a Mediterranean cruise docking in Italian and Croatian ports. S2 refers to twelve distinct place names, two of which appear twice in the passage, yielding a total of 14 tagged toponyms (Venice, Croatia, Adriatic Sea, Piran, Slovenia, Italy, Urbino, Crotone, Greece, Korcula, Zadar, and Dubrovnik, with Venice and Adriatic Sea appearing twice).

Fifty-four students ranging in age from 18 to 26 and one professor (over 45) participated in the study. Twenty-eight of our participants were male and 27 were female. Only three of the participants had previously used an eye tracker, five had GIS experience, and eight had traveled to the Mediterranean.

Participants were given a brief introduction to our application. Next, they were seated in a non-adjustable, non-swivel, stationary chair and a nine-point eye tracker calibration was completed. Then, instructions about how to proceed were displayed on the monitor. They were asked to relax, read at their own pace, and look at whatever interested them. They were not given a time limit. Passage S1 was displayed in our GazeGIS application in the text panel. When a user decided to proceed, another set of instructions was displayed and finally, passage S2 was displayed. This procedure (story S1, followed by story S2) was used for each participant. Once participants completed these steps, they were given a post-session survey to rate their experience and their perception of the application.

4.1 User Preference Evaluation

The post-session survey asked questions to determine whether displaying spatial information related to the text was perceived as helpful. In particular, did the participants like the content and display format and was the dynamic update based on what was being read positive or distracting. The results of these Likert-scale survey questions (1 = strongly disagree and 5 = strongly agree) for the 55 participants are shown in Table 1. Participants liked the map display and image panel

Table 1 Mean survey results, μ, with standard deviation, σ. Responses range from 1 (Strongly Disagree) to 5 (Strongly Agree)

Question	μ	σ
1. Integrating the eye tracker enhanced the reading experience	3.89	0.71
2. The dynamic display of geographic information was helpful	4.09	0.69
3. The geographic detail was displayed at the appropriate time	3.73	0.92
4. The content of geographic detail was appropriate	4.20	0.44
5. The display of geographic information was distracting	2.35	0.86
6. I liked the image panel	4.15	0.70
7. I liked the map display	4.07	0.76
8. I would have liked a way to disable/enable the dynamic display of geographic details	3.13	0.87
9. If this was an app, I would use it	3.93	0.83

and thought the content was appropriate. In addition, the dynamic screen update with geographic information relevant to the text being read was thought to be helpful and not distracting. Participants expressed some interest in being able to control the reading mode by toggling the dynamic behavior. This may indicate a need to refine update behavior. For example, less detail information could be supplied for large geographic regions that are proximate to the reader's location. Overall, the application was positively received by the participants.

4.2 Gaze Patterns

To investigate how participants interacted with the system, we used the Tobii Studio screen capture function to record participants' eye movements during the study. We expected that the readers would view the map and the images intermittently as they read. The video recording provides a retelling of the user experience, but we also needed a way to summarize the gaze pattern over time. Though gaze monitoring is often visualized with heat maps or gaze paths, these techniques would not work well for observing the patterns of visiting regions on our page. Heat maps suppress the temporal component and gaze paths overlaid on the application result in overplotting and occlusion due to the volume of data and the cyclic paths. Instead, we used a time plot similar to that presented by Räihä et al. [31].

We identified fixations in three areas of interest, the three main display regions, text, map, and image panel. Using an AOI time plot, we visualized the sequence of visits to these regions. Figure 3 shows an example of a gaze pattern that we found to be typical. This graph plots the fixation region visits over time in seconds with time increasing along the vertical axis. Each fixation is binned into one of four classes based on the region of the web page in which the fixation occurred, the text region, the map region, the image region, or a point that falls outside all of these. The fixation's region varies along the horizontal axis. The eight orange horizontal lines mark display updates that occur when a place name is gazed over or when the page is loaded or closed.

In Fig. 3, following the chart from time zero upward, the web page took one second to load. During the next 7.4 s, the participant, P12, read the few words of the story: "Paul Revere was told to use lanterns, 'one if by land, two if by sea', to alert the Boston colonists". When P12's gaze reached the word "Boston" (at $t = 8.4$ s), he inspected the map, looked back at the text ($t = 11.1$ s through $t = 12$ s), and then glanced at the images ($t = 12.5$ s through $t = 13.9$ s). Next, P12 went back to the text and started reading again until reaching the place name, "Charles River" which triggered a display update. This was followed by a look at the map and the images, and then back to the text, and so forth.

In general, the fixations started in the text region and periodically visited the map and/or the image regions, before returning to the text. As expected, periodic visits to the map and text usually corresponded to display updates in these regions. We wondered if these could be involuntary eye movements drawn by the moving map.

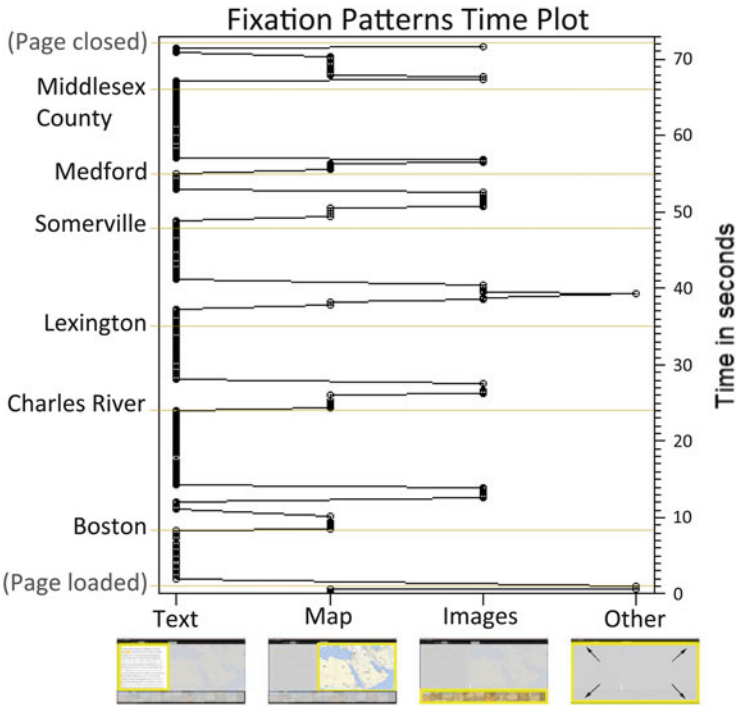


Fig. 3 AOI time plot of a typical fixation pattern visiting four regions (text, map, image panel, or elsewhere). The *eight orange horizontal lines* mark display updates that occur when a place name is gazed over or when the page is loaded or closed. Fixations recorded for one reader (participant 12) while reading story S1

Though a jump from the text to the updated map occurred frequently, it did not occur every time. For example, when the display updated for Lexington, P12 did not look immediately at the map. Also, the apparent lingering fixations in the map and images regions imply interest in the content, rather than a reflexive reaction.

4.3 Discussion

Further analysis is needed to understand the motivation of the patterns we can observe with the time plots. Çöltekin et al. [12] describe techniques that could be extended for sequence analysis of gaze-contingent displays. Analyzing the sequence and duration of gaze visits to the text, map, and image panel regions of the GazeGIS prototype may reveal additional information about usage patterns. Nevertheless, this preliminary analysis indicates that attention switching needs to be considered as the design evolves.

Some questions will require going beyond binning fixations in regions. For example, which map features do readers inspect? Does this vary depending on the type of narrative they are reading? Since the map is dynamic, approaches like those described by Ooms et al. [29] could be used to support this analysis.

Manual inspection of the videos revealed some participants reading with a similar pattern to P12, but then, once they finished the story, going back to gaze at each place name again in the text to see the map and images update, seemingly playing with the system. This is the kind of excitement and engagement the system can potentially stimulate. In general, the responses to the positively framed questions in the user post-session questionnaire were positive. In considering survey results, one concern is a phenomenon known as acquiescence bias, meaning uncertain respondents are more likely to agree. Though we only asked two negatively framed questions, these questions (5 and 8) did receive lower scores. If a strong acquiescence bias were at play, these questions should receive similarly high scores as the rest. This adds support for an interpretation of the survey as a positive reception for the application.

5 Intelligence Report Application

Our user study applied GazeGIS to educational and leisure-related documents and user feedback was positive, lending support to the potential for GazeGIS to be adopted for day-to-day text consumption. However, these casual uses are only a part of our motivation for developing GazeGIS. We are also interested in how our system can support critical sense-making tasks. For example, intelligence community analysts study events in which only part of the story is known. As a crucial component for interpreting events, analysts compose a narrative report to share with fellow analysts. To explore this capability, we used a report on the Malaysian Airlines Flight (MH-17) that exploded over Ukrainian airspace in July 2014.

The MH-17 simulated intelligence report was composed by an analyst to demonstrate the characteristics of an authentic report. The report indicates that the plane was shot down, but it was not definitively known by whom or with what intention. Additional information has since been uncovered, but this reflects knowledge at the time of writing. The report provides background on the region's political climate in the months leading up to the event. The narrative encodes key geographic content in both toponyms and latitude/longitude coordinates. Location is also given in relative terms. For example, contact was lost with the MH-17 plane 50 km from the Russian-Ukraine border. Places-of-interest (e.g., airports) and aircraft direction and speed are also reported.

There was tension in the Ukraine regarding its ties with Europe and Russia. As a result, pro-Russian rebels were beginning to seize airports and government buildings in eastern Ukraine. Crimean citizens voted to secede from the Ukraine and join Russia. Two Ukrainian military airplanes were destroyed, possibly by rebels, in June and July prior to the MH-17 incident. After the second plane went down, a no-fly

Table 2 Elements of interest in the simulated report

Report component	Example
Geographic locations	"...pro-Russian insurgents controlled Luhansk..."
Geographic coordinates	"...aircraft debris... at 480818N 0383818E..."
Relative location	"...50 km from the Russian-Ukraine border..."
Place-of-interest	"...destroyed as it came in to land at Luhansk Airport..."
Movement	...at 51.2265 24.8316 ... 107 degree Eastward course at 562 MPH..."
Date/time	...14 July 2014 at 1548Z..."
Figures	Photo of Luhansk
People	"...Ukrainian President Viktor Yanukovych..."
Reference URLs	BBC News Europe, 14 July 2014

zone was established. When shot down, flight MH-17 may have been flying above the no-fly zone at 10.05 km. The restricted area had been capped at 9.75 km. The report discusses reactions to the crash and finally presents a conclusion as to who was responsible for the crash and whether they believed it was a commercial flight or a Ukrainian military flight.

We inspected the report for references to spatial elements and other elements of interest, some examples of which are listed in Table 2. These additional report components include dates, figures, people's names, and content sources. The dates for important events are given. For example, the second Ukrainian military plane was shot down July 14, 2014 and the MH-17 incident occurred July 17, only three days later. The report also contained seven figures (including a photo of a Russian insurgent controlled Ukrainian town and a screen shot of a Malaysian Airlines tweet reporting loss of contact with the flight). Several political characters are named in the report. For example, the background discussion mentions former and interim Ukrainian Prime Ministers. The ten sources listed in the report include news and social media outlets as well as aviation reports and databases.

The implications of this report are complex. A number of the spatial elements and other components have visual analogues that could aid analysts in rapidly reading or writing a report expressing such complexities. To discuss these ideas with expert users, we loaded the simulated report into GazeGIS and demonstrated it to the report author and twelve others from the intelligence community. These experts found GazeGIS appealing for its potential to streamline workflow. For example, an analyst under time pressure would not need to interrupt report writing or reading to look up the location of an unfamiliar city or to map a geographic location given in latitude/longitude coordinates. As apparent in the MH-17 case, this information can be key to drawing conclusions about events. Where the debris was found, where ground control lost contact, and the boundaries of the no-fly zone were specified in geographic coordinates. In these cases, mapping these points can not simply be bypassed as a casual reader might do. The expert users also reinforced the utility

of automatically displaying pertinent imagery, scenes of the places being discussed. Sourcing in-house image databases would also be advantageous for their domain application. They also said that mapping non-geographic location names, places-of-interest such as airport or museum names would be informative. Additionally, there was interest in adding more GIS features. They suggested displaying an overview map alongside the main map. They also suggested overlaying additional GIS layers to the map display. As an example, cloud cover and other weather conditions on the day of the MH-17 incident could influence opinions of whether the shooters intended to target a commercial airline or thought it was a Ukrainian military plane.

This idea of a novel means for interacting with narrative and GIS was well received by these experts. The detailed feedback we received implies that the intelligence analysts we consulted are readily able to envision using this system to their advantage. As a followup, we have begun to investigate extensions to our system that could incorporate these suggestions.

6 Extending GazeGIS

Analysis of the intelligence report, discussions with expert users, and observations of user interactions with the system provided direction for extending this application beyond the initial prototype. For intelligence analysis, the system's key strength is automatically delivering convenient access to additional information about the sites under consideration. In fact, the analysts would like to have even more information. To test extensibility of the application, we added functionality.

To provide supplementary information, such as the historical importance, the timezone, and the current weather conditions of extracted toponyms, we sourced the GeoNames database and added information to the information window for each place-name marker on the map. GeoNames Web services, which can be used to link related information to geographic locations, contains over ten million geographical names. Along with weather station observations, we linked to the Reverse Geocoding Web service to find the timezone and the Wikipedia Fulltext Search Web service to link to Wikipedia information about the place.

For additional GIS data layers, we experimented with WXTiles[4] overlays, which provides weather data (precipitation, pressure, wind speed, temperature, and wave height) and satellite imagery. Two weeks of forecasts by the National Oceanic and Atmospheric Administration prediction models and two days of historical weather data are available at three hour intervals.

To provide an overview map, we used the ESRI Leaflet API,[5] along with the GeoJSON World Countries Polygon dataset, a database of boundaries for countries world-wide. In the overview map, we can display a city's country membership.

[4] http://www.wxtiles.com

[5] http://esri.github.io/esri-leaflet

To extract geographic coordinates formatted as degree, minutes, and second from unstructured text we used regular expressions and converted the points to latitude and longitude coordinates for mapping.

The MH-17 report contained figures illustrating important points and the text references them, though due to the limitations of conventional document formatting, these figures may not appear on the same page as the reference. Gaze-contingent behavior provides a means to rectify this. For our extended application, we parsed figure tags in an HTML document to display each figure dynamically in an additional panel when the reader reaches a reference to the figure.

Figure 4 shows a modified GazeGIS with panels labeled A through E. The MH-17 report is displayed in panel A and the city of Luhansk is tagged in the map in panel B. Panel B is also displaying a WXTiles weather overlay which shows a precipitation front (magenta) dropping off sharply to the east of Luhansk. Panel C is displaying Panoramio scenes from the Luhansk area. Panel D is displaying one of the figures included in the report. In the bottom right corner, Panel E is displaying an overview map, with a polygon overlay of the Ukraine, the country in which Luhansk resides.

Named entity extraction can be used to automatically identify dates and people's names (e.g., the date and time of the plane explosion or the name of the Ukrainian president, as listed in Table 2). These can be tagged and inserted in the feedback loop to provide additional forms of gaze-contingent displays. Extracting proximity and movement information presents a more interesting problem. Recent work by Moncla

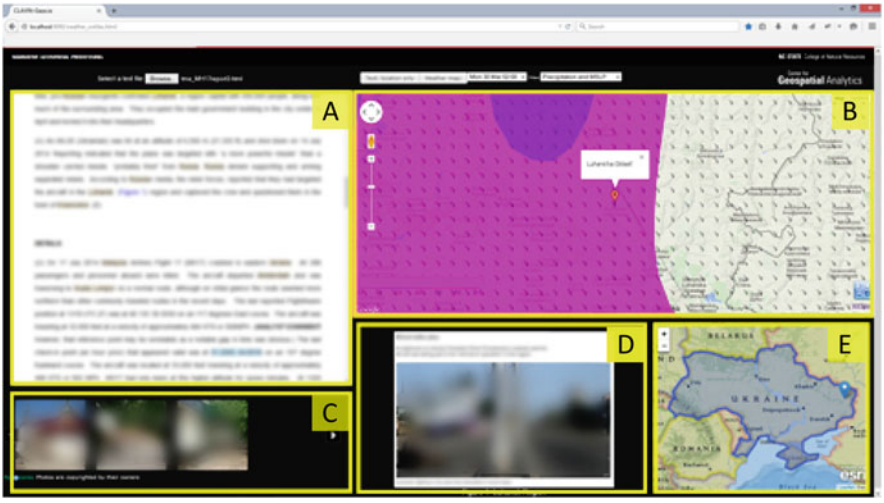


Fig. 4 Extended application: (**a**) text region, (**b**) map region, (**c**) image panel, (**d**) figure panel, and (**e**) overview map panel

et al. reconstructed itinerary maps from unstructured text descriptions of hikes [28]. Additional work would be required to extend this to broader applications so that routes described within the text could be dynamically derived and mapped. For places-of-interest (POIs) that are not geographic locations, online sources such as GeoNames and Wikipedia pages can provide resources for dynamically harvesting POIs and their geolocations.

Considering the potential number and variety of elements of interest within a text, presenting valuable information while minimizing cognitive load is challenging. Interaction techniques need to be honed to harness the affordances of tools, such as eye trackers, that are emerging to extend our suite of readily available computer interaction devices. Toward this end, we implemented an additional extension based on observations from our user study. Perhaps due to the gaze-contigent behavior of the text panel, some users appeared to expect gaze reactions from the maps. As a preliminary experiment with this, we developed a separate application which implemented gaze-contingent behavior for the map region instead of the text area. In this application, every toponym discovered by geoparsing the text is marked on the map at the outset. Then, when the user gazes on a marker, the marker label appears and the associated place name is highlighted everywhere it is mentioned in the accompanying text (see Fig. 5). We look forward to implementing additional map-gaze interaction to further enhance the user experience.

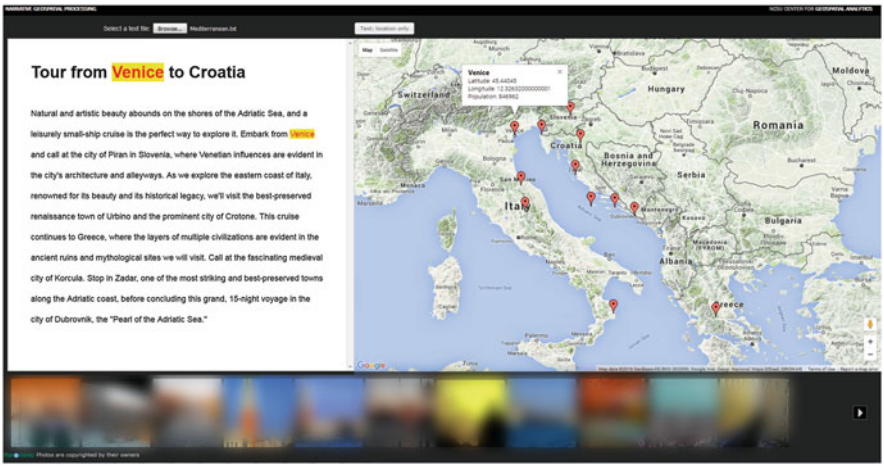


Fig. 5 This application enables map-based gaze browsing. All the place names that appear within the text are tagged on the map. When the user gazes at a tag on the map, every occurrence of the place name within the text is highlighted

7 Conclusions

We have presented a novel idea to change the way we read narratives by using gaze-contingent displays to augment the text with geographic information. We implemented the GazeGIS tool using geoparsing and eye-tracking technology to dynamically update map and image displays to provide a reader with easy access to pertinent geographic information. User experience feedback from laymen was strongly positive.

We exposed our system to expert analysts and studied a compelling use case scenario. These experts indicated interest in using the system for their work. Additional information and visual analytics can be added to the display to support specific domains. In this vein, some directions for future work are adapting the system for geographic literacy development and other educational purposes, for travel planning, or to facilitate efficiently summarizing the spatial elements of a document and generating maps for reports. Geographic context is fundamental for comprehending the nuances of the narratives we routinely encounter in our consumption of ebooks, news, and other electronic reading. With systems like GazeGIS, gaze-contingent behavior can provide pertinent geovisualizations in a timely and convenient fashion. Reading materials have shifted from analog to digital and eye-tracking capabilities are being built into laptops and hand-held devices. As the accuracy of eye tracking in consumer devices improves, applications such as this will be widely adopted.

Acknowledgements The authors wish to thank Tonya Adelsperger for her significant contributions to our research. This material is based upon work supported in whole or in part with funding from the Laboratory for Analytic Sciences (LAS). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the LAS and/or any agency or entity of the United States Government.

References

1. Abascal-Mena, R., Lopez-Ornelas, E., Zepeda, J.S.: Geographic information retrieval and visualization of online unstructured documents. Int. J. Comput. Inf. Syst. Ind. Manag. Appl. **5**, 089–097 (2012)
2. Abel, R.R., Kulhavy, R.W.: Maps, mode of text presentation, and childrens prose learning. Am. Educ. Res. J. **23**(2), 263–274 (1986)
3. Bektas, K., öltekin, A.C., Krüger, J., Duchowski, A.T.: A testbed combining visual perception models for geographic gaze contingent displays. In: Eurographics Conference on Visualization (EuroVis)-Short Papers (2015)
4. Berico Technologies: Open source software for geotagging unstructured big data – CLAVIN. All Things Open (2013)
5. Biedert, R., Buscher, G., Dengel, A.: The eyebook–using eye tracking to enhance the reading experience. Informatik-Spektrum **33**(3), 272–281 (2010)
6. Biedert, R., Buscher, G., Schwarz, S., Hees, J., Dengel, A.: Text 2.0. In: CHI'10 Extended Abstracts on Human Factors in Computing Systems, pp. 4003–4008. ACM (2010)

7. Biedert, R., Hees, J., Dengel, A., Buscher, G.: A robust realtime reading-skimming classifier. In: Proceedings of the Symposium on Eye Tracking Research and Applications, pp. 123–130. ACM (2012)
8. Bolt, R.A.: A gaze-responsive self-disclosing display. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 3–10. ACM (1990)
9. Borkin, M.A., Bylinskii, Z., Kim, N.W., Bainbridge, C.M., Yeh, C.S., Borkin, D., Pfister, H., Oliva, A.: Beyond memorability: visualization recognition and recall. IEEE Trans. Vis. Comput. Graph. **22**(1), 519–528 (2016)
10. Brychtova, A., Coltekin, A.: An empirical user study for measuring the influence of colour distance and font size in map reading using eye tracking. Cartogr. J. 53(3):202–212 (2016)
11. Burch, M., Konevtsova, N., Heinrich, J., Hoeferlin, M., Weiskopf, D.: Evaluation of traditional, orthogonal, and radial tree diagrams by an eye tracking study. IEEE Trans. Vis. Comput. Graph. **17**(12), 2440–2448 (2011)
12. Çöltekin, A., Fabrikant, S.I., Lacayo, M.: Exploring the efficiency of users' visual analytics strategies based on sequence analysis of eye movement recordings. Int. J. Geogr. Inf. Sci. **24**(10), 1559–1575 (2010)
13. Detweiler, D.M., Reisch, D.: Gettysburg: The Story of the Battle with Maps. Stackpole Books, Mechanicsburg (2013)
14. DIgnazio, C., Bhargava, R., Zuckerman, E., Beck, L.: Cliff-clavin: determining geographic focus for news. NewsKDD: Data Science for News Publishing, at KDD (2014)
15. Duchowski, A.: Eye Tracking Methodology: Theory and Practice, vol. 373. Springer Science & Business Media, London (2007)
16. I. Environmental Systems Research Institute: Story maps: everyone has a story to tell. http://storymaps.arcgis.com/en/. Accessed 24 July 2015
17. Esposito, V.J., Elting, J.R.: A military history and atlas of the napoleonic wars. Greenhill Books, New York (1999)
18. Gelernter, J., Balaji, S.: An algorithm for local geoparsing of microtext. GeoInformatica **17**(4), 635–667 (2013)
19. Hyrskykari, A., Majaranta, P., Räihä, K.-J.: From gaze control to attentive interfaces. In: Proceedings of HCII, vol. 2 (2005)
20. Isaksen, L., Barker, E., Kansa, E.C., Byrne, K.: Gap: a neogeo approach to classical resources. Leonardo **45**(1), 82–83 (2012)
21. Jacob, R.J.: What you look at is what you get: eye movement-based interaction techniques. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 11–18. ACM (1990)
22. Jacob, R.J., Karn, K.S.: Eye tracking in human-computer interaction and usability research: ready to deliver the promises. Mind **2**(3), 4 (2003)
23. Kern, D., Marshall, P., Schmidt, A.: Gazemarks: gaze-based visual placeholders to ease attention switching. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 2093–2102. ACM (2010)
24. Kulhavy, R.W., Stock, W.A., Verdi, M.P., Rittschof, K.A., Savenye, W.: Why maps improve memory for text: the influence of structural information on working memory operations. Eur. J. Cogn. Psychol. **5**(4), 375–392 (1993)
25. Leidner, J.L., Lieberman, M.D.: Detecting geographical references in the form of place names and associated spatial natural language. SIGSPATIAL Spec. **3**(2), 5–11 (2011)
26. Machado, J., Pedrosa, G., Freire, N., Martins, B., Manguinhas, H.: User interface for a geotemporal search service using DIGMAP components. In: Research and Advanced Technology for Digital Libraries, pp. 483–486. Springer, Berlin/Heidelberg (2009)
27. Maglio, P.P., Campbell, C.S.: Attentive agents. Commun. ACM **46**(3), 47–51 (2003)
28. Moncla, L., Gaio, M., Mustiere, S.: Automatic itinerary reconstruction from texts. In: Geographic Information Science, pp. 253–267. Springer, Cham (2014)
29. Ooms, K., Coltekin, A., De Maeyer, P., Dupont, L., Fabrikant, S., Incoul, A., Kuhn, M., Slabbinck, H., Vansteenkiste, P., Van der Haegen, L.: Combining user logging with eye tracking for interactive and dynamic applications. Behav. Res. Methods **47**(4), 977–993 (2015)

30. Pouliquen, B., Kimler, M., Steinberger, R., Ignat, C., Oellinger, T., Blackler, K., Fuart, F., Zaghouani, W., Widiger, A., Forslund, A.-C., et al.: Geocoding multilingual texts: recognition, disambiguation and visualisation. arXiv preprint cs/0609065 (2006)
31. Räihä, K.-J., Aula, A., Majaranta, P., Rantala, H., Koivunen, K.: Static visualization of temporal eye-tracking data. In: Human-Computer Interaction-INTERACT 2005, pp. 946–949. Springer, Berlin/New York (2005)
32. Rayner, K.: Eye movements in reading and information processing: 20 years of research. Psychol. Bull. **124**(3), 372 (1998)
33. Rupp, C., Rayson, P., Baron, A., Donaldson, C., Gregory, I., Hardie, A., Murrieta-Flores, P.: Customising geoparsing and georeferencing for historical texts. In: 2013 IEEE International Conference on Big Data, pp. 59–62. IEEE (2013)
34. Schwartz, N.H., Kulhavy, R.W.: Map features and the recall of discourse. Contemp. Educ. Psychol. **6**(2), 151–158 (1981)
35. Sibert, J.L., Gokturk, M., Lavine, R.A.: The reading assistant: eye gaze triggered auditory prompting for reading remediation. In: Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology, pp. 101–107. ACM (2000)
36. Tanriverdi, V., Jacob, R.J.: Interacting with eye movements in virtual environments. In: Proceedings of the SIGCHI conference on Human Factors in Computing Systems, pp. 265–272. ACM (2000)

Part II
Data and Metrics

Unsupervised Clustering of EOG as a Viable Substitute for Optical Eye Tracking

Nina Flad, Tatiana Fomina, Heinrich H. Buelthoff, and Lewis L. Chuang

Abstract Eye-movements are typically measured with video cameras and image recognition algorithms. Unfortunately, these systems are susceptible to changes in illumination during measurements. Electrooculography (EOG) is another approach for measuring eye-movements that does not suffer from the same weakness. Here, we introduce and compare two methods that allow us to extract the dwells of our participants from EOG signals under presentation conditions that are too difficult for optical eye tracking. The first method is unsupervised and utilizes density-based clustering. The second method combines the optical eye-tracker's methods to determine fixations and saccades with unsupervised clustering. Our results show that EOG can serve as a sufficiently precise and robust substitute for optical eye tracking, especially in studies with changing lighting conditions. Moreover, EOG can be recorded alongside electroencephalography (EEG) without additional effort.

1 Introduction

Optical (or video-based) eye-trackers are widely used to address diverse research questions ranging from basic eye-movement coordination [24] to natural scene understanding [16] to the complexities of British tea-making [21]. Without disrupting the behavior that is observed, it records when and how participants direct overt

N. Flad
Max Planck Institute for Biological Cybernetics, Spemannstrasse 41, 72076, Tuebingen, Germany
IMPRS for Cognitive and Systems Neuroscience, Oesterbergstrasse 3, 72074, Tuebingen, Germany
e-mail: nina.flad@tuebingen.mpg.de

T. Fomina
Max Planck Institute for Intelligent Systems, Spemannstrasse 38, 72076, Tuebingen, Germany
IMPRS for Cognitive and Systems Neuroscience, Oesterbergstrasse 3, 72074, Tuebingen, Germany
e-mail: tatiana.fomina@tuebingen.mpg.de

H.H. Buelthoff • L.L. Chuang (✉)
Max Planck Institute for Biological Cybernetics, Spemannstrasse 41, 72076, Tuebingen, Germany
e-mail: heinrich.buelthoff@tuebingen.mpg.de; lewis.chuang@tuebingen.mpg.de

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_9

attention to which objects in the visual scene. However, optical eye-trackers are limited in several ways, especially in their use in visualization research. First, optical eye-trackers can fail to track eye-markers (i.e., pupil or infra-red reflection) under changing lighting conditions. This means, complex visualizations with fast-varying illumination can often result in a large loss of data. Second, optical eye-trackers are difficult to combine with physiological measures, e.g., electromyography (EMG) or electroencephalography (EEG). Such measurements can be useful in revealing a participant's cognitive state when interacting with a visualization. For example, research in reading behavior has attempted to correspond fixated words with their evoked brain potential while allowing participants to move their eyes naturally [8, 20]. Combining optical eye-trackers with physiological measures is challenging because most optical solutions have comparatively low temporal resolutions (i.e., 60–500 Hz), relative to EEG/EMG devices (i.e., 256–20,000 Hz).

Besides optical eye-trackers, electrooculography (EOG) can be used to measure eye-movements. EOG can be used in conjunction with other measures, e.g., EEG, since the two measures can be recorded and connected to the same biosignal amplifier. Currently, EOG is used simultaneously in EEG recordings for the detection of eye-movements (e.g., [9, 28]). However, it is typically used to detect (and remove data from) instances when fixation on a single stimulus is violated, rather than for purposes of tracking eye-movements across multiple regions of interests (ROI). The reason for this is that EOG is considered to deliver low spatial resolution, in spite of its robustness.

Unlike optical eye tracking, EOG signals are minimally influenced by variations in lighting. This is because EOG records the potential of the electrical dipole that results from a positively charged cornea and a negatively charged retina [22]. An eye-movement corresponds to a change in the orientation of the dipole, which induces a measurable change in the potentials that are measured by the EOG electrodes. This shift in the electrical potential is proportional to the amplitude of the eye-movement. At the same time, the shift is insensitive to the lighting conditions, unlike the video-image of the eye's surface used in optical eye tracking. Therefore, EOG recordings could be a viable alternative to optical eye-trackers in situations with variable illumination, e.g., gaze-contingent visualizations, without a negative impact on measurement quality.

Automatic methods already exist that are able to detect eye-movement onsets and directions from EOG signals (e.g., [7, 14]). A k-Nearest Neighbors classification can discriminate saccades, fixations and vestibulo-ocular reflex movements (a reflexive eye-movement to stabilize the image on the retina during head-movements) with more than 80 % accuracy [29]. The ease and robustness of eye-movement detection on EOG-data has led to the development of robust applications in the areas of human-computer-interfaces (HCI) and neuroprosthetics. For example, Bulling et al. [4] developed a mobile EOG that was able to determine when a participant was reading, with up to 80 % accuracy in an everyday scenario [5]. Their device could also identify special eye-movements (so-called eye-gestures), that served as command inputs for an HCI with an average accuracy of 87 % [3]. In a different example, Gu et al. [15] proposed a method that allowed the EOG recordings of a

healthy eye to induce coupled movements in an artificial eye-implant of patients with a missing eye. This application was not only able to reliably identify an eye-movement's occurrence and direction, but also the eye's updated orientation after a movement.

EOG can estimate an eye's orientation with a precision of up to 2° [27],[1] given an appropriate calibration method (e.g., [13]). EOG calibrations serve to establish a mapping between the amplitude of EOG signal deflections and the size of the corresponding eye-movement (e.g., [19]). During calibration, participants perform a set of eye-movements with known amplitudes. This way, a level of precision is achieved that is sufficient in certain applications, e.g., when the objective is to discern dwells on elements of a visualization rather than to determine the fixation of exact pixels.

Typically, visualizations (e.g., graphical user interface) are defined in terms of their comprising regions-of-interest (ROIs), instead of their pixels. Hence, the EOG could be a viable alternative to optical eye tracking. ROIs can be defined beforehand by the experimenter based on his expectations and research question—for example, the frequency for which a table is referenced [17]. Alternatively, automated clustering methods can be used to identify ROIs in fixation data [26]. Such approaches identify ROIs by determining areas that are fixated repeatedly and/or over a long period of time. This is arguably more objective than defining ROI based on a prior hypothesis. However, it holds the implicit assumption that participants fixate ROIs accurately and its efficacy can vary depending on the available spatial resolution of the eye-tracking data.

This paper demonstrates how EOG can be a more reliable method than optical eye tracking, under display conditions that involve variable flicker. In addition, we present an alternative method to extract gaze information from EOG without the use of a dedicated calibration procedure. The remainder of this paper is structured as follows. Section 2 provides an overview of our dataset, as well as an explanation of the EOG-based methods and how we processed the data. In Sect. 3, we present the performance of the two possible EOG-based methods for ROI dwell extraction. Performance was evaluated in terms of the similarity of the extracted dwell sequence to the dwell sequence that our participants were explicitly instructed to perform. The similarities of dwell sequences between the two EOG-based methods and optical eye tracking were also computed to allow us to identify possible causes for ROI dwell misclassification. Chapter 4 summarizes our findings and discusses the benefits and disadvantages of these two EOG-based methods, as well as provide recommendations for further improvements.

[1] It is worth noting that EOG's spatial resolution is considered to be poor relative to the claims of many modern optical eye-trackers (<0.5°) [17].

2 Methods

2.1 Dataset

2.1.1 Participants

Twenty-one right-handed university students (eight female, age range: 21–29) volunteered for this study and were remunerated 12 €/h for their participation. All participants self-reported normal vision and gave written informed signed consent prior to data-collection.

2.1.2 Stimuli and Apparatus

Visual stimuli: Experimental stimuli consisted of 3-letter strings, which had a presentation size of 9° wide and 2.5° high. These strings were random combinations of 'X's or 'O's. 'XOX' was designated as the target stimulus for all participants while all other strings were considered distractor stimuli. The display was divided into four non-overlapping regions-of-interests (ROIs) and stimuli were always presented within these ROIs (see Fig. 1). These ROIs were positioned 10° to either the left and right of the screen center and 7.5° to either above or below the screen center. The brightness of the stimuli was 25.88 cd/m^2 on a background of 0.00 cd/m^2, as measured by a Konica Minolta CS-1000 spectroradiometer.

All stimuli were presented via an Intergraph 21sd107 CRT screen (100 Hz), which extended a field-of-view of 44.6° × 33.9°, width × height, to an observer in a chin-rest placed 50 cm away. The ROIs flickered discretely, between the luminance levels of 0.00 and 9.44 cd/m^2, at different non-harmonic frequencies (i.e., 7.7, 9.1, 11.1, 14.3 Hz). These luminance changes had no influence on the visibility of the stimuli and represented a range of dynamic display conditions that could pose a problem for optical eye-trackers.

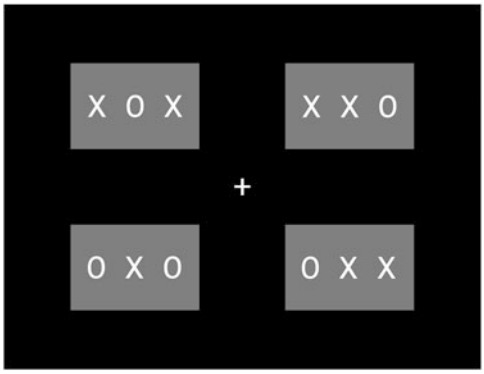


Fig. 1 The visual stimuli. The 3-letter strings are roughly 9° wide and 2.5° high and their centers are 20° apart horizontally and 15° vertically

Electrical recordings: We recorded our EOG data with a 64 channel bio-signal amplifier (BrainampDC; Brain Products, Gilching (Munich), Germany). The data had a sampling frequency of 240 Hz and was bandpass filtered from 0.1 to 1000 Hz. Four active electrodes (actiCAP, Brain Products, Gilching (Munich), Germany) were used for EOG recordings and placed above and below the left eye and at the outer corners of both eyes. The recordings were referenced to an electrode on the scalp (FCz, according to the international 10–20 system). To reduce noise, the participants were grounded over another scalp electrode (AFz). The impedance of all electrodes was kept below 20 kOhm. The other electrodes of the system were recording EEG signals and are not relevant to the current work, which focuses on EOG analyses.

Computer setup: Stimuli presentation and synchronization were controlled by a custom-written software using Matlab (R2011b 32-bit, The MathWorks, Inc.) and Psychtoolbox (V3.0.11). The EOG and EEG data was recorded by proprietary software (i.e., VisionRecorder, Brain Products, Gilching (Munich), Germany). Optical eye-tracker data was recorded using its proprietary software, which was remotely controlled by the Matlab application. For synchronization, the experimental control PC sent triggers to the eye-tracker software and the biosignal amplifier. All data analysis was performed offline using Matlab, EEGLAB and the EYE-EEG toolbox for EEGLAB.

2.1.3 Experimental Design and Procedure

The study consisted of one session of 24 trials, which were grouped into six blocks of four trials each. Each one min trial required 60 fixations (one fixation per second). There was a five min break between blocks and a one min break between trials.

In each trial, the four stimuli on the screen would update one after the other. The participants were instructed to respond with a button press to the target stimulus and to ignore distractor stimuli. In order to perform this task, participants had to move their eyes across the constantly updating ROIs. The stimuli updated in either a clockwise or counter-clockwise sequence. This ensured that the optimal scanning behavior for target detection would be a clockwise or counter-clockwise sequence of eye-movements.The sequence of updating ROI served as a benchmark for evaluating optical eye tracking and our EOG methods, under the assumption that the participants' scanning behavior matched the sequence of updating stimuli. The choice of stimulus that was presented each time was randomly determined with the exception that targets were never presented consecutively. A brief blank interval of 100 ms separated stimuli transitions. If a target appeared, it was always overwritten with a non-target after 1200 ms to discourage backwards saccades, which would constitute a behavioral violation of the benchmark sequence.

Every participant performed four training trials during which feedback was provided. Before every trial, participants were informed of the clockwise or

counter-clockwise order of stimuli presentation. After every trial, participants were asked to enter this presentation order again to ensure that they understood the task.

Finally, a 9-point calibration was performed for every block of trials to calibrate the optical eye-tracker.

2.2 Optical Eye Tracking

Optical eye-tracking data was collected with an iView X (iView X Hi-Speed Eye-tracker, 240 Hz, SensoMotoric Instruments GmbH (SMI), Teltow, Germany). To extract dwells, the optical eye-tracking data was processed as follows:

To begin, blinks, fixations and saccades were detected with a velocity-based method that was provided by SMI GmbH. Saccades were eye-movements that lasted for more than 24 ms with a peak velocity of at least 75°/s. In addition, the velocity profile of the eye-movement had to observe a saccadic profile—that is, the point of highest velocity had to be near the middle of the saccade (between 20 % and 80 % of eye-movement amplitude). Eye-blinks were identified by fast changes in the pupil diameter. Thus, fixations were simply defined as the intervals between saccades and/or blinks.

Following this, the fixations were analyzed with regard to the four ROIs. A fixation was assigned to a ROI, if its pixel-based location value fell inside a 5° margin around the ROI. Fixations outside these zones were marked as non-ROI fixations. Subsequent fixations in the same ROI were merged as a single dwell. This resulted in a sequence of ROI dwells for each trial that was submitted for further analyses.

2.3 EOG-Based Eye Tracking

Our EOG-based method differed from conventional optical eye tracking in one important design choice: We did not have to perform a dedicated calibration procedure for our EOG-based eye tracking to establish a mapping between changes in the EOG signal and changes in the eye's orientation. This means that we did not have to collect controlled fixations with known positions prior to data collection. Instead, our classifiers were trained offline in an unsupervised manner on the dataset to extract dwells. The extraction of dwell-sequences from EOG signals is discussed in the following sections.

2.3.1 Preprocessing of EOG Data

Both EOG-based eye tracking methods share the following pre-processing steps. First, the signals recorded by the four EOG electrodes were re-referenced to derive

two bipolar potentials: a channel for horizontal eye-movements between the outer corners of the eyes and a channel for vertical movements based on electrodes above and below the left eye.

Next, blinks were identified and removed from the data. Blinks are visually identifiable as short high peaks in the vertical EOG signal. For blink-detection, we implemented a spike detection method that relied on a moving window of 0.5 s length in 0.125 s step increments. Within this window, the maximum was first detected and then its adjacent minima. If the difference of both minima and the maximum was one standard deviation of the whole signal or more, this data segment was treated as a blink, 125 ms before and 250 ms after the maxima, and removed from further analyses. The length of the moving window as well as the length of the asymmetric removal was chosen to account for long blinks and the asymmetric profile of blinks [17].

2.3.2 Method 1: DB-Full

To cluster the pre-processed EOG data into meaningful ROIs, we applied DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [6, 12] on the full dataset (DB-Full). DBSCAN is a non-supervised machine learning method. In general, DBSCAN does not require any prior assumptions about the number or the shape of derived clusters. It defines clusters based on a pre-defined density threshold, which requires an assumption on the minimal density of the clusters. In the case of eye tracking, this density-threshold represents the minimum overall duration of fixations that are required in order for an ROI to be considered as being functionally meaningful. Thus, this threshold should be determined by domain knowledge and the experimental design.

We estimated the density threshold used for DB-Full based on the expected density of ROI dwells. For this purpose, we made the following assumptions:

- There are 4 ROIs
- All ROIs have approximately the same number of dwells
- 85 % of the data-points correspond to fixations
- There is an equal density of data-points for each ROI

This resulted in a chosen density threshold of eleven points per μV^2 or 35 points in a 1 μV radius.

For every data-point, DBSCAN first counts its number of neighbors and then compares it against the density threshold. If the threshold is reached, the data-point and its neighbors are treated as one cluster. The central data-point is referred to as the cluster's core. DBSCAN merges two clusters if they share at least one core point.[2] This means that DBSCAN separates clusters only if they are separated by an

[2]If a point is a neighbor to cores of different clusters, it is assigned to one of these clusters at random.

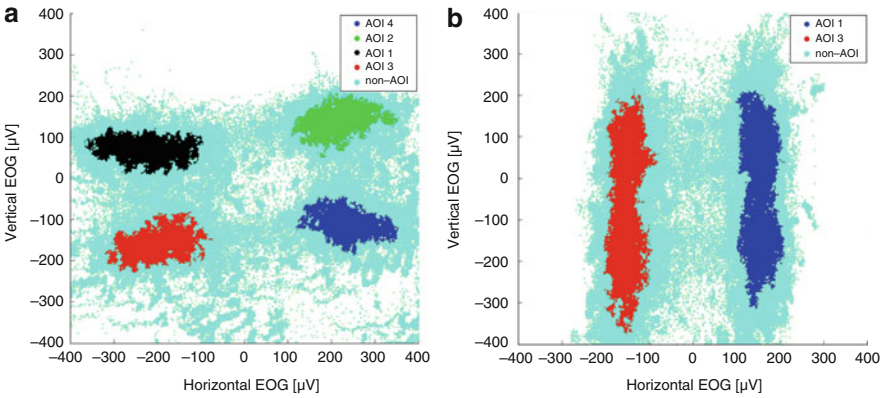


Fig. 2 Examples for density-based cluster. (**a**) Data of a participant where the clustering found four ROI. (**b**) Participant for which the clustering failed to find four clusters. The clusters could not be separated vertically since there is no area with a density lower than the threshold. The plots contain all data-points that are either fixations or saccades, but not blinks. Outliers, that are not part of any ROI, are indicated in *cyan*

area with a density that is below the threshold. In certain cases, however, this can be a weakness, as illustrated in Fig. 2 on the data from two participants. Data points that do not belong to any cluster are labeled outliers and indicated in cyan. The clusters are marked with different colors. In Fig. 2a's example, four clusters were identified. In Fig. 2b's example, only two clusters were found. Here, the data-points cluster into two ROIs instead of the expected four, namely fixations did not discriminate for the upper and lower ROIs. This pattern could have resulted if vertical eye-movements were occasionally accompanied by head movements, which resulted in smaller EOG deflections. If this was the reason, it is worth noting that an optical eye-tracking system that did not track head-movements would have experienced a similar difficulty in discerning ROI clusters from eye-movement data alone.

After labeling the whole dataset with DBSCAN, we removed the outliers that did not belong to any cluster. To obtain the sequence of ROI dwells for further analyses, we merged successive fixations within the same cluster.

2.3.3 Method 2: GMM-Fix

Alternatively, we extracted dwell-sequences from EOG signals by only submitting fixation data-points, instead of the full dataset, for unsupervised learning. Similar to optical eye tracking, fixation data-points were first identified and retained for further analyses, while the data-points of saccade and blinks were discarded. This second method (GMM-Fix) submits fixation data derived from the EOG signals to a Gaussian Mixture Model (GMM, see [2]).

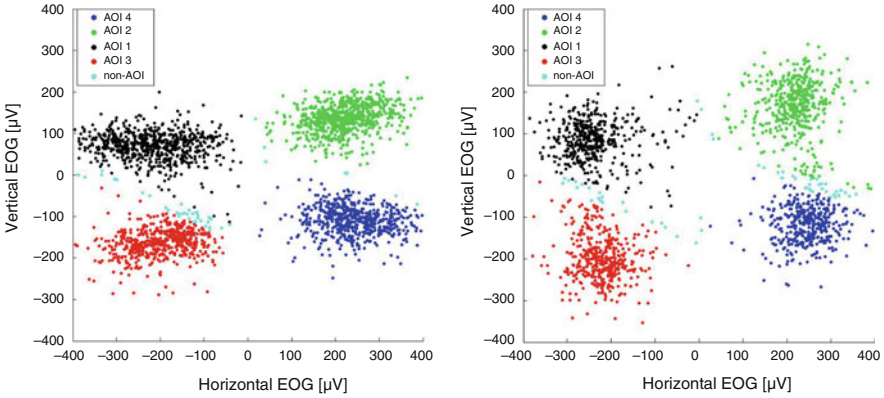


Fig. 3 Examples for classified fixations. Although these are plots from different participants, the four ROIs are easily recognizable and very similar. The plot only contains averaged fixations, which can belong to a ROI or are no-ROI outliers. The plot does not contain data from blinks or saccades

Saccades and fixations were identified from the pre-processed EOG signals by a velocity-based algorithm [10, 11] provided by the EYE-EEG plugin for EEGLAB. We considered an EOG data segment as a saccade, if it was at least 24 ms long with a velocity that was four times the standard deviation of the whole signal's velocity. Data segments between the identified saccades were labeled as fixations and subsequently classified by means of a GMM for ROI dwells.

We assumed that all fixation points were grouped in four clusters with a normal distribution and equal covariance. We decided to keep this method subject-independent. This means that we trained the GMM on the dataset of the first participant that we collected and classified all other participants' datasets with this model. We made this decision to ensure that this method was easy-to-use, under the assumptions that the EOG-deflections that correspond to a fixation are highly prototypical across individuals and robust against small deviations in the positions of the recording electrodes. Two classified datasets can be seen in Fig. 3. For the classification, we assigned every fixation based on the posterior probability to each cluster. However, fixations that did not reach a probability of at least 0.8 for any cluster were labeled 'non-ROI' dwell, similar to invalid fixations in the eye-tracker data. This resulted in a removal of 10 % of the data.

Finally, we combined successive fixations on the same ROI into one dwell.

2.4 Performance Measures

To evaluate the performance of the dwell detection and classification, we used the Needleman-Wunsch (NW) algorithm [23]. NW allows to compare the similarity between the benchmark sequences and the dwell sequences derived from either

optical eye tracking or the two EOG-based methods. In general, the algorithm determines the similarity of two sequences based on their global alignment. For this purpose, it requires a similarity matrix providing a similarity score for every possible pairing of two elements in the compared sequences.

The similarity matrix (Table 1) sets the score (between −1.0 and 1.0) for each individual pair of elements in the sequences that can be aligned by the NW algorithm. Thus, higher scores denote greater similarity between pairs of elements across the sequences. For example, similarity matrices can be employed to reflect the chemical similarity between pairs of elements across the nucleotide sequences and the likelihood that they were exchanged during evolution. To create a similarity matrix for the current dwell sequences, we decided that scores between ROIs would reflect their spatial proximity to each other. Identical ROIs would return a score of 1.0, while ROIs with close proximity to one another would have a score of −0.5 (e.g., top-left and top-right ROI), and ROIs that were furthest from each other would have a score of −1.0 (e.g., top-left and bottom-right). Pairing with non-ROI fixations were scored 0.5 since non-ROI fixations can be between the clusters and nearer to the matched ROI than a different ROI.

When pairing ROIs, gaps occurred when a given ROI from one sequence did not have a corresponding ROI in the compared sequence. A gap can occur when an ROI was missed. Pairings with gaps are scored with a gap penalty (i.e., −0.5) for the first gap and an affine gap penalty (i.e., −1.0) for every further, adjacent gap. This aims to prevent the algorithm from using more gaps than is absolutely necessary.

Eventually, the algorithm returns a normalized Needleman-Wunsch score (NW score) ranging from −1.0 to 1.0. A score of 1.0 means that the two sequences were exactly identical, which would reflect that a given dwell sequence reflected ideal scanning behavior from the participant, which was perfectly captured by either the optical eye-tracker or the EOG-based methods. We did not expect perfect scanning behavior. However, the recorded data was expected to reflect the same underlying scanning behavior, regardless of whether it was generated by optical eye-tracking or an EOG-based method. Therefore, the relative NW scores across the methods served to reflect the accuracy of each method.

Table 1 Similarity matrix for the Needleman-Wunsch algorithm. Each entry contains the score for the alignment between the ROI at the top of the column and left of the row. We applied a gap penalty of −0.5 and an affine gap penalty of −1.0

Matched pair of ROIs	Top-left	Top-right	Bottom-right	Bottom-left	No ROI
Top-left	1.0	−0.5	−1.0	−0.5	0.5
Top-right	−0.5	1.0	−0.5	−1.0	0.5
Bottom-right	−1.0	−0.5	1.0	−0.5	0.5
Bottom-left	−0.5	−1.0	−0.5	1.0	0.5
No ROI	0.5	0.5	0.5	0.5	1.0

3 Results

3.1 Data Loss

Twenty-three participants were initially recruited. However, two participants were removed due to general technical problems. Thus, the EOG data of all remaining 21 participants was submitted in their entirety to both proposed EOG-based methods (DB-Full and GMM-Fix) for dwell sequence extraction.

Given the challenging display conditions, namely variable flicker of stimuli presentation, optical eye-tracking data could only be obtained from twelve of the remaining 21 participants. The data of one participant was lost due to synchronization issues between the optical eye-tracker and the experimental PC. More importantly, the data of eight participants were unobtainable simply because it was not possible to successfully complete the calibration-validation procedure for effective optical eye-tracker under our display conditions. Typically, experimenters would not proceed any further and simply fail to report such unsuccessful attempts at eye tracking.

Next, we removed any trial with a total recorded fixation duration that was less than 25 % of the total trial duration. Such trials represented a failure to track eye-movements during testing. The data-loss due to this criterion was approximately 52 % of optical eye-tracking trials (range: 21–96 % trials across the twelve participants). No EOG recordings had to be removed.

3.2 Tracking Quality

We evaluated the performance of our two EOG-based methods as well as the optical eye-tracker with the NW algorithm described in Sect. 2.4. With it, we compared the dwell sequences against a benchmark sequence.

3.2.1 Performance Scores of Recorded Sequences

Random sequence: To establish a comparison baseline, we created a random sequence of ROI dwells. This random sequence provides the NW scores if our methods had assigned each fixation randomly and with equal probability to any given ROI. We created as many random sequences as we had for GMM-Fix, each with the same length as the benchmark sequence. The mean NW score of the random sequences was 0.35 and the highest score was 0.49. The histogram of the NW scores of random sequences is shown in Fig. 4a. The NW scores of real dwell sequences from optical eye tracking and EOG are also plotted in Fig. 4. A visual comparison reveals that the extracted sequences from recorded data are unlikely to be random, regardless of the method employed.

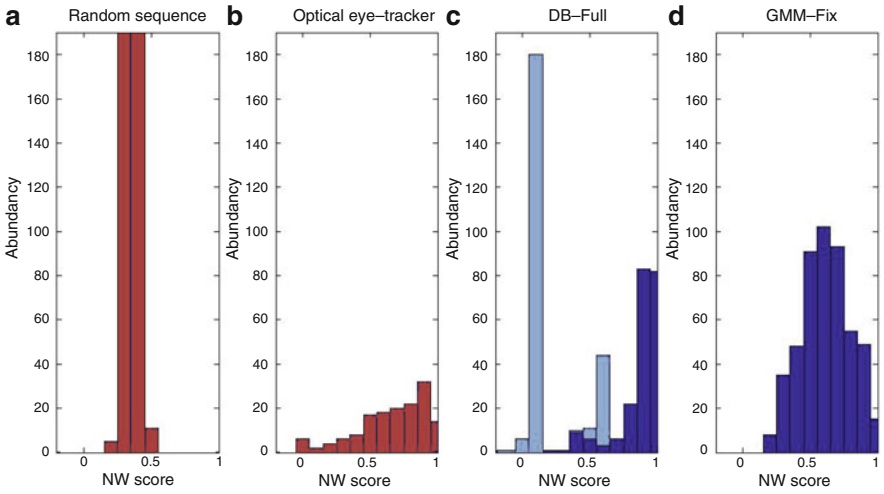


Fig. 4 Histogram of NW scores for all methods. In *red*, there are for comparison the NW scores of random 4-ROI sequences (**a**) and the optical eye-tracker (**b**). In *blue*, there are the sequences from our EOG-based methods. (**c**) shows the results DB-Full, with sequences with less than 4 ROI being paler. (**d**) contains the scores from GMM-Fix

Optical eye-tracker: The ROI dwell sequences recorded with the optical eye-tracker achieved a mean NW score of 0.68 (range: 0.00–0.98). The distribution of all scores is shown in Fig. 4b. The optical eye-tracker has a flat distribution over all values between 0 and 1. This means, there were trials where the optical eye-tracker worked well, but also trials where it performed poorly. Due to the substantial data-loss (i.e., 48 %), there were fewer NW scores for the optical eye-tracking histogram compared to the histograms for EOG-based NW scores.

EOG (DB-Full): The dwell-sequences that were extracted from EOG data using DB-Full resulted in a mean NW score of 0.52 (range: −0.18–1.00). The overall histogram (Fig. 4c) of NW scores shows a bimodal distribution of scores.

Surprisingly, DB-Full produced a sub-set of exceedingly low scores. This occurred because the unsupervised learning produced fewer than the expected four ROI in eleven participants. These datasets with fewer than four clusters (Fig. 4c: light blue) were unlikely to produce results with high experimental validity, since they always produce NW scores that are close to zero due to our chosen similarity matrix. When we only considered the EOG dwell sequences that produced four ROI (i.e., from ten participants; Fig. 4c: dark blue), we obtained a mean NW score of 0.87 (range: 0.17–1.00).

EOG (GMM-Fix): The dwell-sequences extracted by GMM-Fix produced a mean NW score of 0.62 (range: 0.18–0.99), which is comparable to the results of using the optical eye-tracker. The distribution of NW scores achieved by GMM-Fix is shown in Fig. 4d. The scores resemble a Gaussian distribution.

Summary: All extracted dwell-sequences produced results that were unlikely to be random, regardless of the method used. Optical eye-tracking produced the fewest number of NW scores given the difficulty of recording data under our display conditions. Against this standard, we find that GMM-Fix delivers comparable results to optical eye tracking without compromising on data-loss. DB-Full delivers better results than either methods, however this is only achievable when we remove the data of those participants that do not conform to the expected number of ROIs.

3.2.2 Comparison of Recorded Sequences

Low NW scores could have resulted for two reasons. It could have reflected inaccuracies in the dwell-sequence extraction procedure, or it could have been produced by non-ideal visual scanning behavior.

In this section, we analyze the similarities between dwell-sequences that were extracted using either optical eye tracking, DB-Full, or GMM-Fix. The level of similarity between dwell-sequences should reveal why similar NW scores were not obtained across the methods used. For example, if non-perfect NW scores were due to electro-magnetic noise in the testing environment, this would have influenced the EOG signals but not the optical eye-tracking data. This would be reflected in high NW scores between DB-Full and GMM-Fix. If our participants failed to exhibit the eye-movement pattern that was ideal for this visual scanning experiment, which was a basis for our similarity matrix, we would see higher NW scores between the three methods than their NW score against the benchmark sequence.

GMM-Fix and DB-Full: Comparing the sequences (with four ROI) that were extracted by either of our two EOG-based method resulted in a score of 0.67 (range: −0.25–1.00). Therefore, the scores from DB-Full decreased relative to when it was compared to the benchmark sequence, whilst the scores of GMM-Fix increased. This means that DB-Full is more similar to the benchmark sequence than it is to the sequence extracted from GMM-Fix, whilst the sequences from GMM-Fix is more similar to DB-Full sequences than it is to the benchmark sequence. In other words, DB-Full delivers more accurate visual scanning behavior than GMM-Fix given that it is less likely to misclassify dwells, in the instances where it is able to identify the appropriate number of ROIs in the first place.

Optical eye-tracker and DB-Full: Aligning the optical eye-tracking dwell sequences with DB-Full dwell-sequences (with four ROI) resulted in a mean NW score of 0.76 (range: 0.21–1.00). Therefore, the score achieved by the DB-Full decreases when we compare it to the optical eye-tracker, relative to when it was compared to the expected sequence. However, the NW score increases in the case of the optical eye-tracker. This means that DB-Full's sequences are more similar to the benchmark sequences than they are to the sequence extracted from optical eye tracking, whilst the dwell-sequences from optical eye tracking are more similar to DB-Full's sequences than it is to the expected sequence. Thus, both methods reflect

participant dwell scanning behavior, but DB-Full is a more accurate representation in cases where it is able to return the appropriate number of ROIs.

Optical eye-tracker and GMM-Fix: A comparison of the dwell sequences from the optical eye-tracker and GMM-Fix produced a mean NW score of 0.57 (range: 0.05–0.98). These are lower scores compared to those achieved by comparing the sequences of either GMM-Fix to the benchmark sequence or the optical eye-tracker to the benchmark sequence. In other words, these sequences are less alike to each other than to the benchmark sequence. This means it is unlikely that the non-ideal NW scores of both methods arise from a common cause, for example, inaccurate scanning behavior of participants. Instead, the two methods seem to suffer from different issues that are inherent to either optical eye tracking or EOG.

Summary: Our analyses show that it is unlikely that there is a common cause for the non-ideal NW scores across the three methods. This means, our EOG-based methods are not suffering from the same weaknesses as the optical eye-tracker, such as non-ideal lighting conditions. In instances were DB-Full was able to learn the existence of the four ROIs from EOG signals alone, it achieved the highest scores of all methods. GMM-Fix did not have any specific weakness that we could identify. All in all, GMM-Fix provided the most robust results across the three methods presented here.

4 Discussion and Outlook

In this paper, we introduced two EOG-based methods for extracting ROI dwells and compared their performance against the ROI dwells from an optical eye-tracker. Our results showed that EOG-based methods deliver a comparable performance to an optical eye-tracker in determining ROI dwells. In addition, they seem to be more robust against challenging lighting conditions (i.e., flickering display background). One advantage of EOG-based dwell detection for future applications lies in its ability to be easily integrated with electrophysiological recordings, e.g., EEG or EMG.

Optical eye tracking can be unreliable when used under variable lighting or display conditions. In contrast, illumination conditions and display dynamics have a negligible influence on EOG signals. Furthermore, EOG signals are linearly related to the eye's orientation across $\pm 70°$ and have a precision of up to 2° [27]. This spatial resolution is sufficient for many applications. In fact, we expect the availability of EOG methods to vastly increase the applicability of eye tracking to the evaluation of visualizations, especially those that are deployed in difficult lighting conditions, e.g., public displays [1] or those with dynamic content.

Our first EOG-based method (DB-Full) was entirely unsupervised and clustered the full dataset. The density-based clustering only required an assumption about the cluster densities that we expected to find, but not about size, position, shape or number of ROIs. While this leads to the advantage of being applicable even when

the ROI are not known beforehand, it can also lead to incorrect ROI extraction. DB-Full never created new ROI that we did not expect. However, its disadvantage lies in its disability to discriminate between ROIs. It may collapse over several ROI, which reduces its sensitivity. For this reason, DB-Full is not reliable if the exact number of ROI is not known. Its weakness is most likely due to the fact that we submitted the whole EOG signal to the analysis. This way, saccade data points between the ROI are taken into account, possibly preventing the ROI to be separated properly. Nevertheless, DB-Full showed the best performance of all methods in our study in the instances where it found the correct number of ROIs.

Our second EOG-based method (GMM-Fix) required pre-filtered EOG data and operates on extracted fixations. Furthermore, it assumes that the fixations are arranged in Gaussian clusters, representing the ROI. GMM-Fix had a performance comparable to the optical eye-tracker. However, it suffered from no data-loss. Of the methods in our study, GMM-Fix was the most reliable. Unlike DB-Full, GMM-Fix found four ROI for all participants. This superior robustness comes with the cost of a preceding processing step to detect fixations, which can potentially introduce problems due to incorrect detection. Nevertheless, the same processing step is common for optical eye-tracker, which means algorithms exist already to address this issue (e.g., [10, 25]).

The EOG-based methods presented in this work did not require user-based calibration to establish a mapping between EOG signal and eye orientation. On the one hand, this simplifies the application of our procedure in comparison to standard optical eye-tracking procedures. This saves experimentation time. On the other hand, introducing a user-calibration procedure (cf. [27]) could result in even higher accuracies than is currently observed. Subsequent work should look into the cost-benefit tradeoff of doing so. This should be weighted against the level of accuracy that researchers require, given the visualization conditions that they intend to investigate.

Our EOG-based methods should be considered in situations, in which optical eye tracking is likely to be unreliable or inconvenient. The possibility of combining our method with EEG could allow researchers to study simultaneous eye-movements and mental activity (e.g., [18]) without relying on a separate eye-tracker. This would potentially save time, money and data recording errors, that arise with complicated device synchronization setups. Another possible field of application for our methods are, for example, simulators with unpredictable changes in the visualization. Unlike optical eye-trackers, EOG signals are robust against the varying illumination conditions in these environments.

Acknowledgements This research was supported by the Max Planck Society. The authors N.F., H.H.B. and L.L.C. thank the German Research Foundation (DFG) for financial support within project C03 of SFB/Transregio 161.

References

1. Alt, F., Bulling, A., Gravanis, G., Buschek, D.: GravitySpot: guiding users in front of public displays using on-screen visual cues. In: Proceedings of the 28th ACM Symposium on User Interface Software and Technology, UIST 2015. ACM, New York (2015)
2. Bishop, C.M.: Pattern recognition and machine learning. Springer, New York (2006)
3. Bulling, A., Roggen, D., Tröster, G.: It's in your eyes: towards context-awareness and mobile HCI using wearable EOG goggles. In: Proceedings of the 10th International Conference on Ubiquitous Computing, UbiComp 2008. pp. 84–93. ACM, New York (2008). doi:10.1145/1409635.1409647
4. Bulling, A., Roggen, D., Tröster, G.: Wearable EOG goggles: seamless sensing and context-awareness in everyday environments. J. Ambient Int. Smart Environ. **1**(2), 157–171 (2009). doi:10.3233/AIS-2009-0020
5. Bulling, A., Ward, J.A., Gellersen, H., Tröster, G.: Robust recognition of reading activity in transit using wearable electrooculography. In: Indulska, J., Patterson, D.J., Rodden, T., Ott, M. (eds.) Proceedings of the 6th International Conference on Pervasive Computing, Pervasive 2008. pp. 19–37. Springer, Berlin/New York (2008). doi:10.1007/978-3-540-79576-6_2
6. Daszykowski, M., Walczak, B., Massart, D.L.: Looking for natural patterns in data: Part 1. Density-based approach. Chemom. Intell. Lab. Syst. **56**(2), 83–92 (2001). doi:10.1016/S0169-7439(01)00111-3
7. Degler, H.E., Smith, J.R., Black, F.O.: Automatic detection and resolution of synchronous rapid eye movements. Comput. Biomed. Res. **8**(4), 393–404 (1975). doi:10.1016/0010-4809(75)90015-4
8. Dimigen, O., Sommer, W., Hohlfeld, A., Jacobs, A.M., Kliegl, R.: Coregistration of eye movements and EEG in natural reading: analyses and review. J. Exp. Psychol.: Gen. **140**(4), 552 (2011)
9. Ding, J., Sperling, G., Srinivasan, R.: Attentional modulation of SSVEP power depends on the network tagged by the flicker frequency. Cereb. Cortex **16**(7), 1016–1029 (2006)
10. Engbert, R., Kliegl, R.: Microsaccades uncover the orientation of covert attention. Vis. Res. **43**(9), 1035–1045 (2003). doi:10.1016/S0042-6989(03)00084-1
11. Engbert, R., Mergenthaler, K.: Microsaccades are triggered by low retinal image slip. Proc. Natl. Acad. Sci. **103**(18), 7192–7197 (2006). doi:10.1073/pnas.0509557103
12. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD, vol. 96(34), pp. 226–231 (1996)
13. Finocchio, D.V., Preston, K.L., Fuchs, A.F.: Obtaining a quantitative measure of eye movements in human infants: a method of calibrating the electrooculogram. Vis. Res. **30**(8), 1119–1128 (1990). doi:10.1016/0042-6989(90)90169-L
14. Gopal, I.S., Haddad, G.G.: Automatic detection of eye movements in REM sleep using the electrooculogram. Am. J. Physiol. Regul. Integr. Comp. Physiol. **241**(3), R217–R221 (1981)
15. Gu, J., Meng, M., Cook, A., Faulkner, M.: A study of natural eye movement detection and ocular implant movement control using processed EOG signals. In: IEEE International Conference on Robotics and Automation, 2001. Proceedings 2001 ICRA, vol. 2, pp. 1555–1560 (2001). doi:10.1109/ROBOT.2001.932832
16. Henderson, J.M.: Human gaze control during real-world scene perception. Trends Cogn. Sci. **7**(11), 498–504 (2003). doi:10.1016/j.tics.2003.09.006
17. Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., Van de Weijer, J.: Eye tracking: a comprehensive guide to methods and measures. OUP, Oxford (2011)
18. Hutzler, F., Braun, M., Võ, M.L.H., Engl, V., Hofmann, M., Dambacher, M., Leder, H., Jacobs, A.M.: Welcome to the real world: validating fixation-related brain potentials for ecologically valid settings. Brain Res. **1172**, 124–129 (2007). doi:10.1016/j.brainres.2007.07.025
19. Kelly, S., Lalor, E., Reilly, R., Foxe, J.: Visual spatial attention tracking using high-density SSVEP data for independent brain-computer communication. IEEE Trans. Neural Syst. Rehabil. Eng. **13**(2), 172–178 (2005). doi:10.1109/TNSRE.2005.847369

20. Kliegl, R., Dambacher, M., Dimigen, O., Jacobs, A.M., Sommer, W.: Eye movements and brain electric potentials during reading. Psychol. Res. **76**(2), 145–158 (2011). doi:10.1007/s00426-011-0376-x
21. Land, M., Mennie, N., Rusted, J.: The roles of vision and eye movements in the control of activities of daily living. Perception **28**(11), 1311–1328 (1999). doi:10.1068/p2935
22. Mowrer, O., Ruch, T.C., Miller, N.: The corneo-retinal potential difference as the basis of the galvanometric method of recording eye movements. Am. J. Physiol.–Leg. Content **114**(2), 423–428 (1935)
23. Needleman, S.B., Wunsch, C.D.: A general method applicable to the search for similarities in the amino acid sequence of two proteins. J. Mol. Biol. **48**(3), 443–453 (1970). doi:10.1016/0022-2836(70)90057-4
24. Pelz, J., Hayhoe, M., Loeber, R.: The coordination of eye, head, and hand movements in a natural task. Exp. Brain Res. **139**(3), 266–277 (2001). doi:10.1007/s002210100745
25. Salvucci, D.D., Goldberg, J.H.: Identifying fixations and saccades in eye-tracking protocols. In: Proceedings of the 2000 Symposium on Eye Tracking Research & Applications, ETRA'00, pp. 71–78. ACM, New York (2000). doi:10.1145/355017.355028
26. Santella, A., DeCarlo, D.: Robust clustering of eye movement recordings for quantification of visual interest. In: Proceedings of the 2004 Symposium on Eye Tracking Research & Applications, ETRA'04, pp. 27–34. ACM, New York (2004). doi:10.1145/968363.968368
27. Stern, R.M., Ray, W.J., Quigley, K.S.: Psychophysiological recording. Oxford University Press, Oxford/New York (2001)
28. Thut, G., Nietzel, A., Brandt, S.A., Pascual-Leone, A.: α-band electroencephalographic activity over occipital cortex indexes visuospatial attention bias and predicts visual target detection. J. Neurosci. **26**(37), 9494–9502 (2006)
29. Vidal, M., Bulling, A., Gellersen, H.: Analysing EOG signal features for the discrimination of eye movements with wearable devices. In: Proceedings of the 1st International Workshop on Pervasive Eye Tracking and Mobile Eye-Based Interaction, PETMEI'11, pp. 15–20. ACM, New York (2011). doi:10.1145/2029956.2029962

Accuracy of Monocular Gaze Tracking on 3D Geometry

Xi Wang, David Lindlbauer, Christian Lessig, and Marc Alexa

Abstract Many applications such as data visualization or object recognition benefit from accurate knowledge of where a person is looking at. We present a system for accurately tracking gaze positions on a three dimensional object using a monocular head mounted eye tracker. We accomplish this by (1) using digital manufacturing to create stimuli whose geometry is know to high accuracy, (2) embedding fiducial markers into the manufactured objects to reliably estimate the rigid transformation of the object, and, (3) using a perspective model to relate pupil positions to 3D locations. This combination enables the efficient and accurate computation of gaze position on an object from measured pupil positions. We validate the of our system experimentally, achieving an angular resolution of 0.8° and a 1.5 % depth error using a simple calibration procedure with 11 points.

1 Introduction

Understanding the viewing behavior of humans when they look at objects plays an important role in applications such as data visualization, scene analysis, object recognition, and image generation [33]. The viewing behavior can be analyzed by measuring fixations using eye tracking. In the past, such experiments, especially for object exploration tasks, were performed with flat 2D stimuli presented on a screen [13]. However, since the human visual attention mechanism has been developed in 3D environments, depth may have an important effect on viewing behavior [21]. To understand the role of depth information, some studies [9, 16, 22] recently combined eye tracking with stereoscopic displays. However, these displays fail to provide natural depth cues; for example they suffer from stereoscopic decoupling, the mismatch of accommodation and vergence for the displayed depth [14]. Since our research objective is to investigate the viewing behavior of humans for stimuli that are genuinely three-dimensional, we need to be able to track 3D gaze positions with high accuracy.

X. Wang (✉) • D. Lindlbauer • C. Lessig • M. Alexa
TU Berlin, Berlin, Germany
e-mail: xi.wang@tu-berlin.de; david.lindlbauer@tu-berlin.de; christian.lessig@tu-berlin.de; marc.alexa@tu-berlin.de

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_10

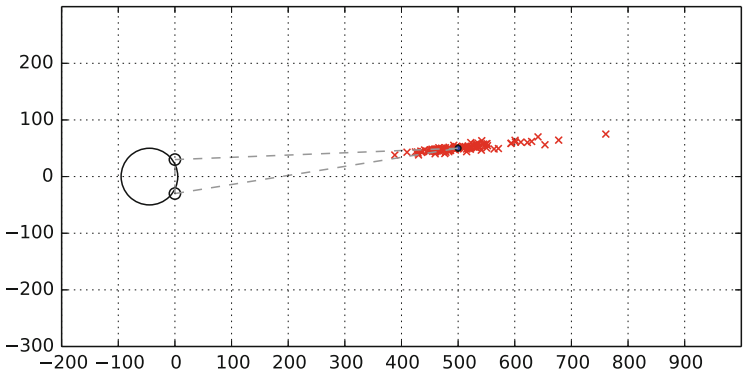


Fig. 1 Inherent error of vergence-based depth estimation for an object at a distance of 500 mm away from the eyes. The *red crosses mark* estimated 3D positions for normally distributed gaze directions with mean equal to the correct angle for the object (*black dot*) and a variance of 0.5°. The highly acute triangle that leads to the ill-conditioning of the depth calculation is shown as *dashed lines*. The worst case relative error is almost 50 %

Standard eye-tracking setups only determine human's viewing direction. The most common approach for determining viewing depth is to employ a binocular eye tracker and measure eye vergence, that is the orientation difference between the left and the right eye that ensures both are focused on the same point in space. However, as exemplified in Fig. 1, experimentally determining depth from binocular vergence is inherently ill-conditioned. Even for an object at a modest distance the eyes and the object form a highly acute triangle so that the inevitable inaccuracies in measuring pupil positions [13] lead to large errors in the estimated depth values. Although nonlinear mappings can be employed to reduce the error [1, 8, 12, 20, 23, 24, 26], these require complex calibration and expensive optimization of the mapping while still leading to relatively large inaccuracies.

We base our approach on a mapping between viewing directions gathered by an eye tracker and the physical world. This is done similar to EyeSee3D [27] by tracking fiducial markers in physical space with a camera mounted on the eye tracker. We extend their approach by not only acquiring establishing which object is looked at but also determining the exact 3D gaze position on the particular object. The main components to achieve such accurate tracking are:

1. 3D stimuli are generated by digital manufacturing so that their geometry is known to high accuracy and also available in digital form without imposing restrictions on the geometry that is represented.
2. fiducial markers are integrated into the 3D stimuli in order to reliably and accurately estimate the stimuli's 3D position relative to the head.
3. A simple calibration procedure that allows for an accurate computation of the perspective mapping from 3D positions to monocular pupil positions.
4. An error model for the mapping enables the computation of plausible positions on the 3D stimulus.

Our results demonstrate that for typical geometries we are able to obtain 0.8° angular resolution and reliable depth values within 1.5 % of the true value, including around silhouettes where the geometry has a large slope and depth estimation is hence particularly difficult. We accomplish this with only a monocular eye tracker and an 11-point calibration procedure.

In the next section, we discuss related work on 3D gaze tracking. Subsequently, in Sect. 3, we detail our setup and explain how 3D positions can be related to pupil coordinates. This is followed by a discussion of how 3D viewing positions can be obtained from pupil positions in Sect. 4. Experimental results verifying the accuracy of our approach are presented in Sect. 5. We conclude the paper in Sect. 6 with a discussion and possible directions for future work.

2 Related Work

The viewing behavior of humans is typically analyzed using eye tracking by measuring a subject's fixations. However, usually only flat 2D stimuli on a screen are employed, e.g., [5, 17, 25, 28], even when one is interested in 3D objects. Only recently the first studies considering the effect of depth were performed. Lang et al. [22] collected a large eye fixation database for still images with depth information presented on a stereoscopic display. Their results show that depth can have a significant influence on a subject's fixations. Jansen et al. [16] also employed a stereoscopic display to analyze the effect of depth, demonstrating that depth information leads to an overall increase in spatial distribution of gaze positions for visual exploration tasks. Both Lang et al. [22] and Jansen et al. [16] report that visual attention shifts over time from objects closer to the viewer to those farther away. Differences in fixations between 2D and 3D stimuli were recently also investigated for stereoscopic video [9, 10, 15, 29]. Discrepancies were mainly observed for scenes that lack on obvious (high-level) center of attention, with fixations having a larger spatial distribution when depth information is present.

Existing work investigating the role of depth information on fixation locations hence demonstrates that, at least under certain circumstances, depth has a significant effect on a subject's viewing behavior. However, so far only stereoscopic displays were employed, which do not provide all depth cues and suffer from stereoscopic decoupling [14]. Moreover, Duchowski et al. [7] showed that for stereoscopic displays the gaze depth of subjects does not fully correspond to the presented depth. Therefore, we believe that to understand viewing behavior for 3D objects, one should study stimuli that are genuinely three-dimensional. This provides the principal motivation for our work.

With 3D stimuli, also the depth values of fixation points have to be determined. The most common approach for obtaining fixation depth is to measure the vergence using a binocular eye tracker. However, computing depth values from binocular vergence is ill-conditioned since already for modest distances minuscule measurement errors in the pupil positions lead to large depth errors, cf. Fig. 1. To improve the

accuracy, Essig et al. [8] trained a neural network that maps from eye vergence to depth values. Maggia et al. [24] proposed a somewhat simpler but also nonlinear model for the mapping from measured disparity to depth. Building on these works, current techniques [1, 12, 20, 23, 26] that employ binocular vergence to determine fixation depth obtain an error that is within 10 % of the correct depth value.

Our work was inspired by existing approaches relating view directions to *known* geometry, e.g., in applications of virtual reality [6, 32]. Pfeiffer and Renner used fiducial markers to align the physical world to camera space [27]. By measuring eye vergence, they achieved an angular accuracy of 2.25°, which gives correctly classified fixation targets on the scale of whole objects. However, for investigating human viewing behavior on the surface of 3D objects, more accurate gaze tracking is required. Consequently, we create a setup with the goal of tracking visual attention on 3D objects.

3 From 3D Positions to Pupil Coordinates

In this section we describe our setup and how it enables to accurately determine gaze positions on an object. We use a monocular head mounted eye-tracking device with a front facing world camera capturing the environment and an eye facing camera capturing the pupil movement.

The world camera yields the position and orientation of fiducial markers, for example fixed to objects, relative to the subject's head relative to its reference frame. A projective mapping is then relates these 3D coordinates to pupil positions relative to the camera tracking the eye. This establishes a mapping between points in 3D space and pupil coordinates (this basic idea is illustrated in Fig. 2).

The mapping is calibrated by having a subject focus on markers at different locations, including varying depths. Once the mapping is established, 2D pupil positions can be turned into rays corresponding to gaze directions in 3D space. The gaze directions then determine the 3D positions on the object a subject is looking at, by intersecting the rays with the known 3D geometry.

In the following we will describe these steps in more detail.

3.1 From Local 3D Positions to World-Camera Coordinates

We employ fiducial markers to determine the 3D coordinates of locations in space in the world camera coordinate system. The mapping of a position $\mathbf{x} \in \mathbb{R}^3$, for example a point on a marker, to its projection $\mathbf{m} \in \mathbb{R}^2$ in the world camera image is given by

$$\begin{pmatrix} \mathbf{m} \\ 1 \end{pmatrix} = \mathbf{K}(\mathbf{R}\mathbf{x} + \mathbf{t}), \quad \mathbf{R}^{\mathsf{T}}\mathbf{R} = \mathbf{I} \tag{1}$$

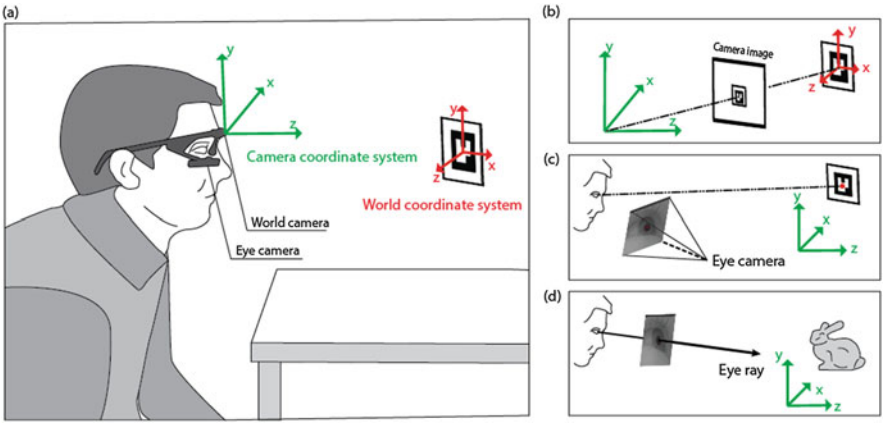


Fig. 2 The main idea of our approach is to establish a mapping between points in 3D space (i.e., world coordinate system) and pupil coordinates in the image coordinate system of the eye camera. We consider all 3D positions relative to the coordinate system of the world camera (i.e., camera coordinate system). (**b**) A point in world coordinate system is first transformed into the camera coordinate system. (**c**) We model the mapping between pupil position in the eye camera image and a location in world camera space as projection. (**d**) From the estimated projective transformation, we can estimate a corresponding eye ray for each pupil position

where $\mathbf{K} : \mathbb{R}^3 \to \mathbb{R}^2$ is the intrinsic world camera matrix, modelling the perspective mapping, and $\mathbf{R}$ and $\mathbf{t}$ are the rotation and translation of the camera, forming the rigid transformation. The mapping of $\mathbf{x}$ to its representation $\mathbf{w} \in \mathbb{R}^3$ in the world camera coordinate system is hence

$$\mathbf{w} = \mathbf{R}\mathbf{x} + \mathbf{t}. \tag{2}$$

We determine the intrinsic world camera matrix $\mathbf{K}$, which includes both radial and tangential distortion, in a preprocessing step using the approach of Heng et al. [11]. To determine the rigid transformation given by $\mathbf{R}$ and $\mathbf{t}$ we exploit that detected marker corner points $\mathbf{m}_i \in \mathbb{R}^2$ in the camera image have known 3D locations $\mathbf{x}_i \in \mathbb{R}^3$ in the marker's local coordinate system. Given at least three such points $\mathbf{m}_i$ in the camera image, we can determine $\mathbf{R}$ and $\mathbf{t}$ by minimizing the reprojection error.

Once $\mathbf{R}$ and $\mathbf{t}$ have been estimated, we can employ Eq. 2 to determine the position of the center of the marker in the world camera coordinate system, as required for calibration, or to map an object with a fixed relative position to a marker into the space, as is needed to determine gaze positions.

3.2 From World Camera Coordinates to Pupil Positions

Given positions $\mathbf{w} \in \mathbb{R}^3$ in the world camera coordinate system, obtained as described in the last section, we have to relate these to a person's gaze direction, described by pupil positions $\mathbf{p}$ in the eye camera image. We model the mapping as a projective transformation, because the cameras and the system of the eye (i.e., the head) are in fixed relative orientation and position. In homogeneous coordinates the transformation is given by

$$s \begin{pmatrix} \mathbf{p} \\ 1 \end{pmatrix} = \mathbf{Q} \begin{pmatrix} \mathbf{w} \\ 1 \end{pmatrix} \tag{3}$$

where $\mathbf{Q} \in \mathbb{R}^{3\times4}$ is a projection matrix that is unique up to scale. Given a set of correspondences $\{(\mathbf{w}_i, \mathbf{p}_i)\}$ between 3D points $\mathbf{w}_i$ in the world camera coordinate system and pupil positions $\mathbf{p}_i$ describing the gaze direction towards $\mathbf{w}_i$, we can determine $\mathbf{Q}$ by minimizing

$$E(\mathbf{Q}) = \sum_i \left\| s_i \begin{pmatrix} \mathbf{p}_i \\ 1 \end{pmatrix} - \mathbf{Q} \begin{pmatrix} \mathbf{w}_i \\ 1 \end{pmatrix} \right\|_2^2. \tag{4}$$

Fixing one coefficient of $\mathbf{Q}$ to eliminate the freedom on scale (we choose $\mathbf{Q}_{3,4} = 1$), this is a standard linear least squares problem. In practice, we solve this problem using correspondences $\{(\mathbf{w}_i, \mathbf{p}_i)\}$ obtained during calibration, as described in Sect. 5.

Since $\mathbf{Q}$ is a projective transformation we can factor it into an upper triangular intrinsic camera matrix $\mathbf{A}_Q$ and a rigid transformation matrix $\mathbf{T}_Q = (\mathbf{R}_Q, \mathbf{t}_Q)$. The factorization is given by

$$\mathbf{Q} = \mathbf{A}_Q \mathbf{T}_Q = (\mathbf{A}_Q \mathbf{R}_Q, \mathbf{A}_Q \mathbf{t}_Q) \tag{5}$$

and hence can be determined from the RQ decomposition of the left 3×3 block $\mathbf{A}_Q\mathbf{R}_Q$ of $\mathbf{Q}$. It can be computed using the QR decomposition as

$$\mathbf{J}(\mathbf{A}_Q\mathbf{R}_Q)^T\mathbf{J} = (\mathbf{J}\mathbf{A}_Q^T\mathbf{J})(\mathbf{J}\mathbf{R}_Q^T\mathbf{J}) \tag{6}$$

where $\mathbf{J}$ is the exchange matrix, which in our case is the column inversed version of the identity matrix.

3.3 From Pupil Positions to View Cones

So far we have related 3D locations to pupil positions. To determine a gaze point on an object we also have to relate pupil positions to a cone of positions in space. This also corresponds to the angular accuracy of our setup.

With the intrinsic eye camera matrix $\mathbf{A}_Q$, as determined in the last section, we can relate a homogeneous pupil position $\hat{\mathbf{p}} = (\mathbf{p}, 1)^T$ to an associated ray $\mathbf{r}$ in 3D world camera space:

$$\hat{\mathbf{p}} = \mathbf{A}_Q \mathbf{r}. \tag{7}$$

The depth along $\mathbf{r}$ is indeterminate since $\mathbf{A}_Q$ is a projection matrix. The angle between two rays $\mathbf{r}_i, \mathbf{r}_j$, represented by pupil coordinates $\mathbf{p}_i, \mathbf{p}_j$, is hence given by

$$\cos \eta_{ij} = \frac{\mathbf{r}_i^T \mathbf{r}_j}{\|\mathbf{r}_i\| \|\mathbf{r}_j\|} = \frac{\hat{\mathbf{p}}_i^\mathsf{T} \mathbf{A}_Q^{-\mathsf{T}} \mathbf{A}_Q^{-1} \hat{\mathbf{p}}_j}{\|\mathbf{A}_Q^{-1} \hat{\mathbf{p}}_i\| \|\mathbf{A}_Q^{-1} \hat{\mathbf{p}}_j\|}. \tag{8}$$

This suggests to interpret the matrix $\mathbf{A}_Q^{-\mathsf{T}} \mathbf{A}_Q^{-1}$ as an induced inner product $\mathbf{M}_Q = (\mathbf{A}_Q \mathbf{A}_Q^\mathsf{T})^{-1}$ on homogeneous pupil coordinates. The angle η_{ij} then becomes

$$\cos \eta_{ij} = \frac{\hat{\mathbf{p}}_i^\mathsf{T} \mathbf{M}_Q \hat{\mathbf{p}}_j}{(\hat{\mathbf{p}}_i^\mathsf{T} \mathbf{M}_Q \hat{\mathbf{p}}_i)^{1/2} (\hat{\mathbf{p}}_j^\mathsf{T} \mathbf{M}_Q \hat{\mathbf{p}}_j)^{1/2}}. \tag{9}$$

For multiple pairs $\mathbf{p}_i, \mathbf{p}_j$, Eq. 9 can be solved efficiently when the involved matrices are precomputed.

4 From Pupil Coordinates to Locations on an Object

Our objective is to determine a gaze position $\bar{\mathbf{w}} \in \mathbb{R}^3$ in space from a pupil position $\bar{\mathbf{p}}$ describing a gaze direction. Central to our approach for determining $\bar{\mathbf{w}}$ is that the geometry of the observed object is known to high accuracy. This is ensured by 3D printing the object $\mathscr{M}$ from its digital representation as a triangulated surface $\mathbf{M}$. The printed object also includes a fiducial marker, which allows us to determine the rigid transformation of the object in space as described in Sect. 3.1.

As explained before, in view of inaccuracies, the pupil position $\bar{\mathbf{p}}$ describes a cone in 3D space. Consequently, we wish to identify the vertices on the object that intersect the cone and are visible. We could then potentially identify the vertex closest to the center of the cone as the desired gaze location. The approach is illustrated in Fig. 3.

Let

$$\hat{\mathbf{p}}_i = \mathbf{Q}(\mathbf{R}\mathbf{v}_i + \mathbf{t}) \tag{10}$$

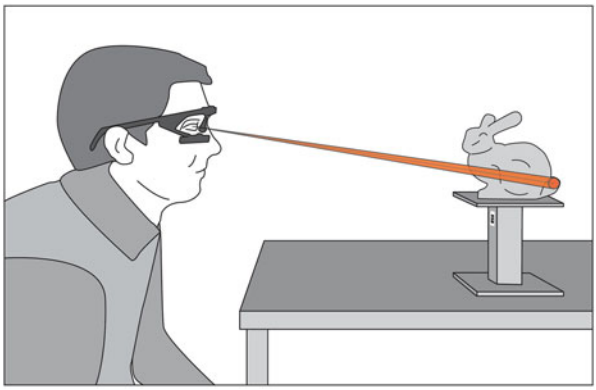


Fig. 3 Pupil positions provide by the eye tracker correspond to cones in 3D space. The fiducial marker on the 3D printed marker allows tracking the geometry in 3D space. Intersecting the cone against the geometry yields gaze points on the object

be the homogeneous pupil position $\mathbf{p}_i = (p_{i_1}, p_{i_2}, p_{i_3})^\mathsf{T}$ corresponding to vertex $\mathbf{v}_i$. Then we find the set of vertices

$$\Gamma_c(\bar{\mathbf{p}}) = \left\{ \mathbf{v}_i \in M \;\middle|\; \frac{\hat{\bar{\mathbf{p}}}^\mathsf{T} \mathbf{M}_Q \hat{\mathbf{p}}_i}{(\hat{\bar{\mathbf{p}}}^\mathsf{T} \mathbf{M}_Q \hat{\bar{\mathbf{p}}})^{1/2} (\hat{\mathbf{p}}_i^\mathsf{T} \mathbf{M}_Q \hat{\mathbf{p}}_i)^{1/2}} > \cos c \right\} ; \tag{11}$$

that is, we are determining which vertices $\mathbf{v}_i$ on the object lie within the cone of angular size c centered around the eye ray corresponding to $\bar{\mathbf{p}}$. From these vertices, we consider the one closest to the eye as the intersection point. This point can be determined efficiently solely using p_{i_3}. Note that since the metric $\mathbf{M}_Q$ has a natural relation to eye ray angle, we can choose c based on the accuracy of our measurements.

4.1 Spatial Partitioning Tree

For finely tessellated meshes, testing all vertices based on Eq. 11 above results in high computational costs. Spatial partitioning can be used to speed up the computation, by avoiding to test vertices that are far away from the cone. Through experimentation we have found sphere trees to outperform other common choices of spatial data structures (such as kd-trees, which appear as a natural choice) for the necessary intersection against cones.

Each fixation on the object is the intersection of the eye ray cone with the object surface, which is represented by a triangulated surface $\mathbf{M}$. Therefore, in the first step we perform an in-cone search to find all intersected vertices. This intersection result contains both front side and back side vertices. We are, however, only interested in visible vertices that are unoccluded with respect to the eye.

Popular space-partitioning structure for organizing 3D data are K-d trees, which divide space using splitting hyperplanes, and octrees, where each cell is divided into eight children of equal size. For our application, such axis-aligned space partitionings would require a cone-plane or cone-box intersection, which potentially incurs considerable computational costs. In order to avoid this, we build a space-partitioning data structure based on a sphere tree.

Sphere tree construction Our sphere tree is a binary tree whose construction proceeds top-down, recursively dividing the current sphere node into two child nodes. To determine the children of a node, we first apply principle component analysis and use the first principle vector, which corresponds to the largest eigen value of the covariance matrix, as the splitting direction. A partitioning hyperplane orthogonal to the splitting direction is then generated so that the elements in the node are subdivided into two sets of equal cardinality. Triangle faces intersecting with the splitting hyperplane are assigned to both sets. The child nodes are finally formed as the bounding spheres of the two sets and computed as proposed in [30].

We calculate the sphere-cone intersection following the method proposed in [31]. The problem is equivalent to checking whether the sphere center is inside an extended region, which is obtained by offsetting the cone boundary by the sphere radius. Note that the extended region differs from the extend cone, and its bottom is a sector of the sphere. For each intersected leaf node, we perform the following in-cone test to find the intersected vertices.

In-cone test A view cone is defined by an eye point $\mathbf{a}$ (i.e., the virtual eye position), a unit length view direction $\mathbf{r}$, and opening angle δ. The in-cone test allows us to determine if a given point $\mathbf{v}_i$ lies inside this cone. Given the matrix $\mathbf{M} \in \mathbb{R}^{4\times4}$

$$\mathbf{M} = \begin{pmatrix} \mathbf{S}, & -\mathbf{Sa} \\ -\mathbf{a}^T\mathbf{S}, & \mathbf{a}^T\mathbf{Sa} \end{pmatrix}, \tag{12}$$

where $\mathbf{S} = \mathbf{r}\mathbf{r}^T - \mathbf{d}^2\mathbf{I}$ with $\mathbf{d} = cos\delta$, the point $\mathbf{v}_i$ lies inside the cone only when $\hat{\mathbf{v}}^T\mathbf{M}\hat{\mathbf{v}} > 0$ where

$$\hat{\mathbf{v}} = \hat{\mathbf{v}}_i - \hat{\mathbf{a}} = \begin{pmatrix} \mathbf{v}_i \\ 1 \end{pmatrix} - \begin{pmatrix} \mathbf{a} \\ 1 \end{pmatrix}. \tag{13}$$

Visibility test The visibility of each intersected vertex is computed by intersecting the ray from eye point to the vertex with the triangle mesh. The vertex is visible if no other intersection is closer to the eye point. We use the Möller-Trumbore ray-triangle intersection algorithm [19] for triangles in intersected bounding spheres. In our implementation, the maximum tree depth is set to 11, which allows for fast traversal and real-time performance.

4.2 Implementation

Our software implementation uses OpenCV [4], which was in particular employed to solve for the rigid transformations **R**,**t** as described in Sect. 3.1. We determine **Q** using Eq. 4 with the Ceres Solver [2]. The optimization is sensitive to the initial estimate, which can result in the optimization converging to a local minimum, yielding unsatisfactory results. To overcome this problem, we use a RANSAC approach for the initial estimate, with the error being calculated following Eq. 14 and 1000 iterations. The result of this procedure serves as input for the later optimization using the Ceres solver.

5 Experiments

In the following, we will report on preliminary experimental results that validate the accuracy of our setup for tracking 3D gaze points and that demonstrate that a small number of correspondences suffices for calibration. These results were obtained using two exploratory experiments with a small number of subjects ($n = 6$).

Participants and apparatus We recruited 6 unpaid participants (all male), all of which were students or staff from a university. Their age ranged from 26 to 39 years and all had normal or corrected-to-normal vision, based on self-reports. Four of them had previous experience with eye tracking.

The physical setup of our experiment is shown in Fig. 4. For measuring fixations we employed the Pupil eye tracker [18] and the software pipeline described in the previous sections.

5.1 Accuracy of Calibration and Gaze Direction Estimation

In Sect. 3.2 we explained how the projective mapping **Q** from world camera coordinates to pupil positions can be determined by solving a linear least squares

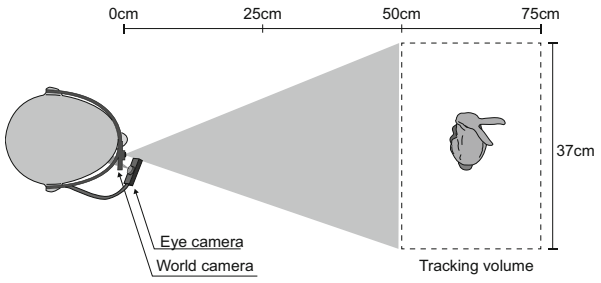


Fig. 4 Physical setup used in our experiments

Fig. 5 We accurately estimate 3D gaze positions by combining digital manufacturing, marker tracking and monocular eye tracking. With a simple procedure we attain an angular accuracy of 0.8°

problem. As input to the problem one requires correspondences $\{(\mathbf{w}_i, \mathbf{p}_i)\}$ between world camera coordinates $\mathbf{w}_i$ and pupil positions $\mathbf{p}_i$. The correspondences have to be determined experimentally, and hence will be noisy. The accuracy with which $\mathbf{Q}$ is determined therefore depends on the number of correspondences that is used. In our first experiment we investigated how many correspondences are needed to obtain a robust estimate for $\mathbf{Q}$. The same data also allows us to determine the angular error of our setup.

Procedure We obtained correspondences $\{(\mathbf{w}_i, \mathbf{p}_i)\}$ by asking a subject to focus on the center of a single fiducial marker (size 4×4 cm) while it is presented at various locations in the desired view volume (see Fig. 5, third image). We have augmented the center of the marker with a red dot to make this task as unambiguous as possible. At each position of the marker, we estimate a single correspondence $(\mathbf{w}_i, \mathbf{p}_i)$ based on the estimation of the rigid transformation for the marker, cf. Sect. 3.1. For each participant, we recorded 100 correspondences $\{(\mathbf{w}_i, \mathbf{p}_i)\}$ for two different conditions, resulting in a total of 200 measurements per participant. In the first condition the head was fixed on a chin rest while in the second condition participants were only asked to keep facing towards the marker. For both conditions the marker was moved in a volume of 0.37 m (width) $\times$ 0.4 m (height) $\times$ 0.25 m (depth) at a distance of 0.75 m from the subject (see Fig. 4).

Data processing For each dataset we perform 10 trials of 2-fold cross validation and estimate the projection matrix using 7 to 49 point pairs. In each trial, the 100 correspondences are randomly divide into 2 bins of 50 point pairs each. One bin is used as training set and the other as testing set. Point pair correspondences from the training set are used to compute the projection matrix $\mathbf{Q}$ which is then employed to compute the error between the gaze direction given by the pupil position $\mathbf{p}_i$ and the true direction given by the marker center $\mathbf{w}_i$ for the points in the test data set. From Eq. 9 this error can be calculated as

$$\eta_i = \cos^{-1} \frac{\mathbf{p}_i^\top \mathbf{M}_Q \mathbf{Q} \mathbf{w}_i}{(\mathbf{p}_i^\top \mathbf{M}_Q \mathbf{p}_i)^{1/2} (\mathbf{w}_i^\top \mathbf{Q}^\top \mathbf{M}_Q \mathbf{Q} \mathbf{w}_i)^{1/2}}. \tag{14}$$

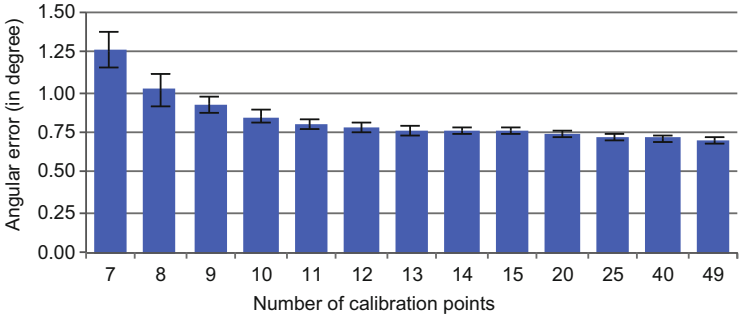


Fig. 6 Mean values and standard errors for angular error as a function of the number of calibration points ranging from 7 to 49. No significant changes in angular error occur when using 11 or more calibration points

Analysis and results In order to analyze the influence of the number of calibration points as well as the usage of the chin rest on the estimation accuracy, we performed a repeated measures ANOVA ($\alpha = 0.05$) on the independent variable *Chin rest* with 2 levels (with, without) and *Calibration* with 43 levels (the corresponding number of calibration points, i.e., 7 to 49). The dependent variable was the calculated angular error in degree. We used 10 rounds of cross validation for our repeated measures, with each data point being the average angular error per round. This resulted in an overall of 860 data points per participant (2 *Chin reset* $\times$ 43 *Calibration* $\times$ 10 cross validation).

Results showed a main effect for *Calibration* ($F_{42,210} = 19.296$, $p < 0.001$). The difference between 20 points ($M = 0.75$, $SE = 0.02$) and 42, 44, 45, 46, 47 and 48 points (all $M = 0.71$, $SE = 0.02$) was significantly different, as well as 22 points ($M = 0.74$, $SE = 0.02$) compared to 45 points (all $p < 0.05$). No other combinations were statistically significantly different, arguably due to high standard deviation for lower number of calibration points. Mean values and standard errors are depicted in Fig. 6.

When using 11 to 49 calibrations points, the angular error averages at around 0.73° ($SD = 0.02$), which is within the range of human visual accuracy and goes in line with the specifications of the pupil eye tracker for 2D gaze estimation [3, 18]. The results furthermore demonstrate that even for a relatively low number of calibration points, comparable to the 9 points typically used for calibration for 2D gaze estimation [13, 18], our method is sufficiently accurate.

No significant effect for *Chin rest* ($F_{1,5} = 0.408$, $p = 0.551$; with chin rest $M = 0.73$, $SE = 0.05$; without chin rest $M = 0.78$, $SE = 0.04$) was present, suggesting that the usage of a chin rest has negligible influence on the angular accuracy and our method is hence insensitive to minor head motion. This goes in line with the observation that light head motion has no effect on the relative orientation and position of eye, eye camera, and world camera. It should be noted, however, that participants, although not explicitly instructed, were mostly trying to keep their head

steady, most likely due to the general setup of the experiment. Giving participants the ability to move their head freely is an important feature for exploring objects in a natural, unconstrained manner. However, quantifying the effect of large scale motion on accuracy should be subject to further investigations.

5.2 *Accuracy of 3D Gaze Position*

In our second experiment we explored the accuracy of our approach when viewing 3D stimuli. As model we employed the Stanford bunny and marked a set of predefined target points on the 3D printed bunny as shown in Fig. 7, left. After a calibration with 11 correspondences, as described in the last section, the test subjects were asked to focus on the individual targets (between 1 and 2 s). A heat map of the obtained gaze positions is shown in Fig. 7, right. Fixations are calculated based on Eq. 11 where the angular size c is set to be 0.6°. Table 1 shows the angular error of each target in degrees as well as the depth error in mm.

Angular error depends mostly on the tracking setup. However, since the intersection computation with eye ray cones is restricted to points on the surface (vertices in our case), we get smaller angular errors on silhouettes.

Depth accuracy, on the other hand, depends on the slope of the geometry. In particular, at grazing angles, that is when the normal of the geometry is orthogonal or almost orthogonal to the viewing direction, it could become arbitrarily large. For the situations of interest to us where we have some control over the model, the normal is orthogonal or almost orthogonal to the viewing direction mainly only around the

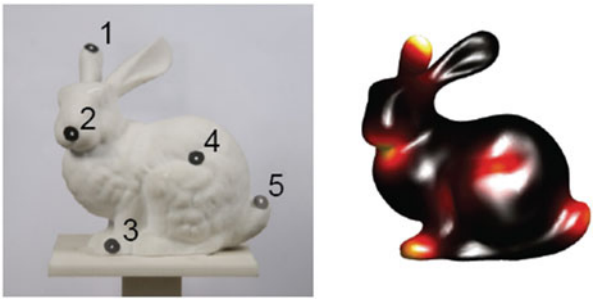


Fig. 7 *Left*: physical bunny model with target markers (numbers indicate order); *right*: heat map of obtained gaze directions

Table 1 Errors of individual markers on bunny

Marker index	1	2	3	4	5
Angular error (deg.)	0.578	1.128	0.763	0.846	0.729
Depth error (mm)	7.998	8.441	10.686	3.036	8.381

silhouettes. Since we determine the point on the object that best corresponds to the gaze direction, we obtain accurate results also around silhouettes. This is reflected in the preliminary experimental results where we obtain an average depth error of 7.71 mm at a distance of 553.97 mm, which corresponds to a relative error of less than 2 %, despite three of five targets being very close to a silhouette.

6 Discussion

The proposed method for estimating fixations on 3D objects is simple yet accurate. It is enabled by:

- generating stimuli using digital manufacturing to obtain precisely known 3D geometry without restricting its shape;
- utilizing fiducial markers in a known relative position to the geometry to reliably determine its position relative to a subject's head;
- using a projective mapping to relate 3D positions to 2D pupil coordinates.

We experimentally verified our approach using two explorative user studies. The results demonstrate that 11 correspondences suffice to reliably calibrate the mapping from pupil coordinates to 3D gaze locations with an angular accuracy of 0.8°. This matches the accuracy of 2D gaze tracking. We achieve a depth accuracy of 7.7 mm at a distance of 550 mm, corresponding to a relative error of less than 1.5 %.

With the popularization of 3D printing, our approach can be easily applied to a large variety of stimuli, and thus usage scenarios. At the same time, it is not restricted to 3D printed artifacts and can be employed as long as the geometry of an object is known, for example when manual measurement or 3D scanning has been performed. Our approach also generalizes to simultaneously tracking gaze with multiple objects, as long as the objects' position and orientation are unambiguously identified, e.g., by including fiducial markers. The tracking accuracy in such situations will be subject to future investigation.

We developed our approach for 3D gaze tracking to analyze viewing behavior for genuine 3D stimuli, and to explore what differences to 2D stimuli exist. Our approach in particular enables researchers to study visual saliency on physical objects without scarifying accuracy. Given the substantial amount of work on saliency and related questions that employed 2D stimuli for studying 3D objects, we believe this to be a worthwhile research question that deserves further attention.

We believe 3D gaze tracking will be a valuable tool for research in computer science, cognitive science, and other disciplines. The fast and simple calibration procedure (comparable to typical 2D calibration) that is provided by our approach enables researcher to extend their data collection without significantly changing their current workflow.

Acknowledgements This work has been partially supported by the ERC through grant ERC-2010-StG 259550 (XSHAPE). We thank Felix Haase for his valuable support in performing the experiments and Marianne Maertens for discussions on the experimental setup.

References

1. Abbott, W.W., Faisal, A.A.: Ultra-low-cost 3D gaze estimation: an intuitive high information throughput compliment to direct brain-machine interfaces. J. Neural Eng. **9**, 1–11 (2012)
2. Agarwal, S., Mierle, K., others: Ceres solver. http://ceres-solver.org. Cited 21 Dec 2015
3. Barz, M., Bulling, A., Daiber, F.: Computational modelling and prediction of gaze estimation error for head-mounted eye trackers. German research center for artificial intelligence (DFKI) research reports, p. 10 (2015). https://perceptual.mpi-inf.mpg.de/files/2015/01/gazequality.pdf. Cited 21 Dec 2014
4. Bradski, G.: The OpenCV Library. Dr. Dobb's. J. Softw. Tools **25**(11), 120, 122–125 (2000)
5. Bruce, N., Tsotsos, J.: Saliency based on information maximization. Adv. Neural Inf. Process. Syst. **18**, 155–162 (2005)
6. Cournia, N., Smith, J.D., Duchowski, A.T.: Gaze-vs. hand-based pointing in virtual environments. In: CHI'03 extended abstracts on human factors in computing systems, pp. 772–773. ACM (2003)
7. Duchowski, A.T., Pelfrey, B., House, D.H., Wang, R.: Measuring gaze depth with an eye tracker during stereoscopic display. In: Proceedings of the ACM SIGGRAPH Symposium on Applied Perception in Graphics and Visualization, p. 15. ACM (2011)
8. Essig, K., Pomplun, M., Ritter, H.: A neural network for 3D gaze recording with binocular eye trackers. Intern. J. Parallel Emerg. Distrib. Syst. **21**, 79–95 (2006)
9. Häkkinen, J., Kawai, T., Takatalo, J., Mitsuya, R., Nyman, G.: What do people look at when they watch stereoscopic movies? In: Woods, A.J., Holliman, N.S., Dodgson, N.A. (eds.) Stereoscopic Displays and Applications XXI, International Society for Optics and Photonics, Bellingham, Washington USA (2010)
10. Hanhart, P., Ebrahimi, T.: EYEC3D: 3D video eye tracking dataset. In: 2014 Sixth International Workshop on Quality of Multimedia Experience (QoMEX), pp. 55–56. IEEE (2014)
11. Heng, L., Li, B., Pollefeys, M.: Camodocal: automatic intrinsic and extrinsic calibration of a rig with multiple generic cameras and odometry. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1793–1800. IEEE (2013)
12. Hennessey, C., Lawrence, P.: Noncontact binocular eye-gaze tracking for point-of-gaze estimation in three dimensions. IEEE Trans. Biomed. Eng. **56**, 790–799 (2009)
13. Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., van de Weijer, J.: Eye tracking: a comprehensive guide to methods and measures. Oxford University Press, New York (2011)
14. Howard, I.P.: Preceiving in depth. Oxford University Press, Oxford (2012)
15. Huynh-Thu, Q., Schiatti, L.: Examination of 3D visual attention in stereoscopic video content. In: Rogowitz, B.E., Pappas, T.N. (eds.) IS&T/SPIE Electronic Imaging, pp. 78650J–78650J. International Society for Optics and Photonics, Bellingham, Washington USA (2011)
16. Jansen, L., Onat, S., König, P.: Influence of disparity on fixation and saccades in free viewing of natural scenes. J. Vis. **9**, 1–19 (2009)
17. Judd, T., Ehinger, K., Durand, F., Torralba, A.: Learning to predict where humans look. In: International Conference on Computer Vision, pp. 2106–2113. IEEE (2009)
18. Kassner, M., Patera, W., Bulling, A.: Pupil: an open source platform for pervasive eye tracking and mobile gaze-based interaction. In: Proceedings of the International Joint Conference on Pervasive and Ubiquitous Computing Adjunct Publication – UbiComp'14 Adjunct, pp. 1151–1160. ACM (2014)

19. Kensler, A., Shirley, P.: Optimizing ray-triangle intersection via automated search. In: IEEE Symposium on Interactive Ray Tracing, pp. 33–38. IEEE (2006)
20. Ki, J., Kwon, YM.: 3D gaze estimation and interaction. In: 3DTV Conference: The True Vision – Capture, Transmission and Display of 3D Video, pp. 373–376. IEEE (2008)
21. Koenderink, J.J.: Pictorial relief. Philos. Trans. R. Soc. A: Math. Phys. Eng. Sci. **356**, 1071–1086 (1998)
22. Lang, C., Nguyen, T.V., Katti, H., Yadati, K., Kankanhalli, M., Yan, S.: Depth matters: influence of depth cues on visual saliency. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) Computer Vision – ECCV 2012, pp. 101–115. Springer Berlin Heidelberg (2012)
23. Lee, J.W., Cho, C.W., Shin, K.Y., Lee, E.C., Park, K.R.: 3D gaze tracking method using purkinje images on eye optical model and Pupil. Opt. Lasers Eng. **50**, 736–751 (2012)
24. Maggia, C., Guyader, N., Guérin-Dugué, A.: Using natural versus artificial stimuli to perform calibration for 3D gaze tracking. In: Rogowitz, B.E., Pappas, T.N., de Ridder, H. (eds.) Human Vision and Electronic Imaging XVIII, International Society for Optics and Photonics, Bellingham, Washington USA (2013)
25. Mathe, S., Sminchisescu, C.: Dynamic eye movement datasets and learnt saliency models for visual action recognition. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) Computer Vision – ECCV 2012, pp. 842–856. Springer Berlin Heidelberg (2012)
26. Pfeiffer, T., Latoschik, M.E., Wachsmuth, I.: Evaluation of binocular eye trackers and algorithms for 3D gaze interaction in virtual reality environments. J. Virtual Real. Broadcast. **5**, 1860–2037 (2008)
27. Pfeiffer, T., Renner, P.: Eyesee3d: a low-cost approach for analyzing mobile 3d eye tracking data using computer vision and augmented reality technology. In: Proceedings of the Symposium on Eye Tracking Research and Applications, pp. 369–376. ACM (2014)
28. Ramanathan, S., Katti, H., Sebe, N., Kankanhalli, M., Chua, T.-S.: An eye fixation database for saliency detection in images. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) Computer Vision – ECCV 2010, pp. 30–43. Springer Berlin Heidelberg (2010)
29. Ramasamy, C., House, D.H., Duchowski, A.T., Daugherty, B.: Using eye tracking to analyze stereoscopic filmmaking. In: Posters on SIGGRAPH'09, p. 1. ACM (2009)
30. Ritter, J.: An efficient bounding sphere. In: Glassner, A.S. (eds.) Graphics Gems, pp. 301–303. Academic Press, Boston (1990)
31. Schneider, P.J., Eberly, D.: Geometric Tools for Computer Graphics. Elsevier science Inc., New York (2002)
32. Stellmach, S., Nacke, L., Dachselt, R.: 3d attentional maps: aggregated gaze visualizations in three-dimensional virtual environments. In: Proceedings of the International Conference on Advanced Visual Interfaces, pp. 345–348. ACM (2010)
33. Toet, A.: Computational versus psychophysical bottom-up image saliency: a comparative evaluation study. IEEE Trans. Pattern Anal. Mach. Intell. **33**, 2131–2146 (2011)

3D Saliency from Eye Tracking with Tomography

Bo Ma, Eakta Jain, and Alireza Entezari

Abstract This paper presents a method to build a saliency map in a volumetric dataset using 3D eye tracking. Our approach acquires the saliency information from multiple views of a 3D dataset with an eye tracker and constructs the 3D saliency volume from the gathered 2D saliency information using a tomographic reconstruction algorithm. Our experiments, on a number of datasets, show the effectiveness of our approach in identifying salient 3D features that attract user's attention. The obtained 3D saliency volume provides importance information and can be used in various applications such as illustrative visualization.

1 Introduction

Direct Volume Rendering (DVR) is commonly used for visualization of volumetric datasets generated by scanners in biomedical imaging (e.g., CT/MR) or by simulations in scientific computing. In DVR, a transfer function is used to classify features by assigning optical properties (i.e., color and opacity) to the scalar field. The visual appearance of various points along a viewing ray is influenced by local geometric attributes such as scalar values and its first and higher-order derivatives. However, as users' interests might be non-uniformly distributed spatially over the volume, traditional DVR can be ineffective in identifying regions of interest. For example, given a rendered image of human feet, doctors may be interested in the details of joints while non-expert users might be interested in the overall feet structure. Therefore, it is useful to understand where people focus in the 3D volume visualization and use such information to better highlight the regions of interest (ROI). In other words, we want to detect the saliency regions of volume datasets.

In this paper, we introduce an approach to detect the 3D saliency regions for a dataset based on the eye-tracking data from multiple viewing angles that are assembled in 3D using tomographic reconstruction. As the eye-tracking data is generated in 2D (i.e., image space), we collect user's saliency information from multiple projection images of the volume. Then, we locate the salient regions by

B. Ma (✉) • E. Jain • A. Entezari
University of Florida, 32611, Gainesville, FL, USA
e-mail: bbo@cise.ufl.edu

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_11

constructing the 3D saliency volume from back projection of 2D saliency maps. The produced 3D map assigns saliency values to the voxels of the original data, which is useful for various applications such as transfer function design and illustrative visualization.

2 Related Work

Several approaches have been developed to compute visual saliency by analyzing the measurements derived from the data without visual feedback. Itti et al. [4] used center surround operators to calculate the saliency map of a 2D image. Lee et al. [9] proposed a model of mesh saliency using center-surround filters with Gaussian-weighted curvatures. Kim et al. [8] provided an evaluation of the computational model of mesh saliency using eye-tracking data. Other approaches located visual saliency in 3D space using eye-tracking glasses [13–15]. They strove to estimate 3D gaze fixations based on the ray/object intersection method. The assumption of these approaches is the first object that is hit by the viewing ray is the target of the fixation. In contrast, in DVR users might focus on the interior structure while the viewing rays pass through the transparent exterior structure. Therefore, the simple ray/object intersection method cannot be applied to locate saliency in DVR images.

Eye tracking has been widely used in graphics and visualization for implicit data collection from users. Unlike traditional methods based on mouse clicks that could burden the user, eye tracking enables the analysis of users attention. Santella et al. [16] used a perceptual model together with eye-tracking data to produce abstracted painterly renderings. Jain et al. [5] used eye-tracking data to track readers' attention in comic book images and assess the artist's success in directing the flow of attention. Burch et al. conducted experiments to evaluate traditional, orthogonal, and radial tree diagrams [2]. Participants were asked to find the least common ancestor of a given set of leaf nodes and eye tracking was used to record their exploration behavior. Eye tracking has also been used in video re-editing to expose the important parts of the video [6]. Mantiuk et al. [12] improved the accuracy of headmounted eyetracker in 3D scenes using both eye-tracking data and prior knowledge of the environment. Instead of tracking in 2D image space, our work adds a new dimension by enabling the construction of 3D eye-tracking profile that can be used for various purposes including saliency detection. The constructed saliency volume can be directly employed for illustrative volume rendering [1, 19], which uses non-photorealistic rendering techniques to enhance important features or filter irrelevant details out. The saliency volume could also be useful for other visualization applications, such as progressive visualization [11] and volume visualization enhancement [7].

Lu et al. [10] is the closest previous work to this research. They used eye tracking to identify the salient points on individual isosurfaces from a volume and used it for parameter selection in direct volume rendering. They visualized isosurfaces (with transparency) to track the user's attention to various regions on

an individual isosurface while the volume is rotating. Users explore the features of the volume by selecting different isovalues while viewing the rotating volume. In this approach, a 3D focus point is identified by finding the intersection of two rays generated from a pair of fixation points from two frames corresponding to consecutive viewing angles. The 3D focus points can be located from fixation points on multiple consecutive frames if the user maintains viewing at the same 3D position. While this technique identifies salient regions on individual isosurfaces, it is challenging to track users attention among different isosurfaces specially since different isosurfaces are presented at different times during the exploration process. In our approach, we also collect the eye-tracking data while the users view a rotating volume; however, we use tomographic reconstruction to construct 3D saliency map, for the entire classified volume, based on the acquired saliency on each viewing angle. We discuss the basic principles in tomographic reconstruction in Sect. 4.

3 Saliency Information Acquisition

Since locating the ROI in a volume could be subjective and vary among users, we collect saliency information from users via an eye tracker (*Eyetribe*, 30 Hz tracking). As fixations [3] best indicate the location of the viewer's visual attention, thus identifying the fixations is the key to finding ROI. We use *Eyetribe* provided software to record fixation locations. As locating the saliency regions in 3D requires additional information besides the 2D saliency information (i.e., image space), we leverage the 2D gaze data from multiple viewpoints

We render the volume (Fig. 1a) based on a pre-defined transfer function. The transfer function can be specified manually or obtained from automatic transfer function design techniques [18, 20]. Obviously the choice of transfer function influences what features of the volume are visualized. Therefore, a proper transfer function is needed that can reveal the relevant features and details from the volume. While the choice of transfer influences the visibility of features and impacts the eye-tracking results, currently we assume that a transfer function, with semi-transparent features, that reveal the relevant structures is given. We envision our 3D eye-tracking system will be useful in providing feedback to the transfer function process (interactively or off-line). Next, as shown in Fig. 1b, we present the rotating volume to users so that they can freely view the interesting features. Under each viewpoint, we collect the gaze data which indicate the most attractive regions on the rendered image. The rotation axis is selected so that the respective projections can best reveal the features of the data. To collect enough gaze data, we rotate the volume 12° every 4 s which yield a set of 2D saliency information from 30 viewpoints (projections).

After the data collection, we analyze the fixations for each 2D projections image separately. We employ two off-line strategies for different requirement:

1. We analyze the fixations obtained from an individual, the gathered saliency information reveals the personal interest of the volume data. This is useful for a

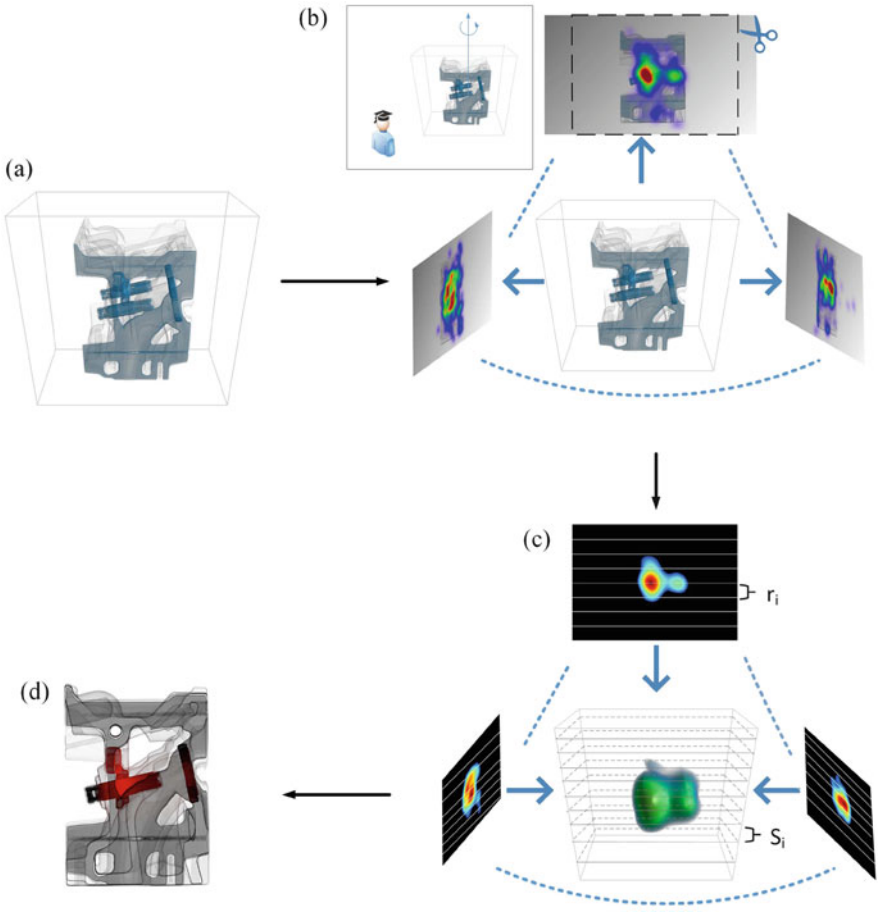


Fig. 1 The process of constructing the 3D saliency volume. (**a**) DVR volume. (**b**) Gaze data collection. (**c**) 3D saliency volume reconstruction. (**d**) Result

domain expert who knows the underlining data and desire to explore a particular part of the data.

2. We process the fixations from multiple users. For a volume, the users general interest is studied, and the regions that catch most of the attention can be identified.

The generated 3D saliency volume can, in turn, inform the transfer function design in DVR (discuss in Sect. 6). With either strategy, we have collected a set of 2D fixations that are the sources of our saliency information.

4 Saliency Volume Construction

Tomography has been widely used in medical (e.g., X-ray CT) and other scientific fields, including physics, chemistry, astronomy. It allows for the reconstruction of an object from its projections (e.g., shadows). Modern tomography involves gathering projection data via scanning the object from multiple directions using a specific X-ray source. Then, the original object can be reconstructed by feeding the projections into tomographic reconstruction algorithms.

Figure 2 shows an example of tomography that uses parallel beams to scan a 2D object $f(x, y)$ at a specific direction. For a given angle θ, the 1D projection of the 2D object is made up of a set of line integrals (shown as blue lines). The data collected at the sensor s_i is the line integrals of the beam b_i which represent the total attenuation of b_i as it travels through the object. $p_\theta(s)$ is the 1D projection of the 2D object $f(x, y)$ which is formed by combining a set of line integrals at angle θ. The extent of $p_\theta(s)$ is determined by the bounding rectangle of the object. Mathematically, $p_\theta(s)$ can be understood as the Radon transform of f:

$$p_\theta(s) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y)\delta(x\cos\theta + y\sin\theta - s)\mathrm{d}x\mathrm{d}y \tag{1}$$

In practice, a set of projections $p_\theta(s)$ from a finite set of directions can be obtained as outputs from a scanner where a rotating object is being scanned. Figure 3a shows an example where projections are collected from 4 angles. From

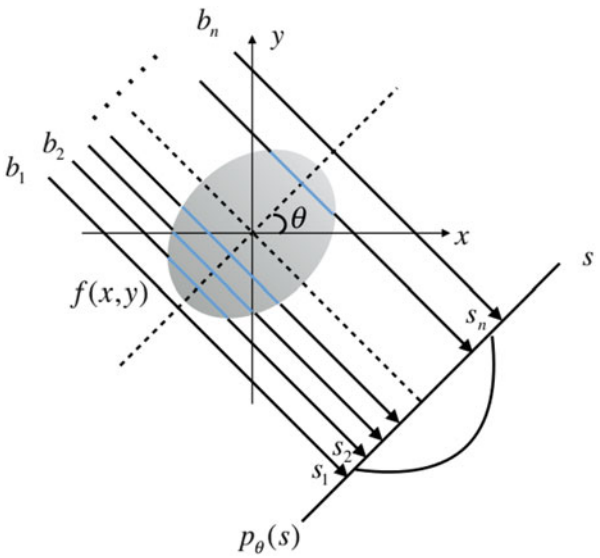


Fig. 2 Parallel beam tomography. Each projection is made up of the set of line integrals through the object

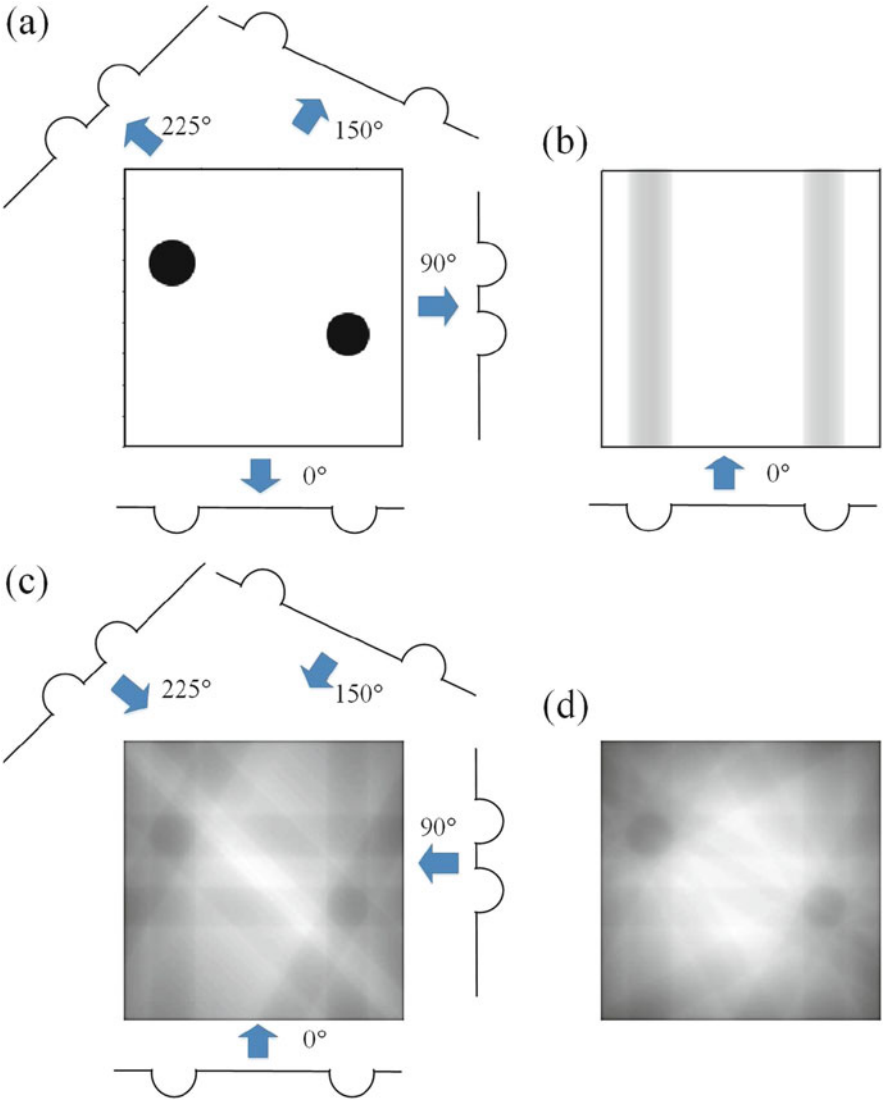


Fig. 3 (**a**) Scanning a 2D object and the corresponding projections. Back projection (**b**) at angle 0°, (**c**) using 4 projections, (**d**) using 12 projections

Fourier slice-projection theorem, we can reconstruct the original object $f(x, y)$ if we have projections $p_\theta(s)$ for many angles. The reconstruction is done by feeding these projections into tomographic reconstruction algorithms, such as filtered-back-projection (FBP) and iterative reconstruction algorithms, that compute the inverse Radon transform and approximate the absorption density of the original object. Figure 3b–d illustrate the basic idea behind back projection for reconstruction.

The projections obtained in Fig. 3a represent the total attenuations of the original object scanned at different angles. Therefore, for each projection, the reconstruction process runs the projection back through and evenly divides the attenuation along the path of the ray. Figure 3b shows an example of back projection at angle 0°. Figure 3c shows the back projection of 4 angles. Note that the original object is beginning to appear. A better reconstruction is achieved by using 12 projections as shown in Fig. 3d.

In our data acquisition process, we track the users viewing a rotating volume which is similar to the tomographic scan where rays are used to scan a rotating object. The saliency information is recorded as fixations in the rendered images that are generated from parallel projections of the volume. Therefore, the 2D saliency map in the image space can be back projected, for each projection angle, similar to the tomographic scans under parallel beam geometry. Intuitively, we can treat each back projection of a 2D saliency map as a 3D saliency estimation from a certain angle. When back projecting multiple 2D saliency maps, the result is the aggregated 3D saliency estimation from different perspectives.

Once we collect the gaze data for a particular angle, we build the 2D saliency map for that angle which is then back projected to the 3D volume in the tomographic reconstruction algorithm. Back projection, shown in Fig. 3, is simple and efficient but the resulting images are blurry. To correct the blurring encountered in back projection, the filtered-back-projection (FBP) algorithm filters (i.e., Ram-Lak filter) each projection image before back projection – compensating for the concentration of information (samples) in the center of the volume where all projections intersect and the lack of information away from the center. For more technical details we refer the reader to [17]. In our method, we use the commonly-used FBP to construct the 3D saliency volume from a set of 2D saliency maps. More accurate reconstructions can be obtained by using more projections that, in turn, increase the acquisition time. The necessary number of projection angles depend on the complexity of the 3D image that is being acquired. As discussed in Sect. 3, we have collected the fixations of 30 projection angles which provided a stable reconstruction, via FBP, of the 3D saliency map. Our preliminary investigation showed negligible improvements from increasing the number of projection angles (e.g., Fig. 4).

The construction of 3D saliency volume involves the following three steps:

1. **Generate 2D saliency map**: We construct the 2D saliency map from recorded gaze data by convolving the map of fixation locations with a Gaussian kernel.

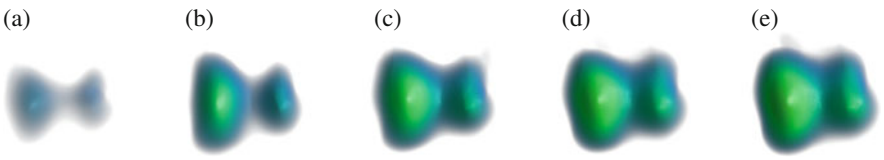


Fig. 4 The 3D saliency volumes of engine dataset. Saliency values from low to high are mapped to colors blue, green, red. (**a**) 12 angles. (**b**) 20 angles. (**c**) 25 angles. (**d**) 28 angles. (**e**) 30 angles

We calibrated subjects to $<0.5°$ visual angle. We removed noise in the saliency map after data collection by thresholding saliency values $<20\,\%$ of the maximum value. The choice of threshold directly influences the construction quality of the saliency volume. A small threshold may involve noisy data in the reconstruction process while a large threshold unnecessarily eliminates valid data. We found by choosing threshold $= 20\,\%$ can achieve a good balance. The size of the constructed saliency map is bounded to the size of the projection image which is shown with high resolution in order to locate user's gaze data accurately. Therefore, the underlying saliency map contains redundant background pixels that need to be removed before being fed into the reconstruction algorithm. As an example shown in the top images of Fig. 1b, c, we crop the 2D saliency maps according to the bounding box of the volume which makes it easier to map the constructed 3D saliency volume to the original volume (step 3).

2. **Construct 3D saliency volume**: Figure 1c shows the process of constructing the 3D saliency volume slice by slice from the rows of the 30 saliency maps. Each slice S_i is located in 2D space and the respective rows r_i of saliency maps are the 1D tomographic projections. We employ the MATLAB function `iradon` that uses the filtered back projection algorithm to compute the inverse Radon transform to construct each slice of the saliency volume.
3. **Resample 3D saliency volume**: Since the reconstructed volume from FBP does not necessarily align with the source volume, we use a resampling step to obtain saliency values for the voxels in the original volume.

After all the steps, we generate the 3D saliency volume that indicate the saliency of the voxels in the original volume.

5 Experiment

We conducted experiments on three volume datasets: an engine block, a CT human head and a carp fish. The volumes are rendered using a semi-automatic transfer function design approach that identifies a set of distinct representative isosurfaces from a dataset [18]. In order to understand users general interest to these data, 12 participants (9 male, 3 female, age from 21 to 28) were recruited. The participants were university students and were compensated with class credit for their participation.

Participants viewed the rotating volume on a 17-inch monitor (1920×1080). The rendered images were placed on the screen as large as possible to increase the accuracy of locating saliency regions. The participants were asked to adjust their chair height and distance to the monitor to make themselves comfortable. Then, the system was calibrated and the rotating volume was present to the participants.

For eye tracking, we used the *Eyetribe* eye tracker with a cloud-based analytical platform. It can locate the gaze locations on the calibrated screen at 30 Hz. The raw

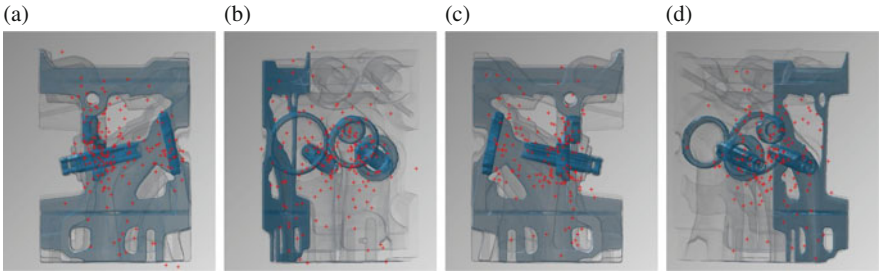


Fig. 5 Plot of the fixation locations on the respective 2D projection images. The original images are 1920×1080. The images shown here are zoomed in to see the fixations. (**a**) Projection at angle $0°$. (**b**) Projection at angle $120°$. (**c**) Projection at angle $192°$. (**d**) Projection at angle $252°$

data was converted into fixations and saccades inside EyeProof platform which can be exported as a spreadsheet for off-line analysis.

In our first experiment, we show the effectiveness of our approach on a CT engine block ($256 \times 256 \times 128$). The structure of the engine is relatively simple, and we anticipated that users would pay more attention to the two internal cylinders (see Fig. 1a). Figure 5a shows some examples of overlaying the fixations (from 12 participants) on the respective projection images. A similar pattern can be observed from these images, that is fixations distributed densely at the cylinder region. Thus, the 2D saliency regions indicated by the fixations matches our assumption. Then, we convolved the fixations with the Gaussian kernel ($\sigma = 50$) to generate the 2D saliency maps (1920×1080). The saliency maps were thresholded and cropped to size 1531×1080. It took 79 s to construct the saliency volume ($1080 \times 1080 \times 1080$) on a standard PC. The constructed volume is further resampled to match the original volume. The middle image of Fig. 1c shows a DVR of the 3D saliency volume where saliency values from low to high are mapped to color blue, green and red. Comparing to Fig. 1 c, the saliency volume generally has high saliency values at the internal cylinder region. Specifically, the left green ball corresponds to the cylinder parts camshaft and timing gears while the right green ball corresponds to the cylinder parts flywheel. Figures 1d and 6 display the visualizations of the engine volume by modifying the color of the voxels based on the saliency volume. For each voxel, a higher red value is assigned when it has a larger saliency value while the blue and green channel are absent. We can clearly see that the internal cylinders are highlighted as red. The ROI indicated by the saliency volume matches our 2D saliency inspection, which further confirmed our assumption and also validate our reconstruction approach. To further demonstrate the accuracy of our approach, we artificially generated fixations for different viewing angles which are corresponding to the same ROI (circled in Fig. 7a). Figure 7b, c show the fixations from two selected angles. The visualization of the saliency volume, shown in Fig. 7d, was generated using the process as mentioned above where we can observe that the region that was artificially selected in 2D projections is highlighted in the 3D saliency image.

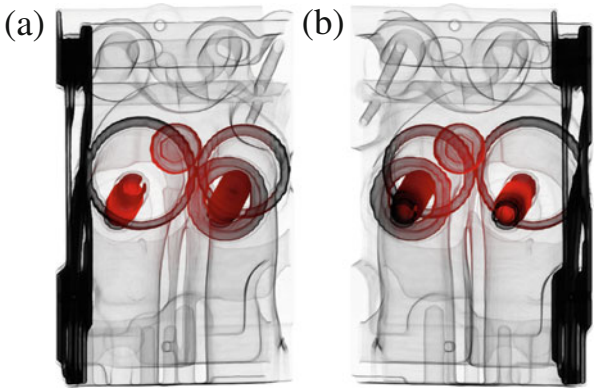


Fig. 6 Saliency volume visualization using red channel at different view angles: higher saliency voxels are assigned color with larger red value while the green and blue channels are absent. (**a**) Angle 84°. (**b**) Angle 264°

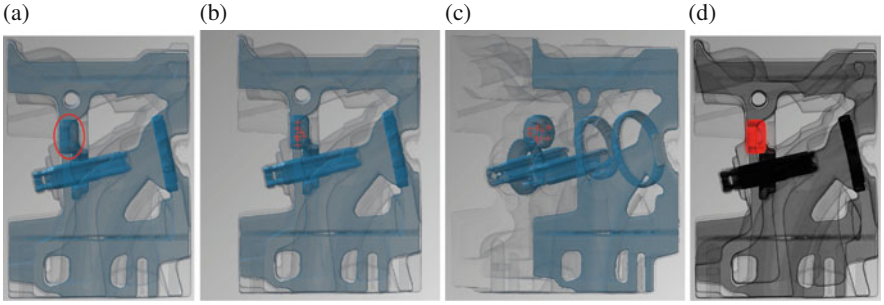
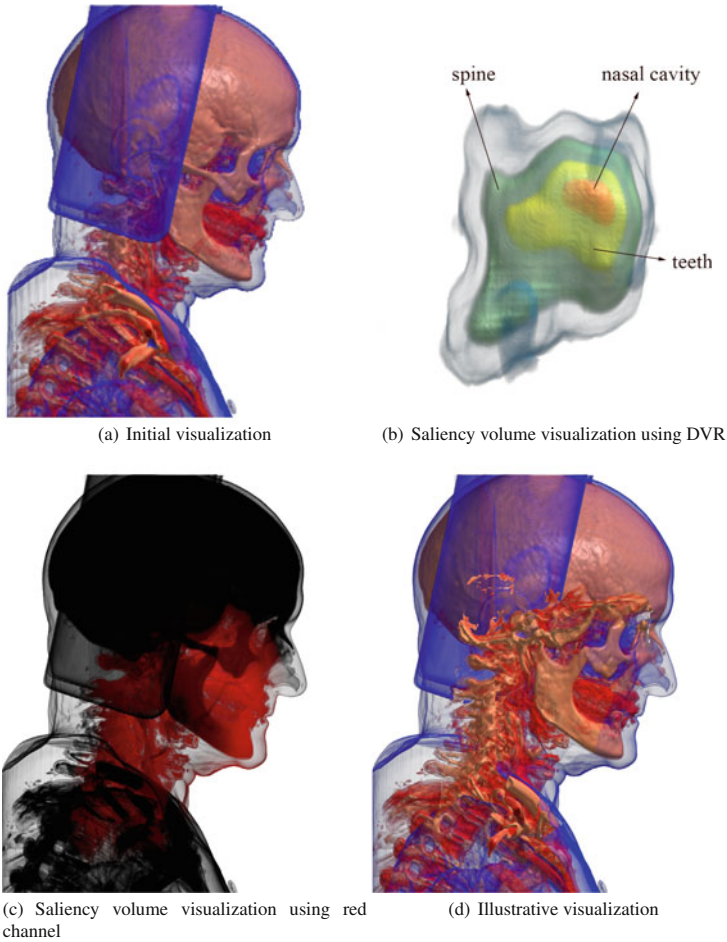


Fig. 7 Experiment using artificial fixations. (**a**) Original volume where the ROI is circled. (**b**), (**c**) Artificial fixations are placed at the ROI. (**d**) The visualization of the resulting saliency volume

We conducted our second experiment on a more complex dataset: a CT human head ($256 \times 256 \times 230$). Figure 8a shows an initially rendered image where a number of features are presented, such as side props, skin, skull, ribs, spine, vessels, and teeth. The 2D saliency maps (1920×1080) were generated by smoothing the fixations with a Gaussian kernel ($\sigma = 50$), then were thresholded and cropped to resolution 1249×1080. By feeding the 2D saliency maps into the reconstruction process, it takes 49s to construct the 3D saliency volume ($880 \times 880 \times 1080$) which is further resampled to match the original volume. Figure 8b shows the DVR of the generated 3D saliency volume where saliency values from low to high are mapped to colors blue, green, yellow, red. The image indicates that instead of the big structures (e.g., skull and ribs), users are more interested in the internal structures such as nasal cavity (shown as red) and teeth (shown as yellow). More interestingly, the constructed 3D saliency volume models the shape (vertical stick) of the partial spine with high saliency which further confirmed the effectiveness of the reconstruction



(a) Initial visualization (b) Saliency volume visualization using DVR

(c) Saliency volume visualization using red channel (d) Illustrative visualization

Fig. 8 Visualizations of CT human head at angle 270°: (**a**) Initial volume rendering based on a pre-defined transfer function. (**b**) Saliency volume visualization using DVR: saliency values from low to high are mapped to colors blue, green, yellow, red. (**c**) Saliency volume visualization using red channel: higher saliency voxels are assigned color with larger red value while the green and blue channels are absent. (**d**) Illustrative visualization: uninteresting regions (with low saliency values) are removed when they occlude interesting regions

process. The saliency regions can be more clearly perceived in Fig. 8c where higher saliency voxels are mapped to color with larger red value while the green and blue channels are absent. Figure 8d shows a simple illustrative volume rendering that makes use of the 3D saliency volume to remove less important parts of the volume to generate cut-away views. In the traditional DVR image (Fig. 8a), the high saliency spine region is occluded by the low saliency side props and skin. In the illustrative

(a) Initial visualization (b) Saliency volume visualization using red channel

Fig. 9 Visualizations of carp fish at angle 0°: (**a**) Initial volume rendering. (**b**) Higher saliency voxels are assigned color with larger red value while the green and blue channels are absent

rendering, the spine are expressed more clearly by removing part of the side props and skins that occlude it.

To further validate our approach, we conducted another experiment on a carp fish dataset ($128 \times 128 \times 128$). As shown in Fig. 9a, the initial rendering solidly exposed the fish head while kept the internal lung and bones clearly visible. We were curious whether users' attention will be attracted to the prominently rendered fish head or the internal details. With a similar procedure, the reconstruction process took 31 s to construct the saliency volume for this dataset. By visualizing the saliency volume using red channel, Fig. 9b shows that the saliency regions are located at the front fish bone (bright red) while the fish head is dark with low saliency. This observation along with the above-mentioned experiments suggest that users tend to focus on internal structures of the volume even though an outstanding exterior structure is shown. But, we need more experiment to verify it.

6 Discussion

We have presented an approach to construct the salient regions in a volume dataset. Our approach has shown to be effective in producing saliency volumes from the eye-tracking data for various datasets. In our process, the saliency information is obtained from users by rendering the volume from multiple viewpoints. The constructed saliency volume can in turn customize the transfer function to better highlight regions of interest.

For the purpose of exploratory visualization, a current limitation of our approach is that the detected saliency regions heavily depend on the choice of transfer function. The volume rendering based on the transfer function determines the areas that catch more users' attention and different transfer functions might result in different saliency volumes. For example, in Fig. 8a, vessels and teeth were rendered with color red that were stand out easily. If we assign them another color such as the color similar to the skull, then users focus might switch to other regions. However, a visual stimuli is needed to obtain saliency information from users, and it make sense to have different saliency regions for different stimuli. In our future work, we plan to use the saliency volume in a feedback loop to evaluate the design of transfer functions. We plan to render a volume with various transfer functions

that are to reveal same set of features. Then the constructed saliency volumes can be used to evaluate which transfer functions successfully attract users attention to the specified set of features. Further experiments could include the study of the differences between expert and non-expert viewers.

In the future, we also plan to improve the filtered back projection (FBP) for saliency volume construction. One improvement is to use the predefined transfer function to modify FBP. In standard FBP, each back projection assigns the same attenuation coefficient to all the locations along the ray. We plan to modify the FBP by increasing the attenuation coefficients at high opacity locations while reducing the ones at low opacity locations. As a result, the generated saliency volume will put more emphasize on solid regions. Another improvement is to extend FBP for more complex projections. In our current method, we let users look at a rotating volume and collect cylindrical-view projections. We plan to allow users to have a freeform interaction with the volume, i.e., they can freely rotate the volume during exploration. The result will be a set of spherical-view projections. Thus, an extended FBP is needed in order to construct saliency volumes from these projections. We also plan to extend our method to detect mesh saliency. Unlike regions in a rendered volume, regions on a mesh are only visible from limited angles due to occlusion of the opaque surface. Therefore, it is crucial to design a customized FBP to accurately locate 3D saliency regions with limited projections.

Acknowledgements This work was supported in part by the Office of Naval Research (N00014-16-1-2228) and US National Science Foundation (NSF IIS-1617101). The datasets are courtesy of the volvis community and OsiriX Foundation.

References

1. Bruckner, S., Gröller, E.: Style transfer functions for illustrative volume rendering. Comput. Graph. Forum **26**(3), 715–724 (2007). doi:10.1111/j.1467-8659.2007.01095.x. http://dx.doi.org/10.1111/j.1467-8659.2007.01095.x
2. Burch, M., Konevtsova, N., Heinrich, J., Hoeferlin, M., Weiskopf, D.: Evaluation of traditional, orthogonal, and radial tree diagrams by an eye tracking study. IEEE Trans. Visual. Comput. Graph. **17**(12), 2440–2448 (2011). doi:10.1109/TVCG.2011.193
3. Duchowski, A.: Eye Tracking Methodology: Theory and Practice, vol. 373. Springer Science & Business Media, London (2007)
4. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Trans. Pattern Anal. Mach. Intell. **20**(11), 1254–1259 (1998). doi:10.1109/34.730558
5. Jain, E., Sheikh, Y., Hodgins, J.: Inferring artistic intention in comic art through viewer gaze. In: Proceedings of the ACM Symposium on Applied Perception, SAP'12, pp. 55–62. ACM, New York (2012). doi:10.1145/2338676.2338688. http://doi.acm.org/10.1145/2338676.2338688
6. Jain, E., Sheikh, Y., Shamir, A., Hodgins, J.: Gaze-driven video re-editing. ACM Trans. Graph. **34**(2), 21:1–21:12 (2015). doi:10.1145/2699644. http://doi.acm.org/10.1145/2699644
7. Kim, Y., Varshney, A.: Saliency-guided enhancement for volume visualization. IEEE Trans. Visual. Comput. Graph. **12**(5), 925–932 (2006). doi:10.1109/TVCG.2006.174

8. Kim, Y., Varshney, A., Jacobs, D.W., Guimbretière, F.: Mesh saliency and human eye fixations. ACM Trans. Appl. Percept. **7**(2), 12:1–12:13 (2010). doi:10.1145/1670671.1670676. http://doi.acm.org/10.1145/1670671.1670676
9. Lee, C.H., Varshney, A., Jacobs, D.W.: Mesh saliency. ACM Trans. Graph. **24**(3), 659–666 (2005). doi:10.1145/1073204.1073244. http://doi.acm.org/10.1145/1073204.1073244
10. Lu, A., Maciejewski, R., Ebert, D.S.: Volume composition and evaluation using eye-tracking data. ACM Trans. Appl. Percept. **7**(1), 4:1–4:20 (2010). doi:10.1145/1658349.1658353. http://doi.acm.org/10.1145/1658349.1658353
11. Machiraju, R., Fowler, J.E., Thompson, D., Soni, B., Schroeder, W.: Evita-efficient visualization and interrogation of tera-scale data. In: Data Mining for Scientific and Engineering Applications, pp. 257–279. Springer, Boston (2001)
12. Mantiuk, R., Bazyluk, B., Mantiuk, R.K.: Gaze-driven object tracking for real time rendering. Comput. Graph. Forum **32**(2), 163–173 (2013). doi:10.1111/cgf.12036. http://diglib.eg.org/EG/CGF/volume32/issue2/v32i2pp163-173.pdf
13. Paletta, L., Santner, K., Fritz, G., Mayer, H., Schrammel, J.: 3D attention: measurement of visual saliency using eye tracking glasses. In: CHI'13 Extended Abstracts on Human Factors in Computing Systems, pp. 199–204. ACM (2013)
14. Pfeiffer, T.: Measuring and visualizing attention in space with 3D attention volumes. In: Proceedings of the Symposium on Eye Tracking Research and Applications, ETRA'12, pp. 29–36. ACM, New York (2012). doi:10.1145/2168556.2168560. http://doi.acm.org/10.1145/2168556.2168560
15. Pfeiffer, T., Renner, P.: Eyesee3D: a low-cost approach for analyzing mobile 3D eye tracking data using computer vision and augmented reality technology. In: Proceedings of the Symposium on Eye Tracking Research and Applications, ETRA'14, pp. 195–202. ACM, New York (2014). doi:10.1145/2578153.2578183. http://doi.acm.org/10.1145/2578153.2578183
16. Santella, A., DeCarlo, D.: Abstracted painterly renderings using eye-tracking data. In: Proceedings of the 2nd International Symposium on Non-Photorealistic Animation and Rendering, NPAR'02, pp. 75–ff. ACM, New York (2002). doi:10.1145/508530.508544. http://doi.acm.org/10.1145/508530.508544
17. Smith, S.W.: The Scientist and Engineer's Guide to Digital Signal Processing. California Technical Pub., San Diego (1997)
18. Suter, S., Ma, B., Entezari, A.: Visual analysis of 3D data by isovalue clustering. In: Advances in Visual Computing. Lecture Notes in Computer Science, vol. 8887, pp. 313–322. Springer International Publishing (2014). doi:10.1007/978-3-319-14249-4-30. http://dx.doi.org/10.1007/978-3-319-14249-4-30
19. Viola, I., Kanitsar, A., Groller, M.E.: Importance-driven volume rendering. In: Proceedings of the Conference on Visualization'04, pp. 139–146. IEEE Computer Society (2004)
20. Zhou, J., Takatsuka, M.: Automatic transfer function generation using contour tree controlled residue flow model and color harmonics. IEEE Trans. Visual. Comput. Graph. **15**(6), 1481–1488 (2009). doi:10.1109/TVCG.2009.120

Visual Data Cleansing of Low-Level Eye-Tracking Data

Christoph Schulz, Michael Burch, Fabian Beck, and Daniel Weiskopf

Abstract Analysis and visualization of eye movement data from eye-tracking studies typically take into account gazes, fixations, and saccades of both eyes filtered and fused into a combined eye. Although this is a valid strategy, we argue that it is also worth investigating low-level eye-tracking data prior to high-level analysis, because today's eye-tracking systems measure and infer data from both eyes separately. In this work, we present an approach that supports visual analysis and cleansing of low-level time-varying data for eye-tracking experiments. The visualization helps researchers get insights into the quality of the data in terms of its uncertainty, or reliability. We discuss uncertainty originating from eye tracking, and how to reveal it for visualization, using a comparative approach for disagreement between plots, and a density-based approach for accuracy in volume rendering. Finally, we illustrate the usefulness of our approach by applying it to eye movement data recorded with two state-of-the-art eye trackers.

1 Introduction

We start earlier than the typical process of eye-tracking analysis and visualization, and argue that a separate visualization of low-level time-varying data can help explore the eye movements regarding reliability. Due to the wide variety of eye-tracking experiments we introduce a generic reference workflow (Sect. 3). Our contributions are a discussion of how we can model uncertainty in the context of eye tracking (Sect. 4), a cleansing technique for time-series oriented eye-tracking data (Sect. 5), and a visualization technique that reveals uncertainty of the left, right, and combined eyes (Sect. 6).

We demonstrate our cleansing approach and visualization technique by applying it to eye-tracking data from previous eye-tracking studies conducted with Tobii T60XL and SMI RED250 eye-tracking devices (Sect. 8). As a major outcome, we

C. Schulz (✉) • M. Burch • F. Beck • D. Weiskopf
Visualization Research Center, University of Stuttgart, Allmandring 19, 70569, Stuttgart, Germany
e-mail: Christoph.Schulz@visus.uni-stuttgart.de; Michael.Burch@visus.uni-stuttgart.de; Fabian.Beck@visus.uni-stuttgart.de; Daniel.Weiskopf@visus.uni-stuttgart.de

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_12

find differences over time between left, right, and combined eyes while visually inferring credibility of the recorded data.

Furthermore, we provide our implementation under the terms of the MIT-License (Sect. 7).

2 Related Work

Much of the previous related work on eye tracking, data cleansing, and uncertainty occurs within isolated domains.

Eye-Tracking Hardware: Singh and Singh [29] and Al-Rahayfeh and Faezipour [3] provide reviews of anatomical and technical aspects of eye tracking in general, which are used for discussion later on.

Eye-Tracking Quality: Holmqvist et al. [18] note that standardized metrics would be of great help when assessing eye-tracking data quality. They further argue that fixation filters and correlation with areas of interest may actually hide errors. Netzel et al. [23] increase fixation data quality through manual annotation of fixations. To our knowledge, it is much more common to enhance study quality by reducing measurement errors introduced by sampling frequency [4] or user movement [5, 12] than communicating uncertainty present in recorded data. In contrast, we specifically show the reliability of the data as a basis for user-controlled improvements of the data quality.

Eye-Tracking Visualization: Eye movements recorded during eye-tracking studies are typically analyzed and visualized by temporal aggregation like in attention maps [22]. While this allows us to derive hot spots [10] of visual attention, we cannot analyze time-varying patterns. If gaze plots [13] are used, the time-varying behavior is explicitly encoded in the visual representation, but for long-lasting tasks and a larger number of study participants, the amount of visual clutter increases, making such a visualization difficult to read. Many visualization techniques have already been developed to analyze eye movement data for patterns [8], but most of them only take aggregated eye movements into account. Hence, we base our work on simple line plots, scarf plots [25], and space-time cubes [21]. We furthermore focus on including uncertainty visualization, which has largely been ignored in eye-tracking visualization so far.

Time-Series Visualization: Eye-tracking data is sampled and time-dependent, so we would like to point out a variety of techniques to visualize different aspects of time [2]. Stacking representations of data were previously used by Shahar et al. in their tool called KNAVE-II [28] in the clinical domain with a focus on semantic navigation. VisuExplore [26] and CareCruiser [15] also use stacked representations of time series while focusing on medical use cases and structure of patient data. Beard et al. [7] describe a system to explore spatial and temporal patterns of sensor data, while dealing with uncertainty of missing data to some extent. We

transfer the idea of stacked representations to eye tracking by including specialized representations for eye-tracking data.

Uncertainty Visualization: We apply a rationale by Skeels et al. [30] to eye tracking, stating that visualizing uncertainty could help make better decisions. Furthermore, we base our discussion on a review of uncertainty visualization by Brodlie et al. [11]. We distinguish between an accuracy-based and a comparative approach in terms of uncertainty.

Data Cleansing: Rahm and Do [24] classify data quality problems for data cleansing in the data warehouse domain. Their definition of single-source and multi-source problems transfers well to our work. Kandel et al. [20] describe a technique to interactively infer mapping functions from manipulation of data, but they do not deal with sequential, time-varying data. Gschwandtner et al. [16] propose design principles and techniques to exploit time specifics for data cleansing, but they do not visualize or propagate different facets of uncertainty originating from a processing pipeline, such as eye tracking.

3 Experiment Workflow

Before discussing uncertainty in eye tracking, we elaborate on the integration of our tool into a reference experiment workflow, shown in Fig. 1: An eye-tracking experiment is designed by a researcher and executed to obtain a recording of time-varying data for each participant. It may be very hard to analyze raw recordings, because recordings may require segmentation and re-ordering, contain recognition errors, and originate from multiple eye trackers. Therefore, proper and comprehensible data cleansing prior to analytics can reduce data quality issues by identifying corrupt data and making data consistent. Our approach to data cleansing is iterative and based on visual feedback. Additionally, a description of how the cleansing was done should be incorporated into the final results, because cleansing has the same potential to hide errors, as fixation filters do [18].

Considering the range of available eye trackers and different types of experiments, we made as few assumptions as possible about hardware and use cases to keep our approach generic. We assume that data is a time-dependent series

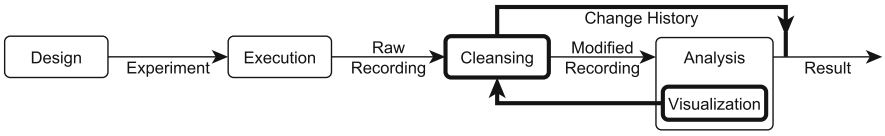


Fig. 1 Reference workflow for eye-tracking experiments. Experiments are designed and executed to gain raw recordings, which can be cleansed and analyzed to obtain a result. The final results of the study are influenced also by how the cleansing was performed

of samples and for implementation reasons, we limit ourselves to stationary eye-tracking setups using static images and video stimuli. We tested a maximum duration of about half an hour per recording, even though our implementation should perform well beyond that.

4 Modeling Uncertainty in Eye-Tracking Data

Our model is based on one question: What reveals flawed data and separates it from trustworthy data, and thus is crucial for decision making during cleansing? This question leads to the topic of uncertainty, which introduces data analysis challenges that we will discuss in the following sections. We adhere to the term uncertainty of the visualization community. Other communities prefer data confidence, quality, or trust.

4.1 Background

We adopt a classification by Skeels et al. [30] to discuss different aspects of uncertainty in the context of eye tracking. Their classification distinguishes between measurement, completeness, inference, disagreement, and credibility (see Fig. 2a). Measurement uncertainty describes accuracy and precision. Completeness uncertainty describes aggregation, missing values, and sampling. Inference uncertainty describes modeling, prediction, and retrodiction. The former three levels are stacked because uncertainty propagates from bottom to top (right part of Fig. 2a), whereas disagreement and credibility on the left span all three levels as derived characteristics (left part of Fig. 2a).

We apply this classification to a simplified eye-tracking pipeline, condensed from related work [3, 27, 29, 32] and depicted in Fig. 2b to illustrate sources of uncertainty originating from eye tracking. The process of optical eye tracking starts with light hitting a raster of sensor pixels, aggregated to a sequence of images, forming a video. This process introduces measurement uncertainty, because of physical

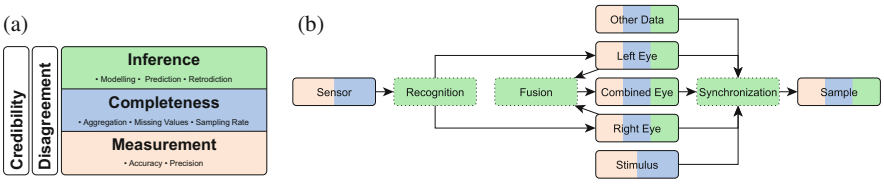


Fig. 2 (**a**) Classification of uncertainty and (**b**) eye-tracking pipeline, introducing and propagating different kinds of uncertainty. Each eye is recognized individually and then fused to a combined eye. Subsequently, all data is synchronized and emitted as samples

properties such as lenses, pixel density, and signal-to-noise ratio, and completeness uncertainty, because of missing eye movements due to a low sampling rate, and the light conditions might be too bad for the pixels to work properly. Subsequently, inference uncertainty is introduced, as each eye is recognized independently, fused into a combined eye, and synchronized with other data, e.g., keyboard and mouse events composing a sample.

Many eye trackers address uncertainty algorithmically, e.g., internal latencies get canceled out, and missing values are estimated using a co-simulation of the participant's eyes [33]. Most vendors provide uncertainty-related information as a part of technical specifications and recorded data, e.g., angular gaze accuracy, sampling resolution, and recognition confidence. Hence, an eye-tracking device exhibits all three levels of uncertainty and many of its sub-systems have to be considered as black boxes. Unfortunately, this means that we have to rely on information provided by vendors, which limits our basis for revealing uncertainty.

As examples, we have examined several Tobii and SMI eye-tracking systems, shown in Tables 1 and 2. They provide quite different quality metrics and technical specifications: Tobii defines a validity code to represent the success of recognition, whereas SMI emits several Boolean and ordinal values for conveying recognition confidence and timing issues with device-dependent availability. With our generic model and handling of uncertainty, we aim to cover this whole variety of technology-driven descriptions of data uncertainty.

We believe that visualizing uncertainty helps researchers find disagreement in data and estimate a credibility of their recordings. Formally, we model and visualize two different types of uncertainty: An accuracy and precision uncertainty model that is based on probability density functions (PDFs) for spatial and temporal dimensions of data, and a failure uncertainty model that is based on black-box metrics emitted by the eye tracker, that indicate whether the data sample was invalidated (Table 2). Visualizing these models allows us to distinguish during cleansing between measurement-related issues, recognition failures, and ambiguities.

Table 1 Measurement-related technical data extracted from vendor documentation, listing upper bounds in case of doubt

Device	Sampling rate [Hz]	Accuracy [°]	Latency [ms]
Tobii			
T60	60	0.5	33
T120	120	0.5	33
SMI			
REDn scientific	30 or 60	0.5	25
RED 250 mobile	60, 120, or 250	0.4	8
RED 500	500	0.4	4
Eye-tracking glasses 2.0	30 or 60	0.5	Measured

Table 2 Quality metrics recorded by vendor software

Field	Description	Scale
Tobii Studio		
Validity(Left|Right)	Validity code for each eye	Nominal
SMI iView		
(L|R) Validity	General quality value for each eye	Boolean
Pupil Confidence	Validity of the pupil diameter	Boolean
Timing	Indicates a timing violation	Boolean
Latency	Required time to process a sample	μs

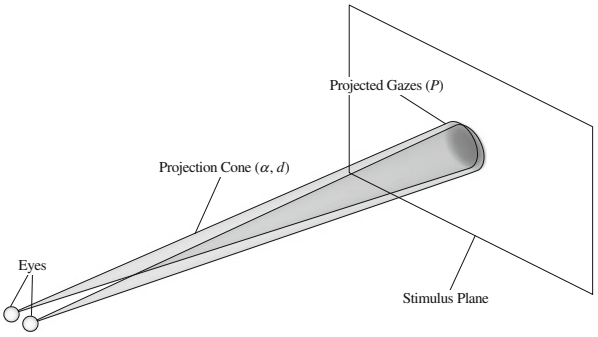


Fig. 3 Projection from eye space to gaze space using cones. Measurement imprecision and disagreement between the left and right eye are exaggerated for illustration purposes

4.2 Gaze Data

We focus on uncertainty present in gaze positions for the eye $P \subset \mathbb{R}^2$, and time $T \subset \mathbb{R}$, modeled as PDF. For one sample, the PDF reads:

$$\rho_P : T \times P \to [0, 1] \tag{1}$$

Here ρ_P describes the probability density of a gaze position to take a given value.

Visually, we aim at propagation of uncertainty by expanding every projection line to a cone (Fig. 3). Note that this cone does not model aspects of the human visual system, such as foveal acuity or gaze contingency. This is only about transforming accuracy and precision of the eye tracker to gaze space, i.e., we need to have all data in gaze space. Since the accuracy σ_P is not directly provided in gaze space, we estimate it using σ_α provided in eye space (Table 1) and the eye–stimulus distance d:

$$\sigma_P = d \ \sin \sigma_\alpha \tag{2}$$

When exploring the density-based visualization shown later on, one should be aware how d was determined, i.e., manually measured or by guessing, because the impression of accuracy and precision can be deceiving otherwise. Accuracy is the proximity of measured samples to the true direction the eye is looking. This is not to be confused with precision, denoting reproducibility of the measurement. Hence, we argue that a normal distribution with 2σ should be assumed, because it covers 95 % of its area. This allows us to define a PDF for each gaze position $\mathbf{p} = (p_x, p_y)$ and accuracy $\boldsymbol{\sigma}_\mathbf{P} = (\sigma_x, \sigma_y)$:

$$\varphi(\mathbf{p}; \boldsymbol{\sigma}_\mathbf{P}) = \varphi_0 e^{-\left(\frac{(p_x(2/\sigma_x))^2}{2} + \frac{(p_y(2/\sigma_y))^2}{2}\right)} \tag{3}$$

Here we assume that the sample position at the origin and normalization using φ_0.

Time can me modeled as simple box function at time t, and sample intervals Δt, if we assume a uniform probability distribution within a temporal sampling interval:

$$\xi(t; \Delta t) = \frac{1}{\Delta t} \begin{cases} 1 & \text{if } t \in [-\Delta t/2, \Delta t/2] \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Again the sample time is at the origin.

We assume that the probability densities for spatial position, φ, and time, ξ, are independent. Therefore, the overall probability, ρ_P, is obtained by multiplication:

$$\rho_P(t, \mathbf{p}; \Delta t, \boldsymbol{\sigma}_\mathbf{P}) = \varphi(\mathbf{p}; \boldsymbol{\sigma}_\mathbf{P})\xi(t; \Delta t) \tag{5}$$

This could be easily extended to binocular vision by adding ρ_P of the left and right eye.

In addition to accuracy and precision issues, the eye tracker can invalidate the sample, which is denoted by metrics, shown in Table 2. Hence, the PDF is only valid if the eye tracker did not invalidate the sample. Invalid samples have to be treated separately by any subsequent processing, in particular by our visualization techniques.

If it is not possible to determine an eye–stimulus transformation, as in Fig. 3, because the required depth information is missing, the depth value needs to be approximated. Most eye trackers make an educated guess by defining the recorded video as stimulus, because the camera–eye distance can be measured quite well. Although this is a good approximation, it poses problems when dealing with multiple recordings, because finding a common space can be a hard problem of itself, especially for mobile devices. Fortunately, it is easy to solve for stationary devices, which allows combined cleansing and preliminary analysis of multiple recordings.

4.3 Signal and Event Data

A recording may include other data that helps put gaze data into context. Signals $S \subset \mathbb{R}^n$, such as electroencephalography (EEG), motion capturing, or eye tracker latencies, can be represented as time-series data analogously to gaze data. Again, we use a PDF to describe uncertainty:

$$\rho_S : T \times S \to [0, 1] \tag{6}$$

Events with parameters $E \times P$, such as keyboard, mouse, and touch input, are considered discrete. We neglect domain-specific uncertainty models for events, e.g., a model for accidental keyboard strokes, and thus only time T may be augmented by a PDF, i.e.:

$$\rho_E : T \to [0, 1] \tag{7}$$

Essentially, this is a generalization of our uncertainty model to fit all remaining aspects of a recording. For the sake of simplicity, we do not deal with manually annotated events [9].

5 Cleansing Technique

Our processing model is based on the pipes and filters pattern, also used by ParaView [1] and VisTrails [6]. Data is interpreted as immutable and processed by a pipeline composed of functions. A function can perform any non-destructive mapping of data. Formally, we want to setup a processing graph $G = (F, C)$ without cycles composed of functions F, a sink function $s \in F$, and connections C. If data is pushed or pulled, all affected functions are recomputed according to their dependencies in the graph. We have chosen this design because of its flexibility. Uncertain and certain data are treated as values V, i.e.:

$$f : T \times V_1 \times \ldots \times V_n \to T \times V_1 \times \ldots \times V_m \tag{8}$$

Visual cleansing means inspecting a function's visual response, while adjusting (optional) parameters to manipulate data. To illustrate this concept, we describe a couple of use cases and functions:

Velocity, Acceleration, and Jerk might be of interest in general for any time-varying positional data. All values can be obtained by simply chaining a differential operator up to three times.

Filtering is likely to be useful for repairing corrupt data. Such a function could do interpolation if a trigger signal is set, and pass-through otherwise. Another approach would be to reconstruct corrupt data using approximation.

Let us assume a simple cleansing function that drops data if a trigger signal is set and does pass-through otherwise. Depending on the amount of data gathered, tuning this trigger signal can be a tedious task. We spare users from small-scale work, by the following approach: We employ trigger signal emitting functions that analyze values, such as recognition confidences or other quality metrics, which can then be fine-tuned and used for other cleansing functions. Assuming the processing graph was set up correctly, this approach allows going from macroscopic to microscopic cleansing iteratively and quickly.

6 Visualization Technique

Our technique uses a stack of specialized, time-aligned visualizations. Time runs from left to right and the stack is sorted in a user-defined fashion. Plot-based stack elements reveal gaps in the data, disagreement between the individual and combined eyes, i.e., the failure uncertainty model. Volume-based stack elements reveal measurement imprecision, i.e., the accuracy and precision uncertainty model. In the following sections, we will discuss how uncertainty connects cleansing and visualization.

6.1 Stereo Plot

We use stereo plots for eye-related data—the name originates from the fact that the right eye plot is flipped below the left eye plot and the combined eye is rendered on top of both. Stereo plots manifest as line plots and scarf plots [25]. Of course, they may be used without this little stereo twist as well, like shown later on. In terms of our uncertainty model, stereo plots visualize the failure uncertainty model through comparison—the accuracy and precision uncertainty model is omitted here.

The line variant is used for ratio-scale data, as shown in Fig. 4. Differences between the individual eyes can be inferred by comparing the top plot Ⓛ and bottom plot Ⓡ. Both plots are overlaid with thick lines Ⓒ, representing the combined eye.

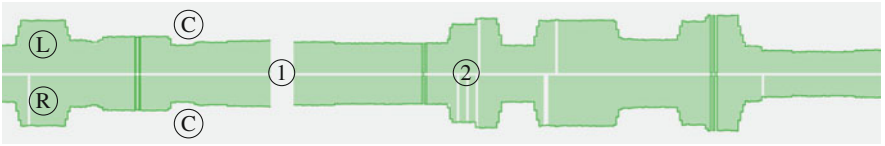


Fig. 4 A stereo line plot showing the left, right, and combined eye (gaze position). The right eye chart is flipped below the *left eye* plot and the combined eye plot is drawn as overlay of the top and bottom plot

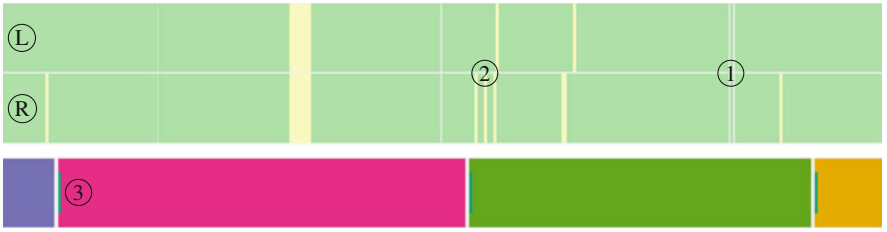


Fig. 5 *Top*: A stereo scarf plots, showing validity over time (*green*: good, otherwise: error) for the left and right eye. The right eye chart is flipped below the left eye plot. *Bottom*: A scarf plot showing activity (big bars: stimulus visibility, small bars: user input)

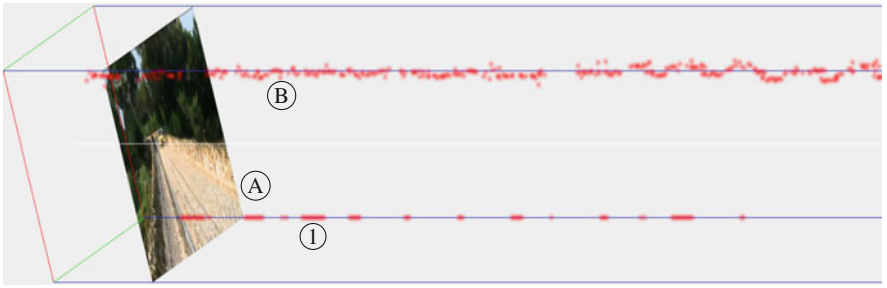


Fig. 6 An orthographic space-time cube showing gaze positions over time (from *left* to *right*), augmented by a PDF along with a video plane to provide some additional context

Undefined samples originating from the eye tracker or cleansing are depicted as gaps shown at ①. Disagreement can be observed at ②.

The scarf plot variant is used for nominal data such as validity codes and events, as shown in Fig. 5. Again, differences can be inferred by comparing the top scarf plot Ⓛ and bottom scarf plot Ⓡ. The combined eye is missing because there was no nominal data for the combined eye. Undefined samples are shown at ①, disagreement is shown at ②. We also use non-stereo scarf plots to depict concurrent activities, hence a scarf plot may be subdivided vertically to indicate concurrency, depicted at ③. We use ColorBrewer palettes [17] to encode quantities by color.

6.2 *Space-Time Cube*

We visualize gaze positions using a space-time cube, as shown in Fig. 6, containing a video plane Ⓐ, similar to the one by Kurzhals and Weiskopf [21], except that we use an orthographic instead of a perspective projection and volumes Ⓑ instead of solid points for rendering. The former prevents distortions around a vanishing point at the cost of a natural feeling of depth. The latter allows us to encode more information such as uncertainty. Hence, our space-time cube provides a visual impression of the

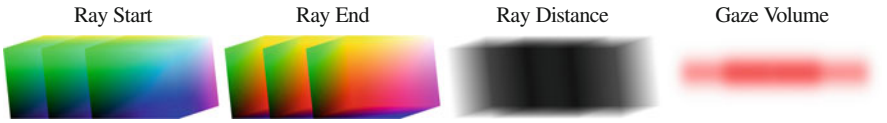


Fig. 7 Volume rendering of three gazes. The bounding box is intersected with the camera-ray in the fragment shader (ray start, ray end, and ray distance). Afterward, ray marching is applied to compute the gaze volume. All densities are blended together afterward

accuracy and precision uncertainty model using density, i.e., the failure uncertainty model is neglected. Low density equals high uncertainty. The video plane provides some additional context during cleansing. At ① magic values emitted by the eye tracker can be observed, while Ⓑ indicates fixations. This would be hard to see if gaze positions were depicted using solid points of arbitrary size. Remember that we use a box function to visualize time and a Gaussian function for spatial precision and accuracy, which allows us to do proper scaling of time and space.

7 Implementation

We provide our C++/Qt-based implementation under the terms of the MIT-License on GitHub.[1]

The pipes and filters pattern maps directly to code, e.g., data is loaded by a *LoadCSV* class, passed to cleansing function classes, and the visualizations are managed by *Display* classes. In terms of rendering, we took two different approaches. Stereo plots and other simple elements are rendered using the *QPainter* API, while space-time cubes are rendered using native OpenGL.

We had to mix a couple of rendering approaches to achieve crisp images and decent performance for our space-time cube. The basic idea is splatting of ray-casted volumes. Splatting is done additive, as shown by Hopf et al. [19]. Each sample is ray cast, as done by Stegmaier et al. [31]. This is different from what Djurcilov et al. [14] do for volume rendering of uncertainty, i.e., their approach of storing the density information inside a volume texture is not feasible for sparse data, such as gaze positions, because of memory constraints.

Gaze points are passed as vertex attributes to the shader program. All points are transformed in the vertex shader and expanded to a cube in the geometry shader. Afterward, we apply ray marching and splatting, shown in Fig. 7. This allows us to render the gaze positions using one draw call, resulting in decent performance. Note that the video plane and axes are rendered using separate draw calls.

[1] https://github.com/schulzch/BinocularVis

8 Case Study

The case study aims at showing how to apply the visualization technique, i.e., how to identify typical relationships between dimensions of eye-tracking data that indicate an error. We have used two data sets for this case study.

The first data set is from a study with five participants using a Tobii T60XL and Tobii Studio 2.2.8. The test was conducted in a distraction-free room, illuminated with diffuse light. The participants had to match a line to a pair of dots with varying line lengths and point distances, leading to very fast, comparative eye movements. The second data set is from a study with fifteen participants using an SMI RED250 and iView 2.8.26. The test was conducted under similar conditions. The participants had to interact with small button-sized input elements of high information density, depicting a probability density function, leading to very subtle gaze position changes. We consider both studies as typical representatives and potentially vulnerable to accuracy, precision, and missing data issues. We start with a description of all visual representations of eye-tracking data used in our case study:

Time is represented as simple ruler and measured in seconds.

Activity is represented as scarf plot. Long bars depict the duration of image stimuli, i.e., their visibility. Short bars (at the end of stimuli) depict mouse clicks by the participant.

Validity is represented as stereo scarf plot for each eye individually. Combined eye confidence is not emitted by the eye tracker. Light green depicts "all fine", other colors depict "error".

Processing Latency is represented as non-stereo line plot and measured in microseconds. SMI devices start to drop samples if latency is too high.

Camera–Eye Distance & Pupil Sizes are represented as stereo line plots. The former is the Euclidean distance computed from the eye coordinates. The latter is an estimate of the true pupil's size.

Gazes are represented as space-time cube containing density splats and a video plane for additional context.

Gaze X & Y are represented as stereo line plots. In addition to the individual eyes, the combined eye is depicted using a darkened line.

Gaze Velocity is derived from gaze coordinates through differentiation and represented as stereo line plot. Again, the combined eye is depicted using a darkened line.

We will demonstrate cleansing by visually inspecting slices Ⓐ to Ⓓ from the first data set shown in Fig. 8 and draft possible steps for cleansing.

Slice Ⓐ shows minor recognition jitter for both eyes in the validity plot around 1.5 s, which reveals corrupt values in the plots below. Note how the dark lines of the combined eye in the gaze plots do not intersect with the light colored areas of the individual eye data. This allows us to infer that the eye tracker has repaired those errors during eye fusion. If one wants to compare the left, right, and combined eye

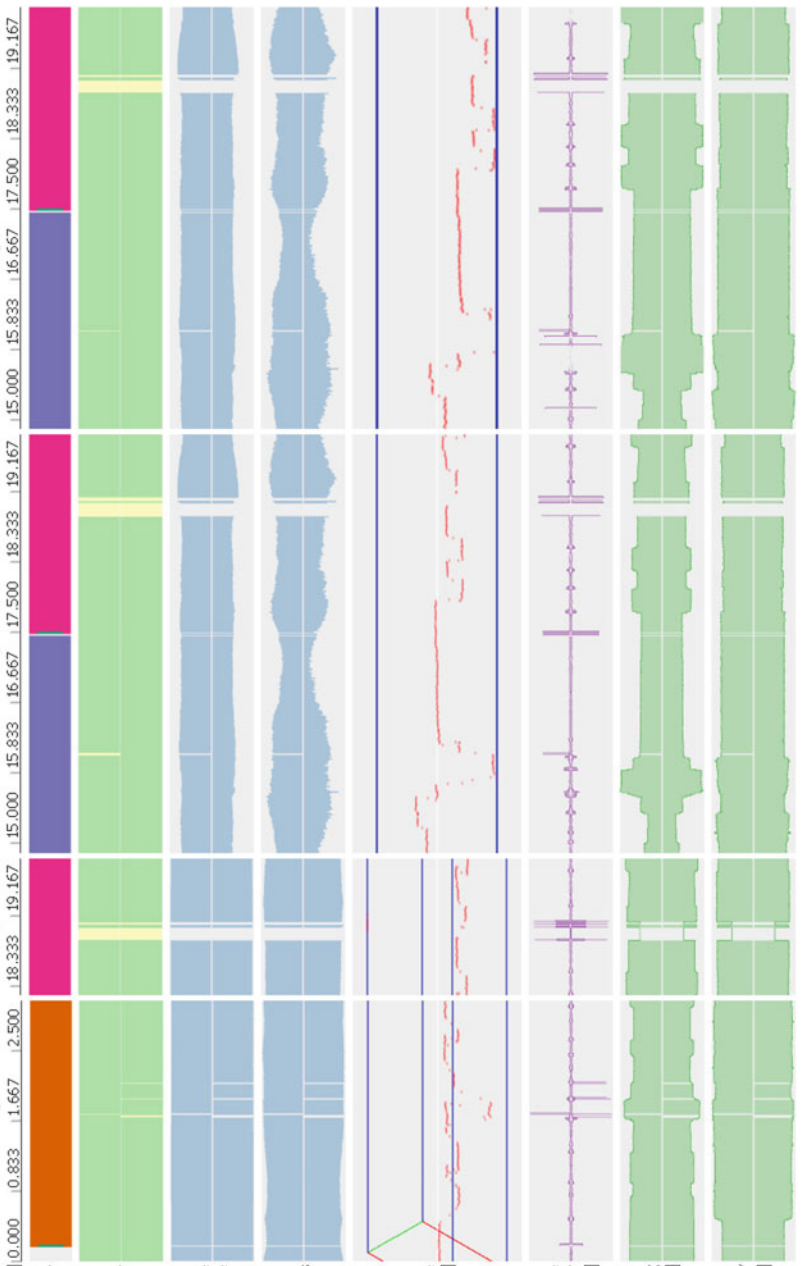


Fig. 8 Four slices from our first data set. From top to bottom: time, activity, validity, camera–eye distance, pupil size, gaze coordinates, gaze velocity, and decomposed gaze coordinates

data in an analysis, it might be a good idea to reconstruct the missing individual eye data, but for this data set it does not matter.

Slice Ⓑ reveals a real issue, as recognition of both eyes failed around 19 s. We can safely assume that the eye tracker does not use simple interpolation to fill in missing data, since the small time segment between the recognition errors does not seem to be a supporting point. After inspecting actual values, it turns out that the eye tracker emits zero for combined eye data and a negative magic value for individual eye data. This is documented behavior, even though it is considered bad practice to encode undefined and other magic values in $\mathbb{R}$. In addition, notice how camera–eye distance plot and pupil size plot seem featureless, because of the same issue. We fix this issue by filtering the corresponding data, i.e., setting magic values to undefined.

Slice Ⓒ shows the same data with cleansing functions applied. The pupil size plot and camera–eye distance plot now reveal many more features because scaling is no longer influenced by magic values. While rotating the space-time cube and correlating data with the stimulus image (not shown), we notice a weird offset in the data around 5.7 s. After inspection of the actual values, we know that the participant was looking off-screen. We decide to clamp the individual eye data to the screen resolution and set the combined eye data to undefined, if the participant was looking off-screen, so it will not interfere with post-cleansing analysis.

Slice Ⓓ shows the same data with more cleansing functions applied. The missing line on top of the lower two stereo line plots indicates that the participant was looking off-screen. Further inspection of the data reveals that this happens a number of times during comparative eye movements, indicating that this could be an issue resulting from the combination of the stimuli and eye tracker. Just to make sure we removed the data in question using a filter-and-drop function (not shown).

An interesting observation across slices Ⓒ and Ⓓ is that the participant's pupil sizes seem to be unsynchronized. Unless this is caused by the eye tracker, it might be worth investigating.

We finish cleansing of this participant's data set by comparing the space-time cube's pre- and post-cleansing, shown in Fig. 9. This also gives us a first impression of where fixations are (areas of high density).

The first data set is about 45–90 s per participant. It took a few of minutes to cleanse data from the first participant. Data of other participants was much faster to cleanse.

The second data set is about 15 min per participant. In addition to that, the eye tracker has a high sampling rate, hence we choose a different strategy: take a participant's data with minor modifications, or not. We demonstrate this strategy by visually inspecting about 4 min (one task) from one participant, shown in Fig. 10. We can confirm that the participant was reading instructions between Ⓐ and Ⓑ, because the *start* button is in the center of the screen and a line-reading pattern is revealed though density. This can be easily confirmed using the video plane. Around Ⓒ, the behavior changes much, the latency peaks, and the eye tracker starts to drop

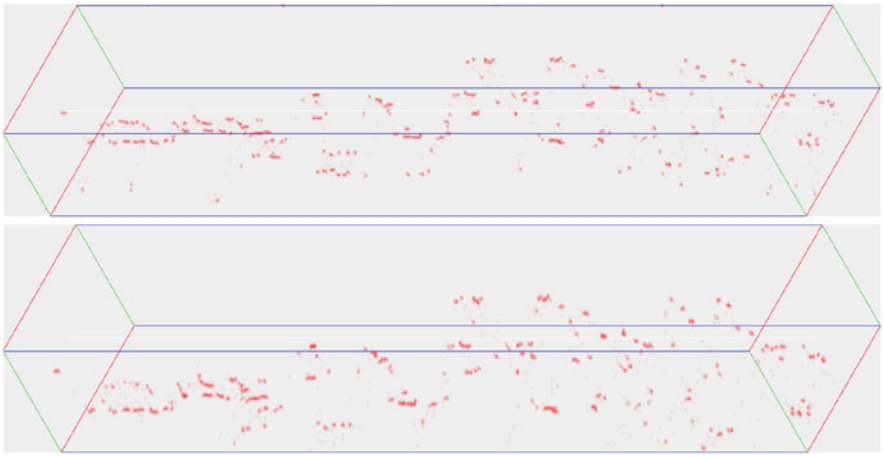


Fig. 9 Non-cleansed (*top*) and cleansed (*bottom*) space-time cubes from our first data set. Note dropped gazes close to the boundaries, where the participant was looking off-screen

data. Using the video plane, we can confirm that the participant has finished the task Again, through inspection of the video plane, the participant has finished the task, hence interesting eye movement data is between Ⓑ and Ⓒ. At Ⓓ, a latency peak in the region of interest and missing data can be observed. Through inspection of the video plane and raw values, the source of this issue is unclear. Because it did not interfere much with the experiment, we drop the affected samples. Around Ⓔ, recognition of the right eye seems odd, but gaze positions look fine, so we decide to ignore it. Finally, we cut the region of interest from the data set. This process took less than a minute to complete.

9 Conclusion and Future Work

We discussed a two-layered uncertainty model in the context of eye tracking and a processing model for data cleansing. Additionally, we presented a technique to visually deduce disagreement and credibility by comparing time-aligned, stacked representations of eye-tracking data. In particular, comparing the left and right eye against the combined eye seems to be a good strategy. Results show that data cleansing can be a fast process, given that enough visual context is provided, such as raw values, a stimulus view, and measured data augmented by uncertainty. We have increased data quality by dropping and clamping unexpected samples. Additionally, we provide an open source implementation for the interested reader.

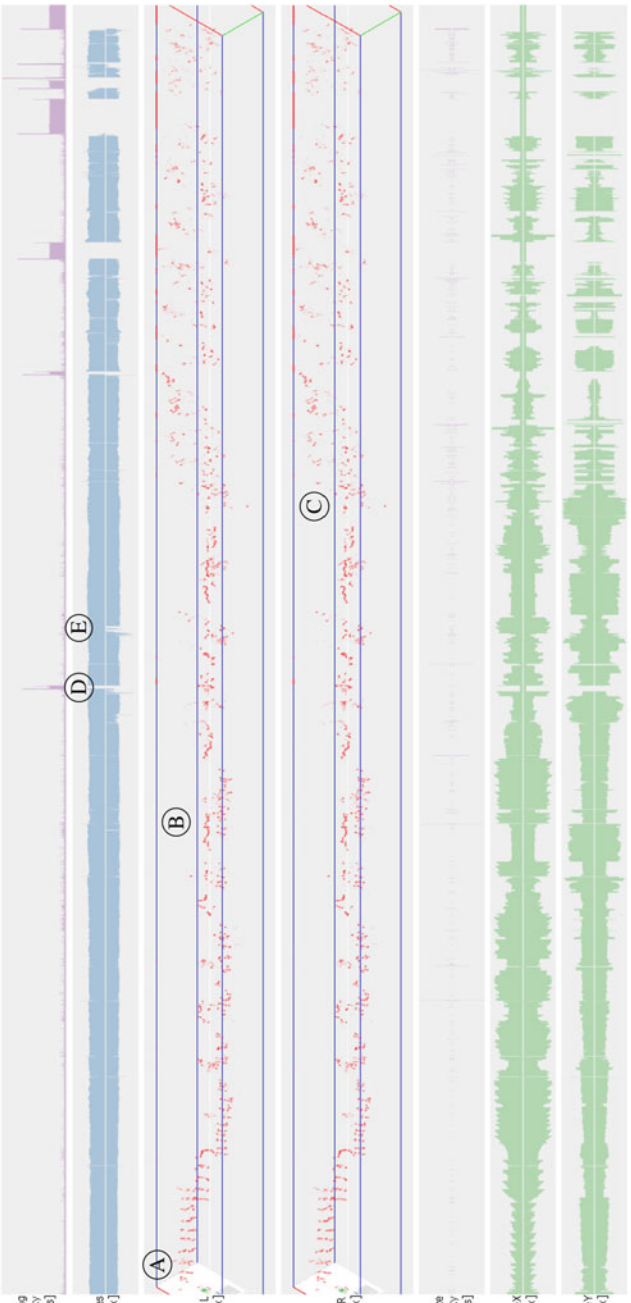


Fig. 10 A high-level view from our second data set, one participant. From top to down: processing latency, pupil sizes, gazes of the left eye, gazes of the right eye, along with gaze velocity and decomposed gaze coordinates of both eyes

In future work, want to extend our density-based approach to areas of interest and fixation filters. We believe that fuzzy intersection with areas of interest might be useful to convey and increase trust into data during analysis. From our own experience, fixation filters are sometimes difficult to tune, hence we envision sensitivity analysis for fixation filters, by mapping the parameter space of fixation filters to gaze space using volume rendering.

Acknowledgements We would like to thank the German Research Foundation (DFG) for financial support within project A01 of SFB/Transregio 161.

References

1. Ahrens, J., Geveci, B., Law, C.: ParaView: an end-user tool for large data visualization. Energy **836**, 717–732 (2005)
2. Aigner, W., Miksch, S., Müller, W., Schumann, H., Tominski, C.: Visualizing time-oriented data—a systematic view. Comput. Graph. **31**(3), 401–409 (2007)
3. Al-Rahayfeh, A., Faezipour, M.: Eye tracking and head movement detection: a state-of-art survey. Transl. Eng. Health Med. **1** (2013). http://ieeexplore.ieee.org/document/6656866/
4. Andersson, R., Nyström, M., Holmqvist, K.: Sampling frequency and eye-tracking measures: how speed affects durations, latencies, and more. J. Eye Mov. Res. **3**(3), 1–12 (2010)
5. Barz, M., Bulling, A., Daiber, F.: Computational modelling and prediction of gaze estimation error for head-mounted eye trackers (2015). https://www.d2.mpi-inf.mpg.de/content/computational-modelling-and-prediction-gaze-estimation-error-head-mounted-eye-trackers
6. Bavoil, L., Callahan, S.P., Crossno, P.J., Freire, J., Scheidegger, C.E., Silva, T., Vo, H.T.: VisTrails: enabling interactive multiple-view visualizations. In: Proceedings of IEEE Visualization, pp. 135–142 (2005)
7. Beard, K., Deese, H., Pettigrew, N.R.: A framework for visualization and exploration of events. Inf. Vis. **7**(2), 133–151 (2007)
8. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: State-of-the-art of visualization for eye tracking data. In: EuroVis STAR, pp. 63–82 (2014)
9. Blascheck, T., John, M., Koch, S., Kurzhals, K., Ertl, T.: VA^2: a visual analytics approach for evaluating visual analytics applications. IEEE Trans. Vis. Comput. Graph. **22**(1), 61–70 (2016)
10. Bojko, A.: Informative or misleading? Heatmaps deconstructed. In: Jacko, J. (ed.) Human-Computer Interaction. New Trends. Lecture Notes in Computer Science, vol. 5610, pp. 30–39. Springer, Berlin/Heidelberg (2009)
11. Brodlie, K., Allendes Osorio, R., Lopes, A.: A review of uncertainty in data visualization. In: Dill, J., Earnshaw, R., Kasik, D., Vince, J., Wong, P.C. (eds.) Expanding the Frontiers of Visual Analytics and Visualization, pp. 81–109. Springer, London (2012)
12. Cerrolaza, J.J., Villanueva, A., Villanueva, M., Cabeza, R.: Error characterization and compensation in eye tracking systems. In: Proceedings of the Symposium on Eye Tracking Research and Applications, ETRA'12, pp. 205–208 (2012)
13. Çöltekin, A., Fabrikant, S., Lacayo, M.: Exploring the efficiency of users' visual analytics strategies based on sequence analysis of eye movement recordings. Int. J. Geogr. Inf. Sci. **24**(10), 1559–1575 (2010)
14. Djurcilov, S., Kim, K., Lermusiaux, P.F.J., Pang, A.: Volume rendering data with uncertainty information. In: Proceedings of the Joint EUROGRAPHICS and IEEE TCVG Symposium on Visualization, pp. 243–252 (2001)

15. Gschwandtner, T., Aigner, W., Kaiser, K., Miksch, S., Seyfang, A.: CareCruiser: exploring and visualizing plans, events, and effects interactively. In: Proceedings of IEEE Pacific Visualization Symposium, PacificVis, pp. 43–50 (2011)
16. Gschwandtner, T., Aigner, W., Miksch, S., Gärtner, J., Kriglstein, S., Pohl, M., Suchy, N.: TimeCleanser: a visual analytics approach for data cleansing of time-oriented data. In: Proceedings of the 14th International Conference on Knowledge Technologies and Data-driven Business, i-KNOW'14, pp. 18:1–18:8 (2014)
17. Harrower, M., Brewer, C.A.: ColorBrewer.org: an online tool for selecting colour schemes for maps. Cartogr. J. **40**(1), 27–37 (2003)
18. Holmqvist, K., Nyström, M., Mulvey, F.: Eye tracker data quality: what it is and how to measure it. In: Proceedings of the Symposium on Eye Tracking Research and Applications, ETRA'12, pp. 45–52 (2012)
19. Hopf, M., Luttenberger Michael, M., Thomas, E.: Hierarchical splatting of scattered 4D data. IEEE Comput. Graph. Appl. **24**(4), 64–72 (2004)
20. Kandel, S., Paepcke, A., Hellerstein, J., Heer, J.: Wrangler: interactive visual specification of data transformation scripts. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 3363–3372 (2011)
21. Kurzhals, K., Weiskopf, D.: Space-time visual analytics of eye-tracking data for dynamic stimuli. IEEE Trans. Vis. Comput. Graph. **19**(12), 2129–2138 (2013)
22. Mackworth, J.F., Mackworth, N.H.: Eye fixations recorded on changing visual scenes by the television eye-marker. J. Opt. Soc. Am. **48**(7), 439–445 (1958)
23. Netzel, R., Burch, M., Weiskopf, D.: Interactive scanpath-oriented annotation of fixations. In: Proceedings of the Symposium on Eye Tracking Research and Applications, ETRA'16, pp. 183–187 (2016)
24. Rahm, E., Do, H.H.: Data cleaning: Problems and current approaches. IEEE Data Eng. Bull. **23**(4), 3–13 (2000)
25. Richardson, D.C., Dale, R.: Looking to understand: the coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. Cognit. Sci. **29**(6), 1045–1060 (2005)
26. Rind, A., Aigner, W., Miksch, S., Wiltner, S., Pohl, M., Turic, T., Drexler, F.: Visual exploration of time-oriented patient data for chronic diseases: design study and evaluation. In: Information Quality in e-Health. Lecture Notes in Computer Science, vol. 7058, pp. 301–320. Springer, Berlin/New York (2011)
27. SensoMotoric Instruments GmbH: BeGaze 2.4 Manual (2010)
28. Shahar, Y., Goren-Bar, D., Boaz, D., Tahan, G.: Distributed, intelligent, interactive visualization and exploration of time-oriented clinical data and their abstractions. Artif. Intell. Med. **38**(2), 115–135 (2006)
29. Singh, H., Singh, J.: Human eye tracking and related issues: a review. Int. J. Sci. Res. Publ. **2**, 1–9 (2012)
30. Skeels, M., Lee, B., Smith, G., Robertson, G.G.: Revealing uncertainty for information visualization. Inf. Vis. **9**(1), 70–81 (2010)
31. Stegmaier, S., Strengert, M., Klein, T., Ertl, T.: A simple and flexible volume rendering framework for graphics-hardware-based raycasting. In: Proceedings of the Fourth Eurographics/IEEE VGTC Conference on Volume Graphics, VG'05, pp. 187–195 (2005)
32. Tobii Technology: Tobii Studio 2.2 User Manual (2010)
33. Tobii Technology: Accuracy and Precision Test Report: Tobii T60 Eye Tracker (2011). 21 July 2011, Version: 2.1.1

Visualizing Dynamic Ambient/Focal Attention with Coefficient $\mathscr{K}$

A.T. Duchowski and K. Krejtz

Abstract Using coefficient $\mathscr{K}$, defined on a parametric scale, derived from processing a traditionally eye-tracked time course of eye movements, we propose a straightforward method of visualizing ambient/focal fixations in both scanpath and heatmap visualizations. The $\mathscr{K}$ coefficient indicates the difference of fixation duration and following saccade amplitude expressed in standard deviation units, facilitating parametric statistical testing. Positive and negative ordinates of $\mathscr{K}$ indicate *focal* or *ambient* fixations, respectively, and are colored by luminance variation depicting relative intensity of focal fixation.

1 Introduction

Visualization plays an increasingly important role in eye-tracking analysis. In their EuroVis state-of-the-art (STAR) report, Blascheck et al. [2] review and classify visualization techniques for eye movement data into three categories: point-based, Area-Of-Interest (AOI) -based, and those using both. They further distinguish between animated and static, 2D and 3D, in-context and not in-context, as well as interactive and non-interactive visualizations. Finally, visualization techniques are classified as either temporal, spatial, or spatio-temporal.

In this paper we propose novel (point-based, static, 2D, in-context, spatio-temporal) scanpath and heatmap visualizations of fixations via color mapping between ambient and focal fixations.

Ambient fixations are characterized by short duration fixations followed by long saccades, while focal fixations are composed of longer duration fixations followed by shorter saccades [26]. The combination of fixation duration and saccadic amplitude immediately following the fixation is thus the basis for the visualization developed below.

A.T. Duchowski (✉)
Clemson University, Clemson, SC, USA
e-mail: duchowski@clemson.edu

K. Krejtz
Department of Psychology, SWPS University of Social Sciences and Humanities, Warsaw, Poland
e-mail: kkrejtz@swps.edu.pl

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_13

Ambient/focal colorization of a scanpath preserves the traditional spatio-temporal characteristics of scanpath visualizations by conveying order of fixations and fixation durations. However, ambient/focal colorization introduces a novel form of visualization of the dynamic interplay between the focal and ambient modes of visual information processing. Generally, at early stages of scene perception, shorter fixations and longer saccades appear to govern initial scene exploration. Once a target has been identified, longer fixations are followed by shorter saccades suggesting a change to a focal mode of processing [9, 28].

The dynamic pattern of visual attention can be attributed to two modes of acquiring information: exploration and inspection. Pannasch et al. [19] showed a systematic increase in the durations of fixations and a decrease of saccadic amplitudes over the time course of scene perception. This relationship was very stable across the variety of studied conditions, including repeated presentation of similar stimuli, object density, emotional stimuli, and mood induction. In their work, the time courses of fixation durations and saccadic amplitudes were considered as two independent streams of data. We combine both streams into a single dynamic stream defined on a novel parametric scale capturing explicitly the interplay of ambient and focal modes (see below). The ambient/focal scale can then be applied to visualization of scanpaths or heatmaps.

2 Background

Blascheck et al. [2] review the state-of-the-art in scanpath visualizations. They note that for a typical scanpath visualization, each fixation is indicated by a circle (or disk), where the radius corresponds to the fixation duration. Saccades between fixations are represented by connecting lines between these circles. The connecting lines may include arrowheads and the fixation circles/disks may include numbers to indicate scanpath order.

Currently, most scanpath visualizations use a constant color to represent fixations. The color may change from scanpath to scanpath, when distinguishing between several individuals if more than one scanpath are composited, but the color does not usually change from fixation to fixation. Our scanpath visualization technique exploits this static choice of color and adjusts it at each fixation depending on where the fixation falls on the ambient/focal parametric scale. For heatmaps, the ambient/focal parameter is used to adjust the polarity of the Gaussian function deposited at each fixation.

Velichkovsky et al. [28] originally suggested characterization of fixations as focal or ambient based on their durations and the amplitude of successive saccades. However, visualizations related to the ambient/focal distinction were limited to graphs resembling histograms depicting either fixation duration or saccade amplitude as a function of viewing time (in 500 ms bins) or as saccade amplitude as a function of fixation duration (in 20 ms bins) [26]. It is important to note that these visualizations are meant to depict the distribution of saccade amplitudes and fixation durations

from which one can see that ambient/focal fixations occurred some time during the course of viewing, but not when.

Using the ambient/focal fixation distinction, Follet et al. [8] provide visualizations of the probability of occurrence of the type of fixation during the time course of viewing. From their visualizations one can see that, for example, ambient fixations are more likely to occur early in the viewing process, but not where.

Krejtz et al. [10] used an ambient/focal attention coefficient, defined as the relation between the current fixation duration and the subsequent saccade amplitude, but did not provide its derivation (see also Biele et al. [1]). Krejtz et al.'s [11] ambient/focal attention coefficient $\mathscr{K}_i$ transforms both fixation and saccade amplitudes into a standard score (z-score), allowing its computation per fixation (and in the aggregate per individual scanpath). Krejtz et al. [12] used the coefficient to analyze map viewing, unfortunately, visualization of the coefficient was never discussed by Krejtz et al. in any of their previous publications.

The coefficient $\mathscr{K}_i$ is calculated for each fixation as the difference between standardized values (z-scores) of the successive saccade amplitude (a_{i+1}) and the current ith fixation duration (d_i) [11]:

$$\mathscr{K}_i = \frac{d_i - \mu_d}{\sigma_d} - \frac{a_{i+1} - \mu_a}{\sigma_a}, \quad i \in [1, n-1] \tag{1}$$

where μ_d, μ_a are the mean fixation duration and saccade amplitude, respectively, and σ_d, σ_a are the fixation duration and saccade amplitude standard deviations, respectively, computed over all n fixations and hence n $\mathscr{K}_i$ coefficients (i.e., over the entire duration of the scanpath). Coefficient $\mathscr{K}_n$ takes on the value of $\mathscr{K}_{n-1}$ since there is no saccade at $n+1$ from which to compute $\mathscr{K}_n$.

Positive values of $\mathscr{K}_i$ show that relatively long fixations were followed by short saccade amplitudes, indicating focal processing. Analogously, negative values of $\mathscr{K}_i$ refer to the situation when relatively short fixations were followed by relatively long saccades, suggesting ambient processing.

For visualization purposes, the n $\mathscr{K}_i$ coefficients, each associated with the ith fixation, are normalized to facilitate colorization. Subsequently, a color map needs to be selected to produce pleasing and informative visualizations. Because $\mathscr{K}_i$ is associated with each ith fixation, colorization of the fixations depicts when ambient and focal fixations tend to occur and where (per individual scanpath). This visualization, based on the traditional scanpath, depicts the dynamics of ambient/focal attention spatio-temporally, at the expense of depicting statistical trends (e.g., as are possible via histogram-like graphs that lack either spatial or temporal information).

In the aggregate, i.e., rendering fixations from multiple participants, colorization of heatmaps with $\mathscr{K}$ adds information about visual attention dynamics, but for a group of viewers. Aggregate heatmap visualization may therefore help eye-tracking researchers distinguish viewing patterns on the ambient/focal scale, between different stimuli images or different participant groups. Although temporal information is missing in cumulative heatmap renderings, they serve as a basis for further

quantitative (statistical) analyses of viewing dynamic patterns. To our knowledge such a combination of information about fixation (or gaze) location and viewing dynamics has not yet been proposed for heatmap visualizations.

Apart from the novelty of the ambient/focal parametric scale itself, because the visualization technique mainly relies on a suitable choice of color palette, here we briefly only touch on what are likely appropriate color mapping selections.

The rainbow color map is the predominant choice for aggregate gaze visualization (e.g., for heat maps) although it is considered harmful because it [3]:

1. confuses viewers through its lack of perceptual ordering,
2. obscures data via uncontrolled luminance variation, and
3. misleads interpretation via introduction of non-data-dependent gradients.

In essence, the rainbow color map can introduce artificial boundaries in its representation. Ratwani et al. [23] show that the boundaries between red, yellow, green, and blue hues form "visual clusters" that serve as object-like units that can influence reasoning about the graph during cognitive integration. Coincidentally, they demonstrated the importance of these visual cluster boundaries empirically by recording fixations at these boundaries. They state that spectral (rainbow) color palettes should be used but only when differentiation between colors is desired. For gaze visualization, this is a key point, because it suggests the appropriateness of the rainbow color map but largely for *discrimination*, or identification, tasks. For *relative judgments*, Breslow et al. [4] make a compelling argument against the rainbow color map, advocating instead color maps based on brightness (luminance) scales.

Because our scanpath visualization relies on a continuous parametric scale, colorization via a spectral color palette unnecessarily transforms the scanpath into a visualization meant for identification of regions instead of one showing regions of relative magnitude.

The Python implementation of $\mathscr{K}$ visualization is not tied to any particular color map and allows selection from a variety of convenient choices. There are numerous single, dual, and multi-hue alternatives to the rainbow color map, including palettes from the Colorbrewer website [5].

3 Empirical Validation

To test different visualization color maps for $\mathscr{K}$, we used data from an experiment designed to replicate empirical procedures reported by Nothdurft [18]. Nothdurft's study showed that serial visual search largely relies on sequential shifts of focal attention whereas no such shifts occur during parallel search.

When performing serial search, therefore, more focal fixations are expected than during parallel search. Due to the pop-out effect during parallel search, fast localization of the target should yield a long saccade (large amplitude) directed to the target. Reaction times reported by Nothdurft support this supposition. They

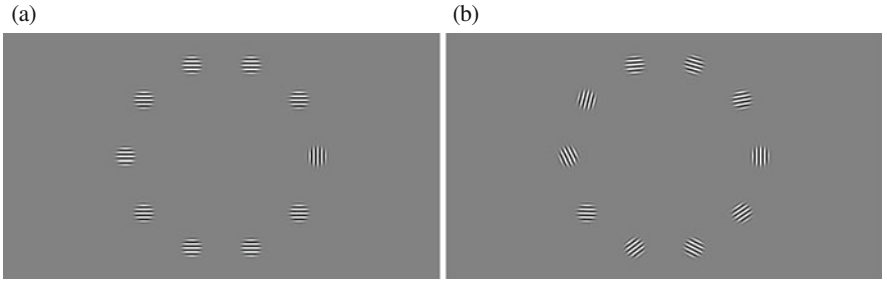
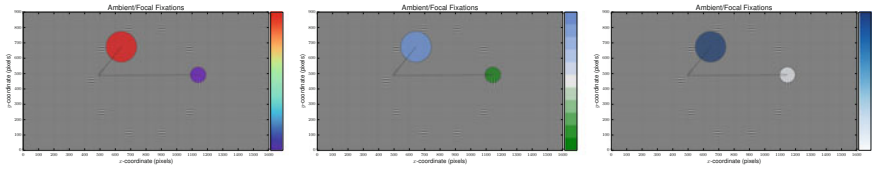
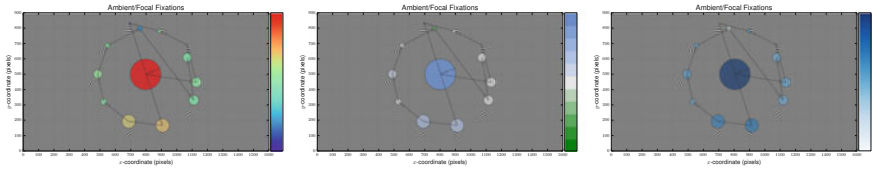


Fig. 1 Visual search stimulus, enhanced for visualization via automatic color balance. (**a**) Parallel search stimulus. (**b**) Serial search stimulus



(a) Ambient fixations: rainbow color map.
(b) Ambient fixations: divergence color map.
(c) Ambient fixations: sequential color map.



(d) Focal fixations: rainbow color map.
(e) Focal fixations: divergence color map.
(f) Focal fixations: sequential color map.

Fig. 2 Scanpath composed of focal and ambient fixations (**a**)–(**c**) and focal (**d**)–(**f**) fixations shown in different choices of color maps

are reminiscent of Treisman and Gelade's response times during *disjunctive* search [25]. The coefficient $\mathscr{K}$ should characterize these eye movements as ambient. Conversely, assuming serial search is composed of longer fixations followed by shorter saccades, $\mathscr{K}$ should reflect serial search as focal.

Replicating Nothdurft's experiment, we followed a within-subjects 2×2 factorial design, with two independent variables of search condition (serial vs. parallel) and target presence (hit vs. rejection). Each participant's task was to find a vertically oriented Gabor patch (the target) within a ring of nine distractor Gabor patches. The distractor patches were either all horizontally oriented (eliciting parallel search) or oriented randomly (eliciting serial search). Examples of stimuli are shown in Fig. 1, enhanced for visibility via automatic color balance. Actual stimuli were not enhanced in this way (see Fig. 2). In both examples, the target is present at 3o'clock.

For brevity we omit details of empirical results and focus on the visualization of $\mathscr{K}$ during searches where the target was present. Results of the experiment are described in detail elsewhere [11].

3.1 Color Map Selection

Analysis bears out the existence of expected types of ambient and focal fixations. Most of the recorded data shows scanpaths that begin with a first focal fixation. This is due to all participants starting by looking at a central fixation point, as per the experimental protocol. The stimulus appeared after a short delay. During parallel search (see Fig. 2a–c), participants usually made one large saccade to the intended target, as expected due to the pop-out effect. During serial search (see Fig. 2d–f), participants usually picked some random Gabor patch and then proceeded to serially inspect the ring of Gabor patches in either clockwise or counter-clockwise order. Figure 2 shows two scanpaths recorded from one participant that is representative of the data set. Scanpaths are colored with three different color maps.

Although the rainbow color map is known to be ineffective, it is nevertheless pervasive, especially in eye-tracking visualizations of heat maps. We use the traditional rainbow color map to test the rainbow map's propensity for distinguishing ambient fixations from focal. Figure 2a, d depict fixations during parallel and serial search, respectively. While the color map depicts a visual difference between $\mathscr{K}$ of focal and ambient fixations during serial search (Fig. 2d), the color hues that were drawn for depicting the focal and ambient fixation during parallel search (Fig. 2a) do not adequately convey the semantic distance between focal and ambient fixations—why should focal fixations be red and ambient ones dark purple? Neither hue is particularly semantically-resonant (see Lin et al. [14] for a discussion of semantically-resonant colors).

In their effective composition of luminance- and chrominance-based divergent color map, Rogowitz and Lloyd [24] show how luminance can be used to depict magnitude (e.g., terrain elevation) and semantic meaning by splitting the color map (e.g., at sea level). Their choice of color map effectively shows increasing luminance with terrain elevation, with landmass (green) demarcated from water (blue). In our case, a similar argument can be made: $\mathscr{K}$ magnitude is continuous, but there is also a demarcation between focal and ambient fixations at $\mathscr{K} = 0$. To implement this type of color map, we use two different color maps, blue for focal fixations ($\mathscr{K} > 0$) and green for ambient fixations ($\mathscr{K} < 0$). Figure 2b and e demonstrate the divergent color map. Although ambient and focal fixations are shown in an adequately dyadic manner, the viewer is required to remember which color depicts which type of fixation: are focal fixations blue or green? (They are blue.)

Unlike Rogowitz and Lloyd's terrain visualization, which itself carried semantic information (e.g., easily recognizable map of the U.S.), scanpaths do not inherently carry meaning. That is, to use Blascheck et al.'s [2] terminology, they are not in-context unless drawn overlaid atop the stimulus, which, in turn, may not necessarily

suggest an inherent viewing order. As such, without any prior expectation as to where or when ambient or focal fixations are expected, specifying divergent colors for their depiction leads to a visual ambiguity.

Our final choice of color map is motivated by visual clarity as well as convenience. Instead of specifying a custom color map for every data set, we would rather just select an appropriate color map from the bevy of choices available in Python's `matplotlib`. The most intuitive choice, based on luminance scaling, is the class of sequential color maps. Figure 2c, f show the effect of the sequential (orange in this case) color map. Instead of associating a color with ambient or focal type of fixation, lightness of hue indicates intensity of the fixation: darker hues suggest more intense (focal) viewing. Figure 2c clearly shows an ambient type of fixation following the initial central focal fixation. In contrast, Fig. 2f shows relatively darker (more focal) fixations proceeding sequentially along distractor patches until the intended target is fixated.

4 Application

Figure 3 shows the selection of a sequential color map in orange hues, used to visually distinguish visual inspection of Chest X-Ray (CXR) images as viewed by experts and novices. Radiologists employ a partially endogenous, cognitive

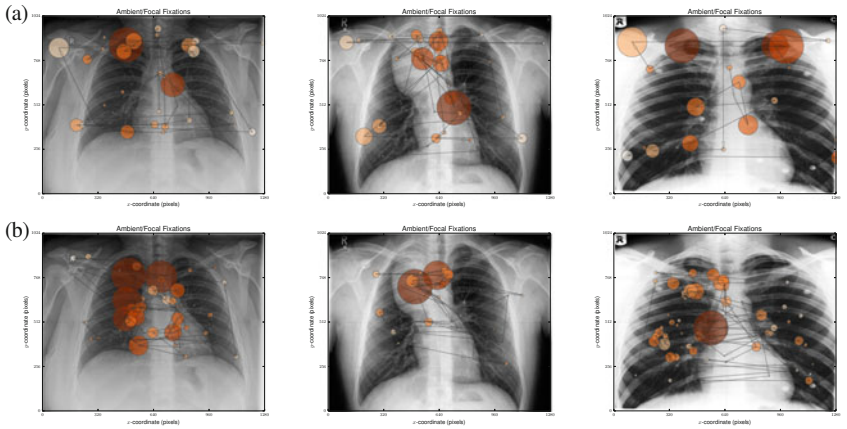


Fig. 3 Example of expert/novice scanpaths over Chest X-Ray (CXR) film. CXR images in the *middle column* feature an anterior mediastinal mass found at about pixel position (635, 768). Images in the *right column* feature an apical pneumothorax at about pixel position (650, 510). Experts tend to execute the visual inspection task considerably faster than novices, with novices tending to dwell longer over abnormalities, if any. Ambient/focal fixation visualization shows a greater preponderance of experts allocating ambient (*lighter*) fixations in peripheral image regions. (**a**) Expert reading CXR images: normal (*left*) and abnormal (*middle*: ant. mediastinal mass, *right*: apical pneumothorax). (**b**) Novice reading CXR images: normal (*left*) and abnormal (*middle*: ant. mediastinal mass, *right*: apical pneumothorax)

visual inspection strategy, related to top-down mechanisms that are based on prior expectations [15], which in turn are couched in training and experience. In the specific case of CXR reading, this strategy may be typified by the ABCDEFGHI mnemonic [29].

The ABCDEFGHI mnemonic guides viewers through a series of checks and assessments to inspect **A**irway, **B**ones, **C**ardiac silhouette, **D**iaphragms, **E**xternal soft tissues, **F**ields of the lungs, **G**astric bubble, **H**ila, and **I**nstrumentation. Figure 3 illustrates qualitatively the differences in expert and novice visual strategies: the expert executes the inspection quickly, tending to "check off" the ABCDEFGHI elements, not pausing excessively on any particular element. Visualization of $\mathscr{K}$ readily depicts this strategy, especially in the peripheral image regions (e.g., when inspecting bones, diaphragm). Conversely, the novice tends to dwell longer on each of the elements, often revisiting previously examined regions of the film. An "outside-in" ambient-to-focal strategy is thus not as clearly depicted as it is for the expert.

5 Aggregate Visualization

Visualizing fixations in the aggregate is usually achieved through the use of heatmaps. Unlike scanpaths, heatmaps provide a depiction of aggregate gaze by combining fixations from multiple viewers while sacrificing temporal information [6].

The heatmap can be thought of as an extension of the histogram, with accumulation of viewing count recorded at each pixel, but instead of discrete bins of data, each bin is represented by a Gaussian "peak" (or "valley", depending on polarity, see below). Heatmaps are also therefore described as Gaussian Mixture Models, or GMMs. The Gaussian functions modeled at each bin (e.g., fixation at its 2D coordinates) results in a blended visualization that is a smooth height map (Gaussian surface) of relatively weighted pixels. Colorized representations vary, with one of the more basic renderings obtained by mapping the height information directly to the alpha channel, resulting in a transparency map. Another popular colorization of the normalized height map features the pervasive rainbow color palette although here sequential or divergent color palettes can also be used to good effect.

The heatmap, or attentional landscape, was introduced by Pomplun et al. [21], and popularized by Wooding [30] to represent fixation maps (both were predated by Nodine et al.'s [17] "hotspots" rendered as bar-graphs). Other similar approaches involve gaze represented as height maps [7, 27] or Gaussian Mixture Models [16].

Heatmaps are generated by accumulating exponentially decaying intensity $I(i,j)$ at pixel coordinates (i,j) relative to a fixation at coordinates (x,y),

$$I(i,j) = \exp\left(-((x-i)^2 + (y-j)^2)/(2\sigma^2)\right) \tag{2}$$

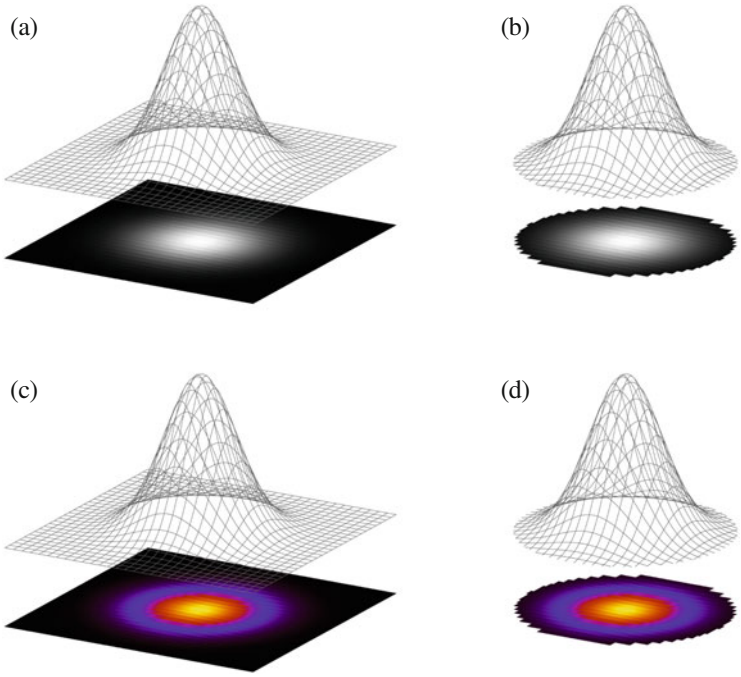


Fig. 4 The Gaussian point spread function as the heatmap's basic component with two different colorizations: basic greyscale and rainbow palette. Truncation past 2σ speeds up rendering but produces blocky artifacts. (**a**) Gaussian over single point. (**b**) Truncated gaussian. (**c**) Gaussian with rainbow palette. (**d**) Truncated rainbow Gaussian

where the exponential decay is modeled by the Gaussian point spread function (PSF). Figure 4 depicts a single Gaussian function aligned over one fixation, along with a luminance-based or rainbow color palette.

For smooth rendering, Gaussian kernel support should extend to image borders, requiring $O(n^2)$ iterations over an $n \times n$ image for *each* fixation, see Fig. 4a. For m fixations, an $O(mn^2)$ algorithm emerges. Following accumulation of intensities, the resultant heatmap must be normalized ($O(n^2)$) prior to colorization ($O(n^2)$). Duchowski et al. [6] rewrite the algorithm for the GPU, preserving the high image quality of extended-support Gaussian kernels while decreasing computation speed through parallelization. With the exception of maximum intensity localization for normalization, with enough GPU cores, each $O(n^2)$ is essentially replaced by an $O(1)$ computation performed simultaneously over all pixels, since each pixel "colors itself" evaluating (2) at $I(i, j)$ by finding the pixel's distance to the Gaussian's center. This process is repeated for each fixation, requiring $O(m)$ operations.

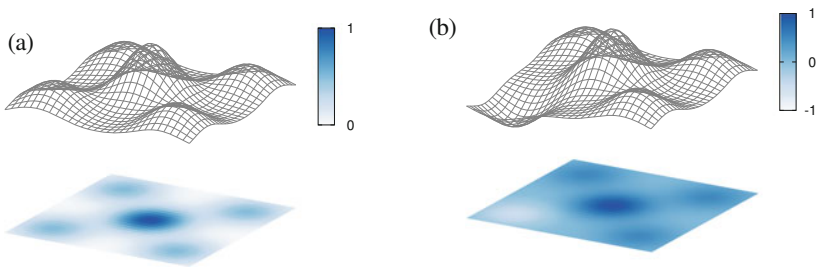


Fig. 5 Mixing Gaussian point spread functions to produce the ambient/focal heatmap. Ambient intensity is modeled by negating the Gaussian function, producing a valley instead of a peak. (**a**) Mixing multiple Gaussian functions. (**b**) Negating one Gaussian to produce ambient intensity

Implementing on a CPU, a slight speedup can be achieved by truncating the Gaussian kernel beyond 2σ [20]. Such an approach procures speed at the expense of blocky image artifacts. (see Fig. 4b).

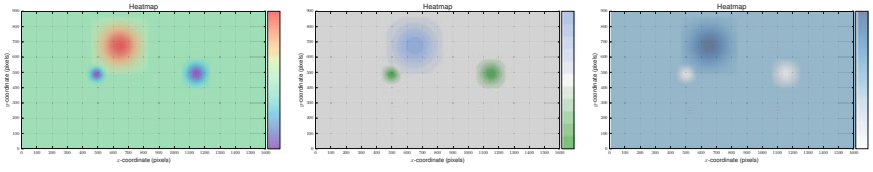
To produce ambient/focal visualization, ambient fixations must be visually distinguished from focal fixations. In the case of scanpaths, each fixation's color was mapped directly to the normalized sequential color palette. To achieve a similar effect with heatmaps, the heatmap must be extended such that locations corresponding to ambient fixations are made to subtract from the mean surface level, i.e., each Gaussian kernel's polarity (up or down) is determined by $\mathscr{K}_i$, using the sign of $\mathscr{K}_i$ to affect the kernel's direction,

$$I(i,j) = \text{sgn}\,(\mathscr{K}_i)\exp\left(-((x-i)^2 + (y-j)^2)/(2\sigma^2)\right) \tag{3}$$

Note that with this construct, it is possible that overlapping of fixations at the same location but with exactly opposite polar magnitudes, i.e., one Gaussian kernel pointing up the other down, would result in a flat surface at that location. That is, the two fixations would neutralize each other. In practice, however, fixations rarely overlap precisely, and so we expect this equal but opposite perfect alignment to be rare (Fig. 5).

5.1 Empirical Validation Revisited

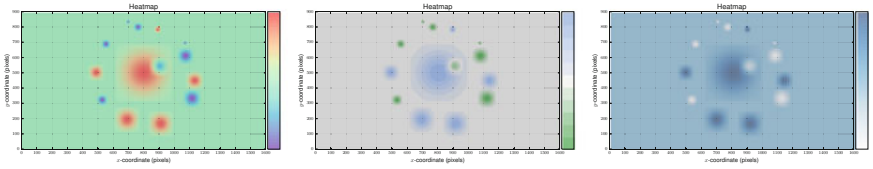
A similar coloring problem is encountered with the heatmap as with scanpaths. Using the same individual data as depicted in Fig. 2, and using (3), corresponding heatmaps are shown in Fig. 6. As with scanpaths, our opinion is that neither the rainbow nor the divergence color maps are appropriate, and for similar reasons,



(a) Ambient fixations: rainbow color map.

(b) Ambient fixations: divergence color map.

(c) Ambient fixations: sequential color map.



(d) Focal fixations: rainbow color map.

(e) Focal fixations: divergence color map.

(f) Focal fixations: sequential color map.

Fig. 6 Heatmaps composed of mainly ambient (**a**)–(**c**) or focal (**d**)–(**f**) fixations shown in different choices of color maps

although some might prefer the divergent color map. Meanwhile, the sequential color map shows similar utility as for scanpaths, so long as color saturation is taken to indicate focal intensity: more focal fixations are more deeply saturated while more ambient fixations are lighter in appearance (see Fig. 6c, f).

Visualizing individuals' ambient/focal fixations with a heatmap, however, only serves to validate the rendering approach. The heatmap visualization is itself a poor choice for depicting individual scanpaths since the order of fixations is lost. The real utility of the heatmap lies in its ability to depict fixations in the aggregate, from multiple viewers. Figure 7 shows such a visualization of all viewers conducting either the serial or parallel searches over stimuli where the target is present at 3o'clock. Figure 7a clearly shows a larger proportion of focal fixations performed during serial search than during parallel search shown in Fig. 7b. Divergent and rainbow colorized aggregate heatmaps are shown in Fig. 7e–h.

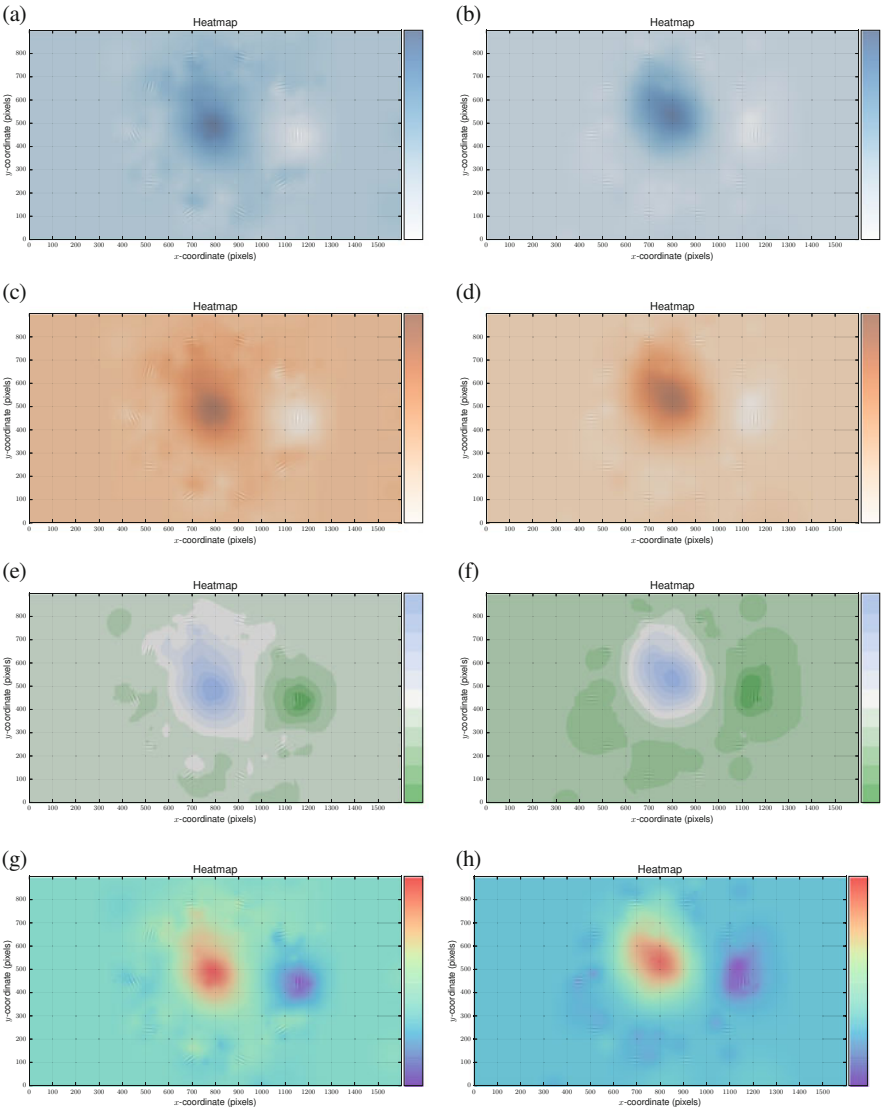


Fig. 7 Aggregate view of serial and parallel searches collected from all participants with target present at 3o'clock rendered with various color palettes. (**a**) Serial search: sequential blue palette (**b**) Parallel search: sequential blue palette (**c**) Serial search: sequential orange palette (**d**) Parallel search: sequential orange palette (**e**) Serial search: divergent color map (**f**) Parallel search: divergent color map (**g**) Serial search: rainbow color map (**h**) Parallel search: rainbow color map

6 User Study

A user study was performed to test the usability of aggregate heatmap colorizations. A Computer Assisted Web Interview, based on LimeSurvey [13], was devised following a monadic evaluation design. Participants (N = 10, 7 females and 3 male, mean age 33.8) were presented with the set of colorized heatmaps of ambient and focal eye movements with the following palettes: divergent, rainbow, sequential (Blue and Orange), see Fig. 7. The heatmaps were shown separately on a computer screen in random order and were immediately followed by four questions:

1. 'Evaluate whether this map represents ambient or focal eye movements',
2. 'Rate the difficulty of the evaluation',
3. 'Rate the difficulty of identifying gaze locations', and
4. 'Gauge the heatmap aesthetics'.

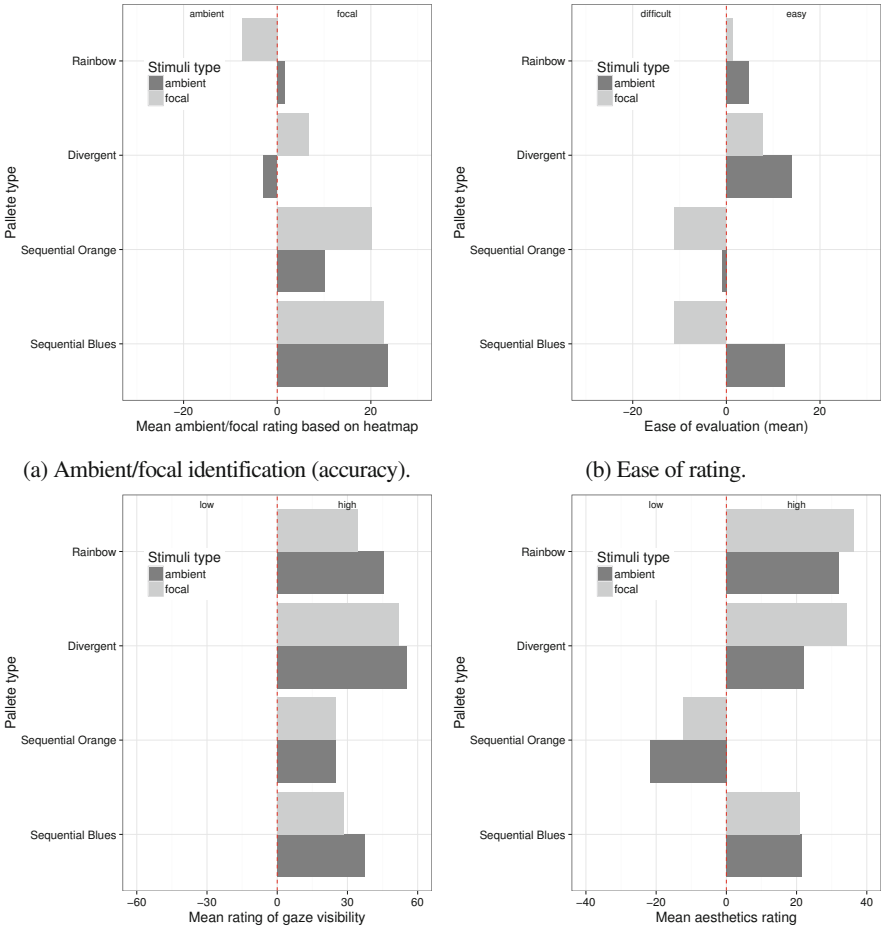
Responses were given on unmarked semantic differential scales (with range $[-100 : +100]$). All participants declared that they had seen heatmaps before. Statistics were based on a 4×2 two-way within-subjects ANOVA with palette type and type of eye movements (ambient vs. focal) as independent factors, followed by Tukey HSD pairwise comparisons when needed. Dependent variables were responses as well as timing on giving the response. Statistics were computed in R [22].

Analysis reveled a significant main effect of color palette on response timing, $F(3, 27) = 6.56, p < 0.001, \eta^2 = 0.17$. Evaluation of both sequential and divergent palettes was performed significantly faster than with the rainbow palette (all pairwise t-tests were significant $p < 0.05$). Descriptive statistics on response time for all palettes are as follows (in seconds): sequential Orange (M = 34.46, SE = 5.99), sequential Blue (M = 38.53, SE = 5.99), divergent (M = 41.81, SE = 5.99), and rainbow (M = 59.08, SE = 5.99).

Comparison of accuracy (correctly distinguishing ambient from focal) showed that participants evaluated heatmaps accurately only when colorized with the divergent palette. With both sequential palettes, participants overrated focal eye movements, while with the rainbow palette they evaluated incorrectly, see Fig. 8a.

Interestingly, analysis of subjective ease of evaluation showed no significant differences between heatmaps, $F(3.27) < 1$, see Fig. 8b. However, subjective evaluation of the ease of locating fixations on heatmaps showed a marginally significant difference between colorizations, $F(2, 27) = 2.61, p = 0.07, \eta^2 = 0.07$. The divergent coloring appeared to help gaze localization better than the sequential Orange heatmap, $t(27) = 2.69, p = 0.051$, see Fig. 8c.

Analysis of visual aesthetics revealed a main effect of palette, $F(2, 27) = 4.35, p = 0.05, \eta^2 = 0.13$. The sequential Orange heatmap was rated significantly lower compared to the divergent or rainbow heatmaps ($p < 0.05$). The Orange palette was also marginally less attractive than the sequential Blue palette ($p = 0.09$). The difference between rainbow and divergent heatmaps was not significant, see Fig. 8d.



(a) Ambient/focal identification (accuracy).

(b) Ease of rating.

(c) Ease of gaze localization.

(d) Aesthetic rating.

Fig. 8 Mean results of participants' evaluation of aggregate heatmap colorizations. (**a**) Ambient/focal identification (accuracy). (**b**) Ease of rating. (**c**) Ease of gaze localization. (**d**) Aesthetic rating

6.1 Discussion & Study Limitations

Both divergent and sequential color maps foster evaluation of ambient/focal fixations, with the divergent palette promoting greater speed and accuracy in disambiguation of ambient and focal fixations. Study participants could correctly identify aggregate focal and ambient fixations only with the divergent color map, although they did so more slowly than with the sequential color maps.

The rainbow color map required the longest time to respond and resulted in incorrect ambient/focal fixation labelling. The rainbow color map thus elicits the

worst performer in terms of both speed and accuracy, yet it was rated the highest in its aesthetic evaluation. This helps explain why it is on the one hand considered harmful by some in the visualization community [3], yet on the other hand endures in its popularity. Aesthetic ratings evince the rainbow palette's popularity even though it leads to incorrect identification of ambient/focal fixations. Only the divergent color map led to correct ambient/focal fixation identification, and was judged the easiest with which to do so. Unfortunately a potential confound exists in the smaller number of color levels used in expression of the divergent color palette, compared to the sequential and rainbow color palettes. The reason is due to the custom construction of the divergent color map as one was not available from the selection of palettes. That is, the divergent color map had to be constructed manually in order to align its central point with $\mathscr{K} = 0$. The resulting palette, coarser in its resolution of color, likely fosters easier identification of fixations due to the resultant discrete boundaries between colors. One could argue that the sequential and rainbow color maps lack these sharp color level boundaries and that with fewer color levels perhaps they could also lead to easier localization of fixations in either map. A future study would need to control for the number of color levels used in the color palettes.

7 Conclusions

We have presented a visualization of Krejtz et al.'s [11] $\mathscr{K}$ coefficient depicting the dynamic interplay between ambient and focal fixations. Visualization is straightforward, resulting from normalization of $\mathscr{K}$ followed by selection of appropriate color map.

In this chapter we developed two novel types of visualizations. The first is based on individual eye movement scanpaths, the second on aggregate fixation heatmaps. Both eye movement visualizations are enhanced via addition of the ambient/focal dimension. Importantly, this additional information does not compromise either scanpath or heatmap readability. With scanpaths, one can still easily visually determine the sequence of fixations as well as their relative durations and saccade amplitudes. With heatmaps, one can still easily find where most of the aggregate fixations were located and which elements of the visual stimuli were attended to or omitted.

The ambient/focal enhancement enriches both types of visualizations with important and unique information which helps in the visual inspection of recorded eye movements and facilitates drawing insights from observed visual behavior. First and foremost, scanpaths are endowed with information on the temporal dynamics of ambient/focal visual scanning. Moving between ambient and focal scanning is now more easily visible. For potential applied examples, consider the process of information foraging from complex web sites. Ambient/focal scanpaths may show not only when the target element of the website is fixated but also how visual search progressed to the target. Additionally, using information about the dynamics of visual search may enlighten potential localization of obstacles in webpage composition.

Although the temporal dynamics of eye movements are lost in the aggregate heatmap visualization, ambient/focal heatmaps may help to quickly differentiate between how stimuli were viewed or how groups of participants viewed the stimuli. This may be particularly relevant when considering complex visual stimuli, e.g., art images or advertisements. Regarding advertisements, one may quickly see whether or not the brand or main claim was fixated in the aggregate (a facility provided by any heatmap) but using $\mathscr{K}$ one could also see how the advertisement was viewed, i.e., carefully (with focal attention) or just generally scanned (with ambient attention). Such conclusions may help in formulating preliminary insights which could then serve to drive subsequent empirical exploration involving further experimentation or statistical analyses.

Acknowledgements We thank Dr. Helena Duchowska (MD, retired) for her help in reading the CXR images and pinpointing the anomalies contained therein.

This work was partially supported by a 2015 research grant "Influence of affect on visual attention dynamics during visual search" from the SWPS University of Social Sciences and Humanities.

References

1. Biele, C., Kopacz, A., Krejtz, K.: Shall we care about the user's feelings? Influence of affect and engagement on visual attention. In: Proceedings of the International Conference on Multimedia, Interaction, Design and Innovation, MIDI'13, pp. 7:1–7:8. ACM, New York (2013). doi:10.1145/2500342.2500349. http://doi.acm.org/10.1145/2500342.2500349
2. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: Start-of-the-art of visualization for eye tracking data. In: Borgo, R., Maciejewski, R., Viola, I. (eds.) EuroGraphics Conference on Visualization (EuroVis). EuroVis STAR—State of the Art Report (2014)
3. Borland, D., Taylor II, R.M.: Rainbow color map (still) considered harmful. IEEE Comput. Graph. Appl. **27**(2), 14–17 (2007)
4. Breslow, L.A., Ratwani, R.M., Trafton, J.G.: Cognitive models of the influence of color scale on data visualization tasks. Hum. Factors **51**(3), 321–338 (2009)
5. Brewer, C., Harrower, M., Woodruff, A., Heyman, D.: Colorbrewer 2.0: color advice for maps. Online Resource (2009). http://colorbrewer2.org. Last accessed Dec 2010
6. Duchowski, A.T., Price, M.M., Meyer, M., Orero, P.: Aggregate gaze visualization with real-time heatmaps. In: Proceedings of the Symposium on Eye Tracking Research and Applications, ETRA'12, pp. 13–20. ACM, New York (2012). doi:10.1145/2168556.2168558. http://doi.acm.org/10.1145/2168556.2168558
7. Elias, G., Sherwin, G., Wise, J.: Eye movements while viewing NTSC format television. Technical report, SMPTE Psychophysics Subcommittee (1984)
8. Follet, B., Le Meur, O., Baccino, T.: New insights on ambient and focal visual fixations using an automatic classification algorithm. i-Perception **2**(6), 592–610 (2011)
9. Irwin, D.E., Zelinsky, G.J.: Eye movements and scene perception: memory for things observed. Percept. Psychophys. **64**, 882–895 (2002)
10. Krejtz, I., Szarkowska, A., Krejtz, K., Walczak, A., Duchowski, A.: Audio description as an aural guide of children's visual attention: evidence from an eye-tracking study. In: ETRA'12: Proceedings of the 2012 Symposium on Eye Tracking Research & Applications, ETRA'12, pp. 99–106. ACM, New York (2012). doi:10.1145/2168556.2168572

11. Krejtz, K., Duchowski, A., Krejtz, I., Szarkowska, A., Kopacz, A.: Discerning ambient/focal attention with coefficient $\mathscr{K}$. Trans. Appl. Percept. **13**(3), Article 11 (2016). http://dx.doi.org/10.1145/2896452
12. Krejtz, K., Duchowski, A.T., Çöltekin, A.: High-level gaze metrics from map viewing: charting ambient/focal visual attention. In: Kiefer, P., Giannopoulos, I., Raubal, M., Krüger, A. (eds.) Proceedings of the 2nd International Workshop on Eye Tracking for Spatial Research (ET4S), Vienna (2014)
13. LimeSurvey Project Team/Carsten Schmitz: LimeSurvey: an open source survey tool. LimeSurvey Project, Hamburg (2012). http://www.limesurvey.org
14. Lin, S., Fortuna, J., Kulkarni, C., Stone, M., Heer, J.: Selecting semantically-resonant colors for data visualization. In: Proceedings of the 15th Eurographics Conference on Visualization, EuroVis'13, pp. 401–410. Eurographics/John Wiley, Chichester (2013). doi:10.1111/cgf.12127. http://dx.doi.org/10.1111/cgf.12127
15. Mello-Thoms, C., Nodine, C.F., Kundel, H.L.: What attracts the eye to the location of missed and reported breast cancers? In: ETRA'02: Proceedings of the 2002 Symposium on Eye Tracking Research & Applications, pp. 111–117. ACM, New York (2002). http://doi.acm.org/10.1145/507072.507095
16. Mital, P.K., Smith, T.J., Hill, R.L., Henderson, J.M.: Clustering of gaze during dynamic scene viewing is predicted by motion. Cogn. Comput. **3**, 5–24 (2011)
17. Nodine, C.F., Kundel, H.L., Toto, L.C., Krupinski, E.A.: Recording and analyzing eye-position data using a microcomputer workstation. Behav. Res. Methods **24**(3), 475–485 (1992)
18. Nothdurft, H.C.: Focal attention in visual search. Vis. Res. **39**, 2305–2310 (1999)
19. Pannasch, S., Helmert, J.R., Roth, K., Herbold, A.K., Walter, H.: Visual fixation durations and saccade amplitudes: shifting relationship in a variety of conditions. J. Eye Mov. Res. **2**(2), 1–19 (2008)
20. Paris, S., Durand, F.: A fast approximation of the bilateral filter using a signal processing approach. Technical report, MIT-CSAIL-TR-2006-073, Massachusetts Institute of Technology (2006)
21. Pomplun, M., Ritter, H., Velichkovsky, B.: Disambiguating complex visual information: towards communication of personal views of a scene. Perception **25**(8), 931–948 (1996)
22. R Development Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna (2011)
23. Ratwani, R.M., Trafton, J.G., Boehm-Davis, D.A.: Thinking graphically: connecting vision and cognition during graph comprehension. J. Exp. Psychol. Appl. **14**(1), 36–49 (2008)
24. Rogowitz, B., Treinish, L.: Data visualization: the end of the rainbow. IEEE Spectr. **35**(12), 52–59 (1998)
25. Treisman, A., Gelade, G.: A feature integration theory of attention. Cogn. Psychol. **12**, 97–136 (1980)
26. Unema, P.J.A., Pannasch, S., Joos, M., Velichkovsky, B.: Time course of information processing during scene perception. Vis. Cogn. **12**(3), 473–494 (2005)
27. van Gisbergen, M.S., van der Most, J., Aelen, P.: Visual attention to online search engine results. Technical report, De Vos & Jansen in cooperation with Checkit (2007). http://www.iprospect.nl/wp-content/themes/iprospect/pdf/checkit/eyetracking_research.pdf. Last accessed Dec 2011
28. Velichkovsky, B.M., Joos, M., Helmert, J.R., Pannasch, S.: Two visual systems and their eye movements: evidence from static and dynamic scene perception. In: CogSci 2005: Proceedings of the XXVII Conference of the Cognitive Science Society, Stresa, pp. 2283–2288 (2005)
29. Vitak, S.A., Ingram, J.E., Duchowski, A.T., Ellis, S., Gramopadhye, A.K.: Gaze-augmented think-aloud as an aid to learning. In: Proceedings of the SIGCHI Conference on Human Factors in computing systems, CHI'12, pp. 1253–1262. ACM, New York (2012). doi:http://doi.acm.org/10.1145/1124772.1124961. http://doi.acm.org/10.1145/1124772.1124961
30. Wooding, D.S.: Fixation maps: quantifying eye-movement traces. In: ETRA'02: Proceedings of the 2002 Symposium on Eye Tracking Research & Applications, pp. 31–36. ACM, New York (2002). doi:http://doi.acm.org/10.1145/507072.507078

Eye Fixation Metrics for Large Scale Evaluation and Comparison of Information Visualizations

Zoya Bylinskii, Michelle A. Borkin, Nam Wook Kim, Hanspeter Pfister, and Aude Oliva

Abstract An observer's eye movements are often informative about how the observer interacts with and processes a visual stimulus. Here, we are specifically interested in what eye movements reveal about how the content of information visualizations is processed. Conversely, by pooling over many observers' worth of eye movements, what can we learn about the general effectiveness of different visualizations and the underlying design principles employed? The contribution of this manuscript is to consider these questions at a large data scale, with thousands of eye fixations on hundreds of diverse information visualizations. We survey existing methods and metrics for collective eye movement analysis, and consider what each can tell us about the overall effectiveness of different information visualizations and designs at this large data scale.

1 Introduction

Eye movements can provide us with clues about the elements of a visual display that people pay attention to, what they spend most time on, and how they redirect their attention between elements. The eyes can also be used as indicators of higher-level cognitive processing like memory, comprehension, and problem solving [21, 23, 32, 39, 40, 54].

Z. Bylinskii (✉) • A. Oliva
Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology, 32 Vassar St., Boston, MA, USA
e-mail: zoya@mit.edu; oliva@mit.edu

M.A. Borkin
College of Computer and Information Science, Northeastern University, 360 Huntington Ave., Boston, MA, USA
e-mail: m.borkin@neu.edu

N.W. Kim • H. Pfister
School of Engineering & Applied Sciences, Harvard University, 33 Oxford Street, Boston, MA, USA
e-mail: namwkim@seas.harvard.edu; pfister@seas.harvard.edu

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5_14

Eye movement analyses have been used to study the perception of natural scenes, simple artificial stimuli, webpages, user interfaces, and increasingly, information visualizations. In human-computer interaction (HCI), eye tracking has often been used for evaluating the usability of systems and studying the related question of interface design [13, 19, 29, 47]. Duchowski provides a survey of different eye-tracking applications in domains ranging from industrial engineering to marketing [13].

In the visualization community, eye-tracking analyses have been used to independently evaluate different visualizations (e.g., graphs [25–27, 39, 49], node-link diagrams [1], tree diagrams [8], parallel coordinates [62]) and to directly compare visualization types [6, 11, 17]. Eye tracking has also been used to understand how a person visually perceives, explores, searches, and remembers a visualization, providing a window into the cognitive processes involved when interacting with visualizations [1, 3, 6, 11, 26, 37, 49, 50, 53].

Information visualizations are specifically designed to be parsed and understood by human observers. Visualizations can be created to help convey a specific message to a general audience, or to help data analysts extract trends and meaning from the data. As visualizations are amenable to specific tasks, observer performance on those tasks can be directly measured (e.g., ability to find a specific piece of information, to solve an analysis task, to remember the content for later retrieval, etc.). Eye movement analyses can then be used to provide possible explanations of task performance (e.g., why a task was completed quicker with one visualization design as compared to another), as complementary performance measurements that take into account human perception. Eye movements can provide a window into the cognitive processing taking place when an observer examines an information visualization.

Although different eye movement metrics have been previously reviewed within the context of different tasks [1, 17, 29, 51], in this manuscript we focus specifically on eye fixation metrics that can be used for *collective analysis* (the aggregation of data across a population of observers and visualizations) of information visualization designs. We provide a review of metrics that can be used for the *quantitative comparison* of different visualization designs in a large data setting. Unlike many previous studies, our analyses are broad, spanning a large diversity of visualization types and sources. We discuss and visualize ways in which different metrics can be used to evaluate the effectiveness of different visualization designs, and we use the MASSVIS dataset [6] to provide some specific examples. The review provided in this manuscript is intended to motivate further research into large-scale eye movement analysis for the broad comparison and evaluation of visualization designs.

2 Methods

2.1 Visualization Data

We used the MASSVIS dataset of 393 labeled target visualizations,[1] spanning four different **source categories**: government and world organizations, news media, infographics, and scientific publications [6]. These visualizations were manually labeled using the LabelMe system [59] and Borkin et al.'s visualization taxonomy [7] (Fig. 1a). Labels classify **visualization elements** as: data encoding, data-related components (e.g., axes, annotations, legends), textual elements (e.g., title, axis labels, paragraphs), pictograms or human recognizable objects, or graphical elements with no data encoding function. Labels can overlap in that a single region can have a number of labels (e.g., an annotation on a graph has an annotation label and a graph label). Labels are available for analyses as segmented polygons.

2.2 Eye-tracking Experiments

We used eye movements collected during the *encoding* experimental phase from the study by Borkin et al. [6]. During this phase, each visualization was shown to participants for 10 s, producing an average of 37.4 (SD: 3.2) eye fixations per visualization, or an average 623 (SD: 93) total fixations per visualization. This duration proved to be of sufficient length for a participant to read the visualization's title, axes, annotations, etc., as well as explore the data encoding, and short enough to avoid too much redundancy in fixation patterns and explorative strategies. Participants were told to remember as many details of each visualization as possible for subsequent experimental phases. During the *recognition* and *recall* phases, respectively, participants completed a memory task and were asked to write descriptions of the visualizations they remembered. We do not directly use this additional data in the present manuscript, but refer to the conclusions made from the eye movement analyses in the context of memory performances.

Eye movements of 33 participants were recorded on 393 target visualizations, with an average of 16.7 viewers (SD: 1.98) per visualization. Equipment included an SR Research EyeLink1000 desktop eye-tracker [63] with a chin-rest mount 22 in from a 19 in CRT monitor (1280 × 1024 pixels). For each eye fixation, available for analysis are its spatial location in pixel coordinates, duration in milliseconds, and ordering within the entire viewing episode (scanpath).

[1] Dataset available at http://massvis.mit.edu.

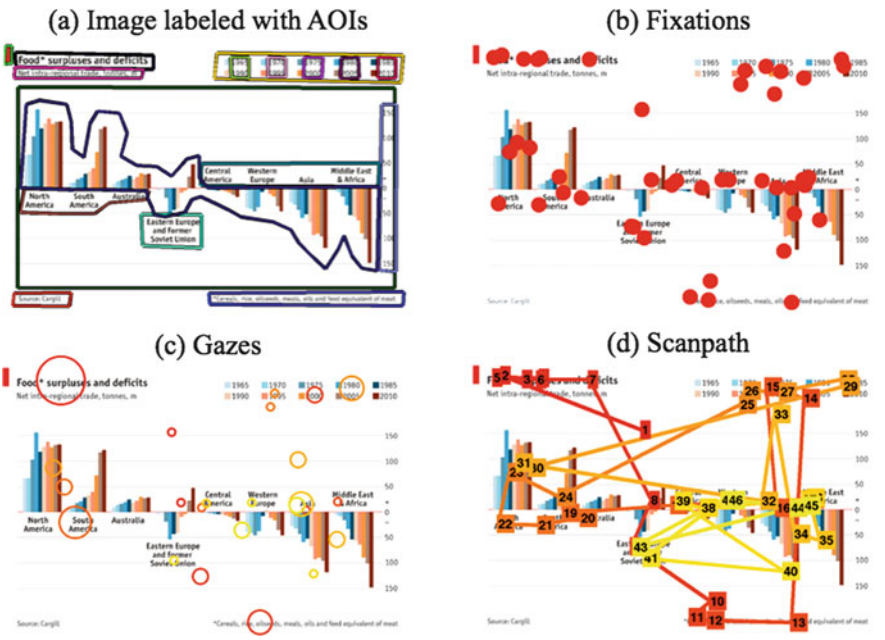


Fig. 1 We plot the fixations of a single observer for demonstration purposes, to visually depict a few key terms used throughout this manuscript. (**a**) The images we use are labeled with AOIs (Areas of Interest), which are elements like the title, axes, and legend. (**b**) Fixations are the discrete locations that an observer's eyes have landed on at some point during the viewing episode. (**c**) Multiple consecutive fixations that land on the same AOIs of an image can be further clustered into gazes. The size of the gaze marker is proportional to the number of fixations making up the gaze, with the marker centered at the mean of those fixation locations. (**d**) A scanpath is the sequence of fixations made. Here, to denote the temporal ordering, fixations are connected by *lines*, numerically labeled, and colored such that the earliest are in *red* and the latest in *yellow*

2.3 Metrics and Visualizations

Depending on the analysis being performed, different aspects of eye movement behavior can be measured including fixation locations, fixation durations, and saccades.[2] **Fixations** are discrete samples of where an eye was looking on a visual display obtained from continuous eye movement data[3] (Fig. 1b). By segmenting the visual stimulus into elements or Areas of Interest, denoted **AOI**, fixations falling on different AOIs can be separately analyzed (Fig. 1a). Consecutive fixations on a specific region or AOI can be further clustered into **gazes** (Fig. 1c).

[2]Saccades are intervals between fixations: the motion of the eyes from one fixation point to the next. The analysis of saccades is beyond the scope of the present manuscript, for which additional metrics would be necessary [40, 51].

[3]The eye has to be recorded as "still" according to prespecified parameters [24, 60]. We use the standard thresholds set by the EyeLink Eyetracker [63].

Apart from summarizing the number and duration of fixations on a visual design or its constituent elements, the spatial and sequential aspects of a viewing episode can be used to compute additional measurements of eye movement behavior for visual design analysis. For instance, the spatial distribution of fixations can be captured by the moments of the distribution or the **coverage** (proportion of visual design fixated at a particular threshold value, Sect. 3.3). The temporal ordering (sequence) of fixations is often referred to as the **scanpath** [45] and is common for analyzing search tasks (Fig. 1d). For instance, one can consider the sequence of AOIs observers fixate while searching for a target or a specific piece of information.

Quantitative eye movement measurements used by previous visualization studies are summarized in Table 1. A review of the most common eye measurements across usability studies more generally is provided by Jacob and Karn [29]. The 5 most common metrics reported across 24 usability studies also appear in Table 1. Different metrics emphasize different aspects of eye movement behavior, which are in turn linked to different underlying cognitive processes. The number or density of fixations allocated to a visual area has been linked to its importance [29, 52]; fixation duration in a visual area has been linked to the area's information content or complexity [32]; and the transitions between fixations have been found to be related to the search behavior and expectations of the viewer [15, 44, 54]. Patterns in the fixation data of a group of observers can also be used to highlight design

Table 1 Eye movement metrics commonly reported in usability studies [29] and for evaluation and comparison of information visualizations. Different perception studies have used these metrics to make conclusions about the importance and noticeability of different visual elements, and to reason about the difficulty of the perception task and the complexity of the visual design [51]. AOI refers to an *Area of Interest*, which can be a component of a graph like the title, axis, or legend

Quantitative measurements	Visualization studies	Possible interpretations
Summary measurements		
Total number of fixations[a]	[17, 39]	Efficiency of searching or engagement [12, 19, 32]
Total number of gazes	[11]	Complexity of inferential process [11]
Mean fixation duration[a]		Complexity or engagement [32]
AOI measurements		
Fixations on AOIs[a] (proportion or number)	[8, 37, 62]	Element importance or noticeability [52]
Gazes on AOIs[a] (proportion or number)	[11]	Element importance or noticeability [29]
Viewing time on AOIs[a] (proportion or total)	[11, 37, 62]	Information content, complexity, or engagement [32]
Time to first fixation on an AOI	[17, 39, 62]	Attention-getting properties [10]
Mostly qualitative analysis	[25–27, 49, 53]	Relative complexity or efficiency of different designs

[a]The marked metrics are the 5 most commonly-reported across a total of 24 usability studies surveyed by Jacob and Karn [29]

features or diagnose potential problems. For instance, the order of fixations has been found to be indicative of the efficiency of the arrangement of visual elements [15]. A visualization designer might be interested in ensuring that the important elements are more likely to be fixated early.

The use of different types of visualizations for highlighting properties of eye movement data have also been useful for complementing and facilitating analysis over groups of observers [1, 18, 41, 57, 64, 66, 68, 69]. A number of previous visualization studies relied mostly on such qualitative analyses (Table 1). Blascheck et al. provide a review of visualizations and visual analytics tools for eye movement data [3]. While visualizations can facilitate data exploration, inferences made from eye movement data are more meaningful when supported by quantitative metrics.

For the explorative analysis of the MASSVIS eye movement data, we utilize **fixation heatmaps** due to their versatility, scalability, and interpretability. Fixation heatmaps are constructed by aggregating a set of fixations and placing a Gaussian[4] at each fixation location. The result is a continuous distribution that can be plotted on top of the image to highlight elements receiving the most attention. This simple visualization is particularly amenable to collective analysis, allowing us to visualize the fixations of any number of observers on a single image. To highlight different trends in the eye movements, we aggregate over different subsets of the data: distinct fixation durations (Fig. 2), time points during the viewing episode (Fig. 3), and observers (Fig. 4). Our coverage plots are also just thresholded fixation heatmaps (Fig. 5).

We note that eye movement analyses are most informative in the context of an objective task that an observer performs. In such cases, eye movements are more likely to be related to task completion itself. Furthermore, eye movement analyses can be used to complement, and provide possible explanations for, other objective performance measurements (e.g., speed or accuracy of task completion). Considered in isolation, eye movement measurements can be open to interpretation, and thus they should complement, not replace, other measurements. For example, the eye movements from the MASSVIS dataset were collected in the context of memory and recall tasks. Participants' fixations were recorded as they examined visualizations, knowing they would have to retrieve the details from memory later. In this manuscript, our focus is on the eye movement metrics themselves and how they can be used for the evaluation and comparison of information visualizations more broadly. We use the MASSVIS dataset for demonstrative examples.

3 Analyses

In this section we demonstrate how the metrics listed in Table 1 can be used for collective eye movement analysis over a large dataset of visualizations and observers. We use the MASSVIS dataset for our examples. Summary fixation

[4]Typically, the sigma of the Gaussian is chosen to be equal to 1 or 2° of visual angle, to model the uncertainty in viewing location.

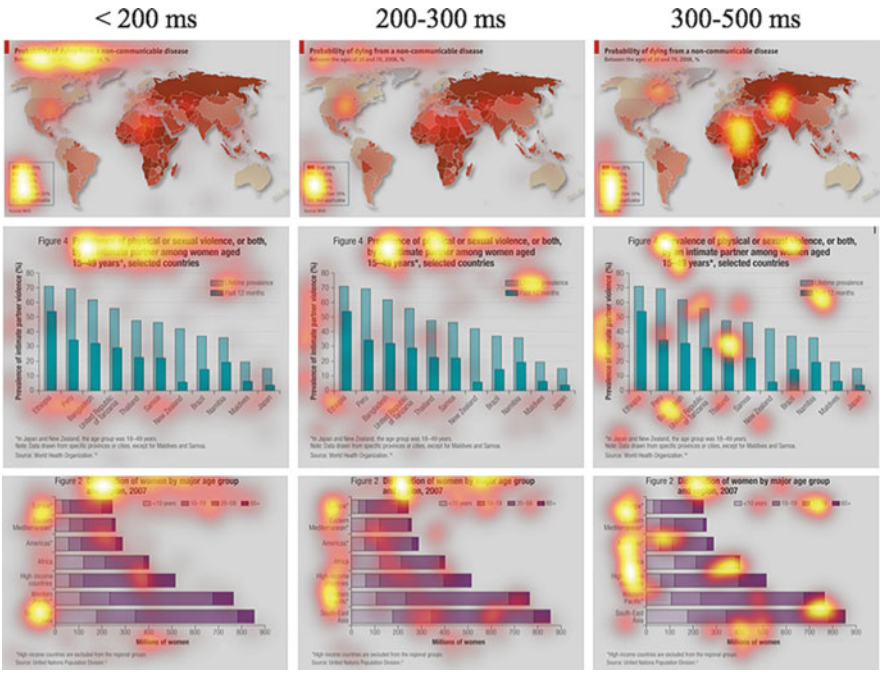


Fig. 2 Heatmaps created by selectively aggregating fixations of different durations, across all observers. Here we see that longer-duration fixations (300–500 ms) are used to explore more of the data elements. Fixation durations have been linked to the complexity and informativeness of a visual area [15, 48, 54]

measurements (Sect. 3.1) can be used for a very coarse analysis of the fixation data to compare groups of visualizations, for instance by source type. Having areas of interest labeled on individual visualizations allows us to perform a finer-grained analysis (Sect. 3.2) to investigate which elements capture observer attention the earliest, the most number of times, and for the longest interval of time. The advantage of a fixed set of labels is that statistics can be aggregated over many different visualizations to discover general trends. Aside from using the common metrics from Table 1, we also show the utility of coverage (Sect. 3.3) and inter-observer consistency (Sect. 3.4) analyses to derive additional diagnostics about visualization designs.

3.1 Summary Fixation Measurements

To summarize fixation behavior across images and observers for a given task, eye-tracking studies often consider the average number and duration of fixations required for task completion. The advantage of these coarse measurements are that

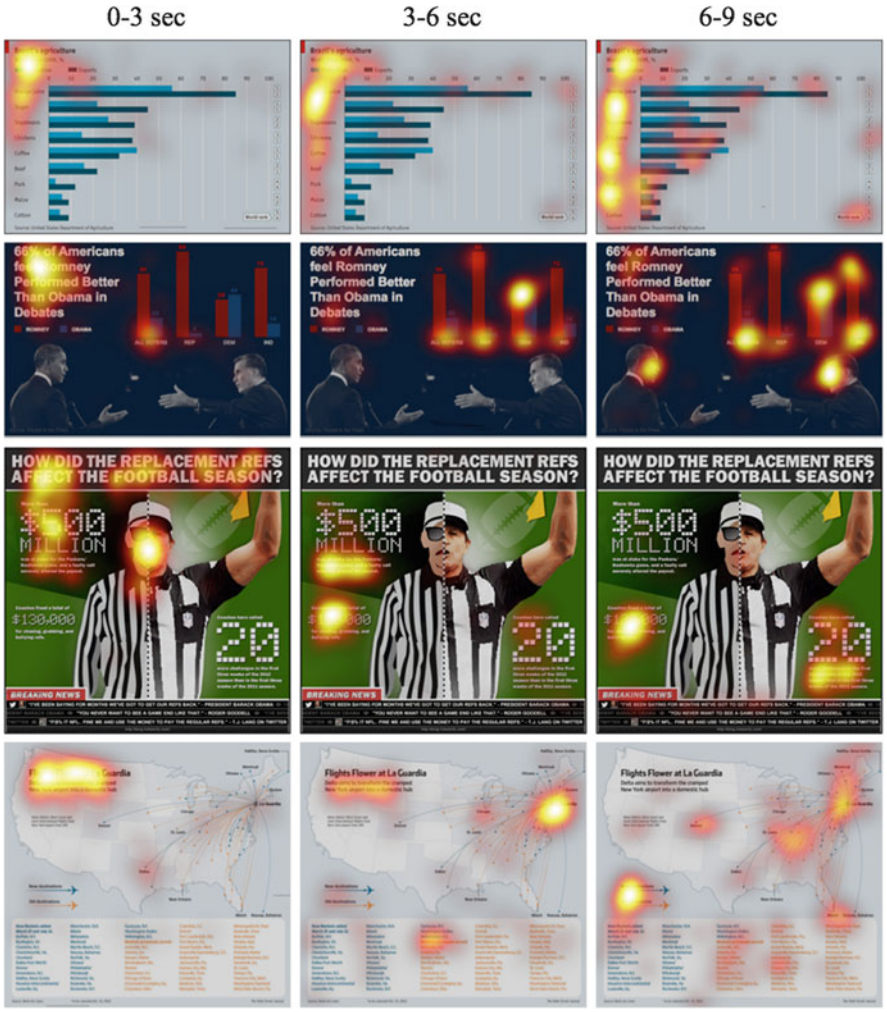


Fig. 3 Viewing behavior unfolding over time is visualized by aggregating fixations during specific intervals of time. Titles are consistently fixated earliest, followed by explanatory paragraphs. The data itself is explored after much of the text

they are easy to compute, independent of image content, and can be aggregated over any number of data points. These measurements are particularly meaningful when there is an objective task for an observer to complete, such as searching for a particular piece of information in a visualization. Studies can investigate whether fewer fixations are required to solve a task using one visualization design compared to another. These measurements can also be used to make inferences about observer engagement, with the caveat that there may be confounding factors such as the amount of information on a visualization, and relying on these metrics alone may

Fig. 4 *Top row*: A visualization with high inter-observer consistency (IOC). All observers have a similar fixation pattern on this visualization. This visualization tends to consistently guide the observer's attention. *Bottom row*: a visualization with low inter-observer consistency (IOC). Different observers examine the visualization in different ways but will they get the same information out of it? For ease of comparing the fixation patterns of different observers, the underlying visualizations have been gray scaled

not be sufficient. All the results reported below correspond to numerical values computed on 393 MASSVIS visualizations, and reported in Table 2 in the Appendix.

Total number of fixations: Aggregating over target visualizations from different source categories, the news media visualizations contained the most fixations on average, significantly more than the other visualization sources.

Total number of gazes: By aggregating fixations into gazes, we can avoid double counting fixations with different pixel coordinates on the image, but still falling within the same set of AOIs. For instance, for an observer reading a piece of text, all *consecutive* fixations falling on the text are considered part of a single gaze. Analyzing gazes, we find that the same patterns hold as with fixation counts, with the news media visualizations containing the largest number of gazes on average. This shows that the eyes moved around most between elements on the news media visualizations than on any of the other visualization sources. Is there more to look at on the news media visualizations? The number of visualization elements is actually highest for the infographics. We can use these metrics to hypothesize that observers were more engaged by the news media visualizations, but additional user studies would be needed for validation.

Mean fixation duration: The duration of individual fixations has significance in the psychology literature. For instance, shorter-duration fixations, less than about 200–250 ms, are sometimes considered involuntary (the eyes move there without

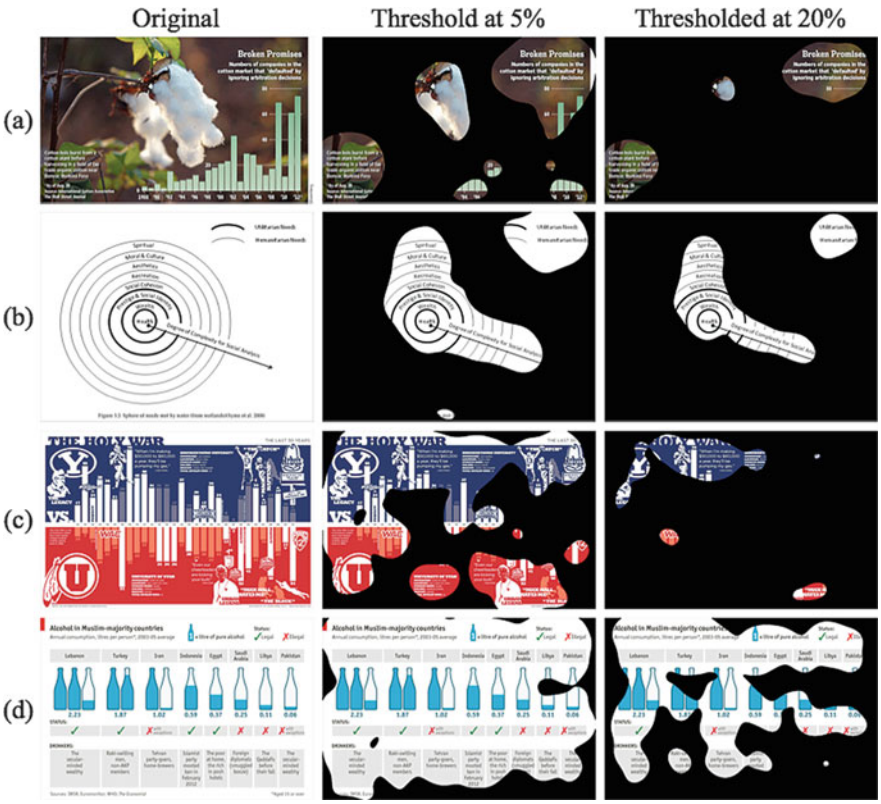


Fig. 5 Analyzing fixation coverage can help diagnose potential design issues. (**a**) The photographic element may have distracted observers, who paid no attention to the bar graph; (**b**) The title at the bottom, explaining the visualization, was missed; (**c**) Less crucial quotes captured more attention than explanatory text; (**d**) A visualization with many components and high coverage – observers were engaged, and examined the majority of the visualization. Different thresholds for plotting coverage can be used to visualize regions of an image fixated by different proportions of observers. We plot the thresholds at 5 % and 20 % of the maximum heatmap value

a conscious decision) [20]. Fixations longer than about 300 ms are thought to be encoded in memory. Across the MASSVIS target visualizations, the mean fixation duration is longer for infographics and scientific visualizations. These visualizations contain many diagrams and other visually-engaging elements, and have been found to be the most memorable [6].

By plotting heatmaps of fixations at various durations in Fig. 2, we can see which elements of a visualization are explored for shorter or longer periods of time, and thus potentially differentially processed. Durations of fixations have been found to be related to the complexity and difficulty of the visual content and task being performed [15, 48, 54]. Thus, considering locations in a visualization

receiving fixations of increased duration could be used to discover elements of the visualization that are engaging the cognitive resources of the observer.

3.2 AOI Fixation Measurements

Having labeled (pre-segmented) visualization elements allows statistics to be aggregated over observers and visualizations, to relate eye movements back to these elements, and get a finer-grained picture of observer attention. In the eye-tracking literature, segmented image regions for quantifying eye movement behavior are often called Areas of Interest (AOIs) or Regions of Interest (ROIs). Note that in some cases, as in ours, the AOIs are meaningful parts of the visual content and are pre-segmented for analysis. In other cases AOIs may be defined by clustering the eye movements during post-processing.[5] All of the results reported below correspond to the plots included in Figs. 6 and 7 of the Appendix.

Fixations on AOIs: Fixation statistics across AOIs can be aggregated over all visualizations to make conclusions about general design principles. For example, over all 393 target visualizations of the MASSVIS dataset, the legend, table header row (i.e., top label row of a table), paragraph, and title elements receive on average the largest number of fixations. However, when aggregating over multiple instances in a visualization of each element type, we find that observers make more fixations on the paragraph and label element types, although any individual label in a visualization would receive fewer fixations than the legend.

Gazes on AOIs: Within a single gaze, paragraphs receive the most fixations. But by aggregating fixations into gazes, the header row and legend receive the most gazes. Observers return to header rows and legends most frequently, which is why they end up with the most fixations overall. These specific elements allow the information in a visualization to be clarified and integrated.

Viewing time on AOIs: The viewing time (in ms) can be a measure of the importance or information content of a visualization element [32]. We find that of the total number of time spent fixating visualizations, legends, header rows, paragraphs, and titles were fixated the longest. This corresponds to the fact that these elements received the most fixations overall, another measure of importance.

Time to first fixation on an AOI: An analysis of scanpaths can indicate which elements are fixated first and which elements are fixated multiple times during the entire viewing episode. Over all observers and visualizations, we can find the average fixation number on which each element was first fixated. Across the MASSVIS target visualizations, the elements fixated earliest are titles, objects,

[5] Goldberg and Helfman [17] discuss implementation choices and issues arising when working with AOIs and fixations.

paragraphs, and header rows. These are textual elements from which an observer can expect to learn the most about what the visualization is conveying (important elements) and visual depictions that attract attention (noticeable elements). A complementary visualization can depict these trends. We selectively aggregated over fixations at different time points in the viewing episode, splitting the viewing time into 3 segments of 3 s each, and computed fixation heatmaps. As depicted in Fig. 3, titles consistently receive attention in the first 3 s of viewing time. Then fixations move to the paragraphs, other explanatory text, and data elements.

Overall, observers tend not to dwell on pictograms and purely-visual elements, and instead spend most of the time reading text. This supports previous findings that viewers start by visiting, and spend more time on, textual elements than pictorial elements [55]. This does not mean that observers do not look at pictograms. However, fixations on these elements do not last long: observers look at these elements, and move on. Considering a number of different fixation metrics concurrently paints a clearer picture of observer eye movement behavior.

Of all the textual elements, titles are often first to be examined, and in general, receive a lot of attention during the viewing episode. Our eye movement analyses point to the importance of these elements, while additional quantitative analyses reported in Borkin et al. confirm that titles are highly memorable elements that are often recalled by participants, and can aid or hinder comprehension of a visualization [6]. In such a way, eye movement measurements can complement additional task-specific analyses.

3.3 Coverage

Coverage, related to spatial density metrics [12, 19], is computed by aggregating the fixations of all observers, thresholding the resulting fixation heatmap at some fixed value, and measuring the amount of image area covered by fixations [68]. Coverage can be visualized (as in Fig. 5) by masking out image regions with fixation values below the threshold. Image regions that survive high thresholds are those that receive the most fixations. Applying the same threshold to different information visualizations can facilitate comparison across designs. A lower coverage value indicates that observers tend to look at a smaller portion of the visualization.

Analyzing coverage can help diagnose potential design issues. If a large part of the visualization is covered in data but fixation coverage is low, observers may have missed important components of the visualization or crucial parts of the message (Fig. 5). Across the MASSVIS target visualizations, infographic visualizations have on average more coverage than any of the other visualization sources. Although these differences are not statistically significant, a trend surfaces across 3 different threshold values. Another way to look at this trend is that among the 50 visualizations with highest coverage (at a 20 % threshold), 38 % are infographics, while of the 50 visualizations with lowest coverage, 38 % are news media. Does this contradict the fixation analyses? Both infographics and news media visualizations receive a

high number of fixations, indicating high observer engagement, but the news media visualizations in the MASSVIS dataset tend to be simpler and have fewer elements. As a result, fixations on the news media visualizations are more clustered around a few components, leading to lower coverage. By considering multiple fixation metrics, a fuller story unfolds.

3.4 Inter-observer Consistency

Inter-observer consistency (**IOC**) is used in saliency research[6] to quantify the similarity of observer fixations on an image. IOC for an image is a measure of how well the fixation heatmap of N-1 observers predicts the fixation heatmap of the remaining observer, averaging over all N observers, under some similarity metric.[7] We propose that IOC analysis can be used to determine how the design of an information visualization guides observers. High IOC implies that observers tend to have similar fixation patterns, while a low IOC corresponds to different observers examining a visualization in different ways. In the latter case, it is worth measuring if the different possible fixation patterns will lead observers to derive similar conclusions from the visualization. Will the message of the visualization be clear no matter how the visualization is examined? Did the designer of the visualization intend the visualization to be viewed in a particular way? Figure 4 contains example fixation heatmaps for a visualization with high IOC and one with low IOC. In general, dense and crowded visualizations with a lot of information have low IOC; there is a lot to look at, and different observers choose to look at different things. Simple, well-structured visualizations (e.g., with a standard layout) direct observer attention, so different observers look at these visualizations in similar ways. For example, across the MASSVIS target visualizations, infographic visualizations have lower IOC than any of the other source categories, and news media visualizations have the highest IOC. This goes along with the coverage story: with fewer elements to look at in a visualization, observers are more consistent about where they look.

4 Conclusion

In this manuscript we reviewed a number of existing eye movement metrics and considered their utility for the collective analysis of large, diverse datasets of visualizations. By aggregating statistics over observers and visualizations, these

[6]This has also been called inter-subject consistency [67], the inter-observer (IO) model [5], and inter-observer congruency (IOC) [42].

[7]Area under receiver operating characteristic curve (AUROC or AUC) is the most commonly used similarity metric [14]. Note that IOC analysis can be extended to the ordering, instead of just the distribution, of fixations [18, 31, 42, 45].

metrics can be used to quantitatively evaluate different types and designs of visualizations. We also discussed techniques for visualizing properties of fixation behavior that these metrics aim to capture.[8] Whereas we focused mostly on the distribution of eye fixations, a more thorough investigation of other properties of eye movement behavior like scanpaths and saccades are likely to provide additional insights. This manuscript contributed a discussion of broader, more large-scale comparison methods than prior visualization studies.

The need will only increase for metrics and analyses that can scale to processing data of potentially hundreds of observers on thousands of images. New methodologies are opening up opportunities of collecting user attention patterns, to approximate or replace costly eye tracker recordings, at larger scales than previously possible [30, 36, 58].

Moreover, some of the design evaluations discussed might be possible without collecting any user data at all. Many computational models have been developed over the past couple of decades to predict eye movements, specifically fixations and attention patterns on natural images.[9] In recent years, computational predictions have begun to come very close to ground truth human eye movements on photographs [9]. Models for predicting eye movements on graphic designs, webpages, and visual interfaces are also beginning to show promise [46, 61]. As computational models continue to evolve, opportunities will open up to evaluate visual designs, including information visualizations, in a fully automatic manner. For instance, O'Donovan et al. computationally predict the importance of different visual elements in graphic designs [46], Berg et al. predict the importance of elements and objects in natural images [2], and Khosla et al. predict the memorability of different image regions, automatically generating a kind of importance map per image [35]. Le Meur et al. directly predict inter-observer congruency (IOC) for images without user data [43]. Automatic predictions of image interestingness [22], style [33], aesthetics [56], and memorability [28, 34] are already possible. Such computational predictions have the potential of making their way into designer tools, to provide real-time feedback on visual designs and visualizations. Importantly, these computational predictions are all informed by studies and measurements of human perception and cognition. The results of eye movement analyses thus have the potential to make simultaneous contributions to the understanding of human cognitive and perceptual processes, visual content design principles, and better automatic design predictions in the future.

Acknowledgements This work was partly funded by awards from Google and Xerox to A.O., NSERC Postgraduate Doctoral Scholarship (PGS-D) to Z.B., NSF Graduate Research Fellowship Program and NSERC Discovery grant to M.B., and a Kwanjeong Educational Foundation grant to N.K.

[8] Labeled visualizations, eye movement data, and code for the visualizations in this manuscript are available at http://massvis.mit.edu.

[9] We suggest the following surveys: [4, 5, 16, 38, 65].

Appendix

Table 2 Eye fixations on a total of 393 MASSVIS visualizations are analyzed and discussed in Sect. 3.1. Measurements are first aggregated across all observers per visualization, to obtain an average value for each visualization. Then statistics are computed over all the visualizations per source category for a comparison across the categories: infographic, news media, scientific, and government. The t-statistic is reported for each pairwise t-test in the final column (Bonferonni-corrected for multiple comparisons)

Summary measurements	Infographics (92 vis)	News (122 vis)	Science (79 vis)	Government (100 vis)	Pairwise comparisons
Number of elements	M = 38.7 (SD = 32.9)	M = 19.7 (SD = 14.0)	M = 18.4 (SD = 10.8)	M = 11.9 (SD = 7.4)	t(212) = 5.73** t(177) = 4.79** t(169) = 5.23**
Total number of fixations	M = 37.3 (SD = 3.1)	M = 39.0 (SD = 2.6)	M = 34.6 (SD = 3.1)	M = 37.7 (SD = 2.5)	t(212) = 4.39** t(177) = 7.56** t(169) = 5.77** t(220) = 3.80**
Total number of gazes	M = 33.7 (SD = 3.7)	M = 33.9 (SD = 3.5)	M = 32.3 (SD = 3.7)	M = 31.9 (SD = 3.3)	t(199) = 3.20* t(190) = 3.65* t(220) = 4.56**
Mean fixation duration	M=238.6 (SD = 26.5)	M = 218.6 (SD = 16.1)	M = 245.3 (SD = 26.9)	M=221.3 (SD = 15.9)	t(212) = 6.82** t(177) = 7.44** t(190) = 5.55**
Coverage (5 %)	M = 0.59 (SD = 0.15)	M = 0.55 (SD = 0.12)	M = 0.57 (SD = 0.14)	M = 0.57 (SD = 0.12)	
Coverage (10 %)	M = 0.43 (SD = 0.15)	M = 0.39 (SD = 0.12)	M = 0.41 (SD = 0.13)	M = 0.42 (SD = 0.12)	
Coverage (20 %)	M = 0.26 (SD = 0.12)	M = 0.23 (SD = 0.09)	M = 0.23 (SD = 0.09)	M = 0.25 (SD = 0.09)	
IOC (20 %)	M = 0.81 (SD = 0.05)	M = 0.83 (SD = 0.03)	M = 0.82 (SD = 0.04)	M = 0.82 (SD = 0.03)	

Colored markers indicate which pairwise comparison each t-statistic corresponds to
Tests with $p < 0.05$ are marked with (*) and those corresponding to $p < 0.01$ are marked with (**)
Note, for clarity, not every pairwise comparison is reported. The highest value for each measurement is highlighted in gray

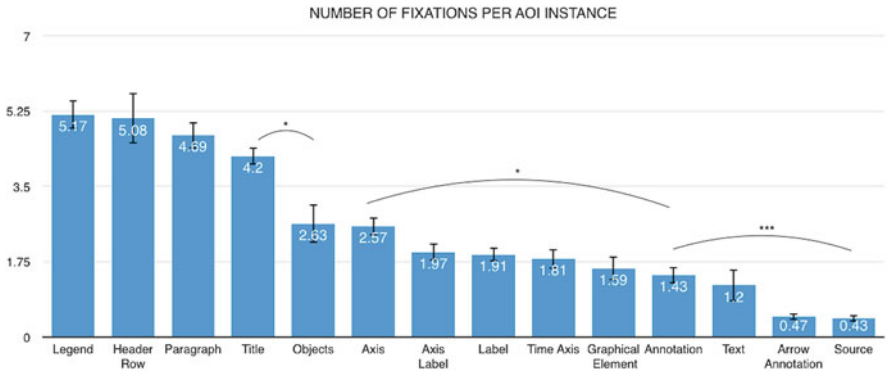


Fig. 6 These plots correspond to the results reported in Sect. 3.2. Note that Bonferonni-corrected pairwise t-tests with $p < 0.05$ are marked with (*), $p < 0.01$ with (**), and $p < 0.001$ with (***). For clarity, not all pairwise comparisons are plotted

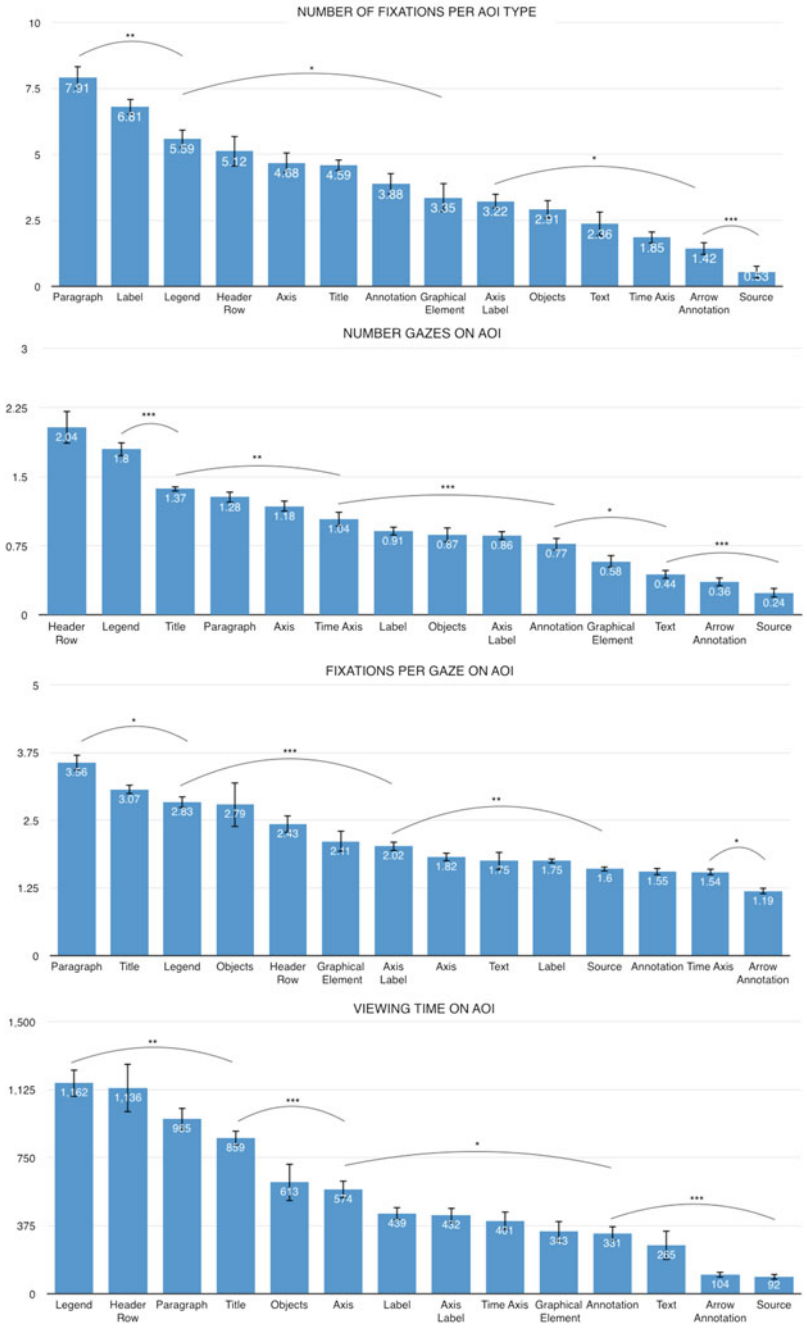


Fig. 7 These plots correspond to the results reported in Sect. 3.2. Note that Bonferonni-corrected pairwise t-tests with $p < 0.05$ are marked with (*), $p < 0.01$ with (**), and $p < 0.001$ with (***). For clarity, not all pairwise comparisons are plotted

References

1. Andrienko, G., Andrienko, N., Burch, M., Weiskopf, D.: Visual analytics methodology for eye movement studies. IEEE TVCG **18**(12), 2889–2898 (2012)
2. Berg, A.C., Berg, T.L., Daume III, H., Dodge, J., Goyal, A., Han, X., Mensch, A., Mitchell, M., Sood, A., Stratos, K., et al.: Understanding and predicting importance in images. In: Computer Vision and Pattern Recognition, pp. 3562–3569. IEEE, Providence, RI (2012)
3. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: State-of-the-art of visualization for eye tracking data. In: Proceedings of EuroVis, vol. 2014, Swansea (2014)
4. Borji, A., Sihite, D.N., Itti, L.: Quantitative analysis of human-model agreement in visual saliency modeling: a comparative study. IEEE Trans. Image Process. **22**(1), 55–69 (2013)
5. Borji, A., Tavakoli, H.R., Sihite, D.N., Itti, L.: Analysis of scores, datasets, and models in visual saliency prediction. In: IEEE International Conference on Computer Vision, Sydney (2013)
6. Borkin, M., Bylinskii, Z., Kim, N., Bainbridge, C.M., Yeh, C., Borkin, D., Pfister, H., Oliva, A.: Beyond memorability: visualization recognition and recall. IEEE TVCG **22**(1), 519–528 (2016)
7. Borkin, M.A., Vo, A.A., Bylinskii, Z., Isola, P., Sunkavalli, S., Oliva, A., Pfister, H.: What makes a visualization memorable? IEEE TVCG **19**(12), 2306–2315 (2013)
8. Burch, M., Konevtsova, N., Heinrich, J., Hoeferlin, M., Weiskopf, D.: Evaluation of traditional, orthogonal, and radial tree diagrams by an eye tracking study. IEEE TVCG **17**(12), 2440–2448 (2011)
9. Bylinskii, Z., Judd, T., Borji, A., Itti, L., Durand, F., Oliva, A., Torralba, A.: MIT Saliency Benchmark. http://saliency.mit.edu/
10. Byrne, M.D., Anderson, J.R., Douglass, S., Matessa, M.: Eye tracking the visual search of click-down menus. In: SIGCHI, pp. 402–409. ACM, New York (1999)
11. Carpenter, P.A., Shah, P.: A model of the perceptual and conceptual processes in graph comprehension. J. Exp. Psychol. Appl. **4**(2), 75 (1998)
12. Cowen, L., Ball, L.J., Delin, J.: An eye movement analysis of web page usability. In: People and Computers XVI, pp. 317–335. Springer, London (2002)
13. Duchowski, A.T.: A breadth-first survey of eye-tracking applications. Behav. Res. Methods Instrum. Comput. **34**(4), 455–470 (2002)
14. Fawcett, T.: An introduction to ROC analysis. Pattern Recognit. Lett. **27**(8), 861–874 (2006)
15. Fitts, P.M., Jones, R.E., Milton, J.L.: Eye movements of aircraft pilots during instrument-landing approaches. Ergon. Psychol. Mech. Models Ergon **3**, 56 (2005)
16. Frintrop, S., Rome, E., Christensen, H.I.: Computational visual attention systems and their cognitive foundations: a survey. ACM Trans. Appl. Percept. (TAP) **7**(1), 6 (2010)
17. Goldberg, J.H., Helfman, J.I.: Comparing information graphics: a critical look at eye tracking. In: BELIV'10, Atlanta, pp. 71–78. ACM (2010)
18. Goldberg, J.H., Helfman, J.I.: Scanpath clustering and aggregation. In: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, Austin, pp. 227–234. ACM (2010)
19. Goldberg, J.H., Kotval, X.P.: Computer interface evaluation using eye movements: methods and constructs. Int. J. Ind. Ergon. **24**(6), 631–645 (1999)
20. Graf, W., Krueger, H.: Ergonomic evaluation of user-interfaces by means of eye-movement data. In: Proceedings of the Third International Conference on Human-Computer Interaction, Boston, pp. 659–665. Elsevier Science Inc. (1989)
21. Grant, E.R., Spivey, M.J.: Eye movements and problem solving guiding attention guides thought. Psychol. Sci. **14**(5), 462–466 (2003)
22. Gygli, M., Grabner, H., Riemenschneider, H., Nater, F., Gool, L.: The interestingness of images. In: International Conference on Computer Vision, Sydney, pp. 1633–1640 (2013)

23. Hayhoe, M.: Advances in relating eye movements and cognition. Infancy **6**(2), 267–274 (2004)
24. Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., Van de Weijer, J.: Eye Tracking: A Comprehensive Guide to Methods and Measures. Oxford University Press, Oxford/New York (2011)
25. Huang, W.: Using eye tracking to investigate graph layout effects. In: APVIS'07, Sydney, pp. 97–100 (2007)
26. Huang, W., Eades, P.: How people read graphs. In: APVIS'05, Sydney, vol. 45, pp. 51–58 (2005)
27. Huang, W., Eades, P., Hong, S.-H.: A graph reading behavior: geodesic-path tendency. In: PacificVis'09, Kyoto, pp. 137–144 (2009)
28. Isola, P., Xiao, J., Torralba, A., Oliva, A.: What makes an image memorable? In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, pp. 145–152. IEEE (2011)
29. Jacob, R., Karn, K.S.: Eye tracking in human-computer interaction and usability research: ready to deliver the promises. Mind **2**(3), 4 (2003)
30. Jiang, M., Huang, S., Duan, J., Zhao, Q.: Salicon: saliency in context. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston (2015)
31. Josephson, S., Holmes, M.E.: Visual attention to repeated internet images: testing the scanpath theory on the world wide web. In: Proceedings of the 2002 Symposium on Eye Tracking Research & Applications, Palm Beach Gardens, pp. 43–49. ACM (2002)
32. Just, M.A., Carpenter, P.A.: Eye fixations and cognitive processes. Cogn. Psychol. **8**(4), 441–480 (1976)
33. Karayev, S., Trentacoste, M., Han, H., Agarwala, A., Darrell, T., Hertzmann, A., Winnemoeller, H.: Recognizing image style (2013). In: Proceedings British Machine Vision Conference (2014)
34. Khosla, A., Raju, A.S., Torralba, A., Oliva, A.: Understanding and predicting image memorability at a large scale. In: Proceedings of the IEEE International Conference on Computer Vision, Santiago, pp. 2390–2398 (2015)
35. Khosla, A., Xiao, J., Torralba, A., Oliva, A.: Memorability of image regions. In: NIPS, Lake Tahoe, pp. 305–313 (2012)
36. Kim, N.W., Bylinskii, Z., Borkin, M.A., Oliva, A., Gajos, K.Z., Pfister, H.: A crowdsourced alternative to eye-tracking for visualization understanding. In: CHI'15 Extended Abstracts, Seoul, pp. 1349–1354. ACM (2015)
37. Kim, S.-H., Dong, Z., Xian, H., Upatising, B., Yi, J.S.: Does an eye tracker tell the truth about visualizations? Findings while investigating visualizations for decision making. IEEE TVCG **18**(12), 2421–2430 (2012)
38. Kimura, A., Yonetani, R., Hirayama, T.: Computational models of human visual attention and their implementations: a survey. IEICE Trans. Inf. Syst. **96-D**, 562-578 (2013)
39. Körner, C.: Eye movements reveal distinct search and reasoning processes in comprehension of complex graphs. Appl. Cogn. Psychol. **25**(6), 893–905 (2011)
40. Kowler, E.: The role of visual and cognitive processes in the control of eye movement. Rev. Oculomot. Res. **4**, 1–70 (1989)
41. Lankford, C.: Gazetracker: software designed to facilitate eye movement analysis. In: Proceedings of the 2000 Symposium on Eye Tracking Research & Applications, Palm Beach Gardens, pp. 51–55. ACM (2000)
42. Le Meur, O., Baccino, T.: Methods for comparing scanpaths and saliency maps: strengths and weaknesses. Behav. Res. Methods **45**(1), 251–266 (2013)
43. Le Meur, O., Baccino, T., Roumy, A.: Prediction of the inter-observer visual congruency (IOVC) and application to image ranking. In: Proceedings of the 19th ACM International Conference on Multimedia, pp. 373–382. ACM, New York (2011)
44. Loftus, G.R., Mackworth, N.H.: Cognitive determinants of fixation location during picture viewing. J. Exp. Psychol. Hum. Percept. Perform. **4**(4), 565 (1978)
45. Noton, D., Stark, L.: Scanpaths in saccadic eye movements while viewing and recognizing patterns. Vis. Res. **11**(9), 929 (1971)

46. O'Donovan, P., Agarwala, A., Hertzmann, A.: Learning layouts for single-page graphic designs. IEEE TVCG **20**(8), 1200–1213 (2014)
47. Pan, B., Hembrooke, H.A., Gay, G.K., Granka, L.A., Feusner, M.K., Newman, J.K.: The determinants of web page viewing behavior: an eye-tracking study. In: Proceedings of the 2004 Symposium on Eye Tracking Research & Applications, San Antonio, pp. 147–154. ACM (2004)
48. Pelz, J.B., Canosa, R., Babcock, J.: Extended tasks elicit complex eye movement patterns. In: Proceedings of the 2000 Symposium on Eye Tracking Research & Applications, Palm Beach Gardens, pp. 37–43. ACM (2000)
49. Pohl, M., Schmitt, M., Diehl, S.: Comparing the readability of graph layouts using eyetracking and task-oriented analysis. In: Computational Aesthetics in Graphics, Visualization and Imaging, Lisbon, pp. 49–56 (2009)
50. Pomplun, M., Ritter, H., Velichkovsky, B.: Disambiguating complex visual information: towards communication of personal views of a scene. Perception **25**, 931–948 (1996)
51. Poole, A., Ball, L.J.: Eye tracking in HCI and usability research. Encycl. Hum. Comput. Interact. **1**, 211–219 (2006)
52. Poole, A., Ball, L.J., Phillips, P.: In search of salience: a response-time and eye-movement analysis of bookmark recognition. In: People and Computers XVIII, pp. 363–378. Springer, London (2004)
53. Raschke, M., Blascheck, T., Richter, M., Agapkin, T., Ertl, T.: Visual analysis of perceptual and cognitive processes. In: IEEE International Conference on Information Visualization Theory and Applications (IVAPP), pp. 284–291 (2014).
54. Rayner, K.: Eye movements in reading and information processing: 20 years of research. Psychol. Bull. **124**(3), 372 (1998)
55. Rayner, K., Rotello, C.M., Stewart, A.J., Keir, J., Duffy, S.A.: Integrating text and pictorial information: eye movements when looking at print advertisements. J. Exp. Psychol. Appl. **7**(3), 219 (2001)
56. Reinecke, K., Yeh, T., Miratrix, L., Mardiko, R., Zhao, Y., Liu, J., Gajos, K.Z.: Predicting users' first impressions of website aesthetics with a quantification of perceived visual complexity and colorfulness. In: SIGCHI, San Jose, pp. 2049–2058. ACM (2013)
57. Ristovski, G., Hunter, M., Olk, B., Linsen, L.: EyeC: coordinated views for interactive visual exploration of eye-tracking data. In: 17th International Conference on Information Visualisation, London, pp. 239–248 (2013)
58. Rudoy, D., Goldman, D.B., Shechtman, E., Zelnik-Manor, L.: Crowdsourcing gaze data collection (2012). arXiv preprint arXiv:1204.3367
59. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: LabelMe: a database and web-based tool for image annotation. Int. J. Comput. Vis. **77**(1–3), 157–173 (2008)
60. Salvucci, D.D., Goldberg, J.H.: Identifying fixations and saccades in eye-tracking protocols. In: Proceedings of the 2000 Symposium on Eye Tracking Research & Applications, Palm Beach Gardens, pp. 71–78. ACM (2000)
61. Shen, C., Zhao, Q.: Webpage saliency. In: European Conference on Computer Vision, Zurich, pp. 33–46. Springer (2014)
62. Siirtola, H., Laivo, T., Heimonen, T., Raiha, K.-J. Visual perception of parallel coordinate visualizations. In: International Conference on Information Visualisation, Barcelona, pp. 3–9 (2009)
63. SR Research Ltd.: EyeLink Data Viewer User's Manual, Version 1.8.402 (2008)
64. Tsang, H.Y., Tory, M., Swindells, C.: eSeeTrack: visualizing sequential fixation patterns. IEEE TVCG **16**(6), 953–962 (2010)
65. Tsotsos, J.K., Rothenstein, A.: Computational models of visual attention. Scholarpedia **6**(1), 6201 (2011)
66. West, J.M., Haake, A.R., Rozanski, E.P., Karn, K.S.: eyePatterns: software for identifying patterns and similarities across fixation sequences. In: Proceedings of the 2006 Symposium on Eye Tracking Research & Applications, San Diego, pp. 149–154. ACM (2006)

67. Wilming, N., Betz, T., Kietzmann, T.C., König, P.: Measures and limits of models of fixation selection. PLoS ONE **6**(9), e24038 (2011)
68. Wooding, D.S.: Eye movements of large populations: deriving regions of interest, coverage, and similarity using fixation maps. Behav. Res. Methods Instrum. Comput. **34**(4), 518–528 (2002)
69. Wu, M.M.A., Munzner, T.: SEQIT: visualizing sequences of interest in eye tracking data. IEEE TVCG **22**(1), 449–458 (2015)

Index

© Springer International Publishing AG 2017

M. Burch et al. (eds.), *Eye Tracking and Visualization*, Mathematics and Visualization, DOI 10.1007/978-3-319-47024-5