Finding the Outliers in Scanpath Data

Michael Burch
Eindhoven University of Technology
m.burch@tue.nl

Ayush Kumar
Stony Brook University
aykumar@cs.stonybrook.edu

Klaus Mueller
Stony Brook University
mueller@cs.stonybrook.edu

Titus Kervezee
Eindhoven University of Technology
t.kervezee@tue.nl

Wouter Nuijten
Eindhoven University of Technology
w.nuijten@tue.nl

Rens Oostenbach
Eindhoven University of Technology
r.oostenbach@tue.nl

Lucas Peeters
Eindhoven University of Technology
l.peeters@tue.nl

Gijs Smit
Eindhoven University of Technology
g.smit@tue.nl

ABSTRACT

In this paper, we describe the design of an interactive visualization tool for the comparison of eye movement data with a special focus on the outliers. In order to make the tool usable and accessible to anyone with a data science background, we provide a web-based solution by using the Dash library based on the Python programming language and the Python library Plotly. Interactive visualization is very well supported by Dash, which makes the visualization tool easy to use. We support multiple ways of comparing user scanpaths like bounding boxes and Jaccard indices to identify similarities. Moreover, we support matrix reordering to clearly separate the outliers in the scanpaths. We further support the data analyst by complementary views such as gaze plots and visual attention maps.

CCS CONCEPTS

• **Human-centered computing → Visualization techniques**;

KEYWORDS

Eye tracking, information visualization, visual analytics

ACM Reference Format:
Michael Burch, Ayush Kumar, Klaus Mueller, Titus Kervezee, Wouter Nuijten, Rens Oostenbach, Lucas Peeters, and Gijs Smit. 2019. Finding the Outliers in Scanpath Data. In *Eye Tracking and Visualization (ETVIS @ ETRA'19), June 25–28, 2019, Denver , CO, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3317958.3318225

1 INTRODUCTION

Scanpaths from a list of eye tracking study participants can be regarded as a list of trajectories over space and time with additional metrics attached. Inspecting those scanpaths in standard visualization techniques, like given in a recent state-of-the-art report [Blascheck et al. 2017], will not clearly indicate the outliers in the data. Those visualizations are merely built for identifying common trends and aggregated visual attention patterns. However, the outliers are worth exploring since they describe abnormal viewing behavior that has to be understood to improve the visual stimulus, or to better guide the visual observer. We define an outlier (or anomaly) in a dataset by a data element (in this case a scanpath) that does not fall into the general category of all the other data elements. There is no concrete definition for the term outlier, but visualization is a powerful concept that makes use of the strengths of the human's perceptual system to identify patterns, trends, or commonalities in a dataset, consequently also outliers that do not follow a common behavior or category.

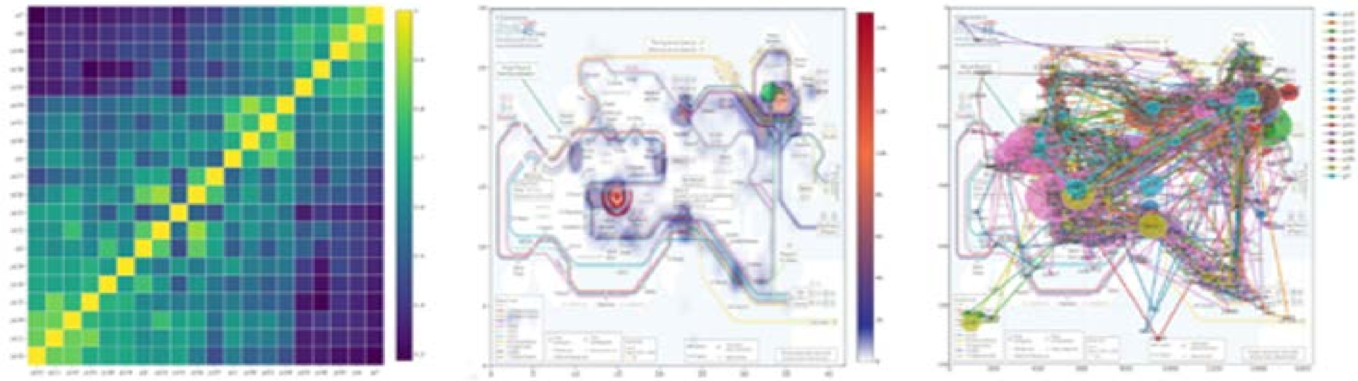
However, although they focus on providing a great overview about the data, the viewer is typically lost in the outlier and anomaly identification process [Healey and Enns 2012]. But without the overview, i.e., the normal scanpath behavior, it is not possible to detect the outliers among the scanpaths. Hence, we add additional views on the dataset by means of visual attention maps and gaze plots overplotted on the displayed stimulus to preserve the contextual information of the inspected scene in the eye tracking study.

The goal of this work is to first compare the scanpaths pairwisely in order to build a visual pattern for the similarities given in a comparison matrix[Kumar et al. 2018a]. These comparisons are built on naive measures like the Jaccard coefficient or the overlap of the smallest bounding boxes. Finally, this comparison matrix is ordered in a way to show the similarity structure of all pairwise comparisons (by applying a Robinsonian technique). Based on the strengths of the human's perceptual system [Ware 2004, 2008], the similar block patterns in the matrix can be identified rapidly, selected, filtered out, and consequently, the remaining non-similar comparison values build the outliers and are worth further investigations.

We implemented a visual outlier detection tool based on Python and the Dash library that supports an easy way for providing interactive graphics in a web browser. The tool allows to upload scanpath data while similarity functions and visualization techniques can be applied to explore the data with a particular focus on the outliers (see Figure 1 for example visualizations).

ETVIS @ ETRA'19, June 25–28, 2019, Denver , CO, USA

ACM ISBN 978-1-4503-6729-5/19/06…$15.00
https://doi.org/10.1145/3317958.3318225



(a) (b) (c)

Figure 1: The combination of different views supports the detection of outliers in the scanpath data. With a single view this is a challenging, even impossible, task to solve. Interactive and linked visualizations support the comparison of the scanpaths and the identification of the similarities and the outliers.

2 RELATED WORK

Finding outliers in datasets is a typical scenario in most information visualization [Ware 2004, 2008] or visual analytics applications [Keim 2012; Kurzhals et al. 2014]. However, to identify outliers or anomalies, we first have to define what we understand by the normal case in the data.

In particular, for eye movement data [Duchowski 2003; Holmqvist et al. 2011; Yarbus 1967] this means, first comparing the data over space and time [Andrienko et al. 2012] and then either identify those by algorithmic approaches or by using a visualization technique. Algorithmic solutions to this problem sometimes do not lead to the right answer since the used algorithms [Burch 2018; Burch et al. 2018a,b] cannot be specified in enough detail due to the fact that the human user is not able to really describe the meaning of similarity.

Looking at the pure gaze data in form of a gaze plot [Goldberg and Helfman 2010] does not help because if too many scanpaths are displayed we will end up in a visual clutter situation [Rosenholtz et al. 2005] that actually hides the outliers. Moreover, a visual attention map [Bojko 2009; Burch 2016] does not support the comparison of several study participants unless for each person an individual map is displayed next to each other. This, however, is not a scalable solution and it is also difficult to compute or identify real outliers, for which the aggregation over space, time, and participants is to blame.

To improve this drawback for example, the GazeStripes [Kurzhals et al. 2016] approach projects the scanpaths to horizontal bands of time-aligned image thumbnails and lets the user decide which parts look similar and which ones different. A comparison algorithm is integrated that hierarchically clusters the thumbnail bands based on image similarities. However, if the stimulus looks rather similar like a checker board, the visual inspection of these image bands does not lead to a solution. Moreover, algorithmic and visual grouping of participants and eye-tracking metrics can be too much to be analyzed by the analyst [Kumar et al. 2018b].

In our approach, we build a new comparison matrix-based visualization that is independent of the displayed stimuli and that scales to a large number of study participants and even long-durating tasks [Blascheck et al. 2015]. Moreover, it is easily extendable by additional comparison functions and matrix reordering techniques.

3 OUTLIER VISUALIZATION

We provide a visualization tool composed of several different views. To this end we support a color coded and reordered adjacency matrix complemented by additional perspectives on visual attention by a heatmap-like representation and a gaze plot, displayed on top of the stimulus.

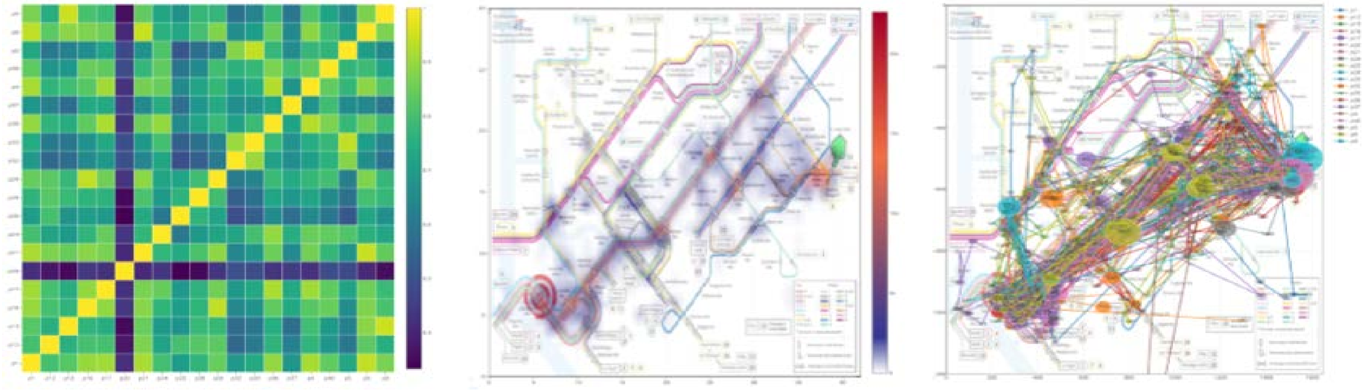
3.1 Graphical User Interface

The visualization tool is divided into three components. On top of the web page, users select the data they want to visualize, this consists of a stimulus and a set of participants. Below the data selection, there are multiple visualizations: on the left hand side are visualizations of eye tracking data and on the right hand side we depict the computed and ordered comparison matrix.

3.2 Visualization Techniques

Our tool has 3 different visualization techniques to get insights in the data. For the similarity comparison of the scanpaths of the different users we used an adjacency matrix (see Figure 2 (a)). This adjacency matrix contains all the users that are selected in the tool and correspond to their scanpath of the chosen stimulus. It goes without saying that the diagonal of this adjacency matrix is always the maximal value, each user corresponds with the own scanpath by 100 percent. Furthermore, we are able to apply reordering techniques in order to unveil certain patterns or anomalies in the dataset.

The adjacency matrix is the most important view in the visualization tool since it gives a scalable variant of scanpath comparisons, hence, similar patterns and anomalous behavior can be identified.



(a) (b) (c)

Figure 2: A color coded comparison matrix gives insights in scanpath outliers like the participant in column 6 and row 15 in (a). A visual attention map (b) and a gaze plot (c) give an overview of the scanning behavior on top of the visual stimulus. With this approach we can have a look at the stimulus for context information while at the same time observing the compared and clustered scanpaths.

Further interactions can be applied to reduce the amount of comparison data and to let the outliers pop up more clearly.

If only the aggregated visual attention is of interest we support the viewer by a visual attention map (see Figure 2 (b)). For looking at the raw scanpath data, we supply the user with a gaze plot (see Figure 2 (c)). In the gaze plot we see all the different coordinates of all the different users. We color code all coordinates according to the users they belong to, and their size corresponds to the fixation duration. Also, consecutive coordinates are connected by a line. In the plot the user has the option to isolate certain scanpaths, or exclude them from the plot to reduce visual clutter.

3.3 Comparison Methods

We support comparison functions for the scanpath data while several more will follow in the future.

- **Bounding box:** We create a bounding box around the complete scanpath of a user. A bounding box is consequently the minimal rectangular area that encloses the entire scanpath. For two scanpaths we compare both corresponding bounding boxes and compute the overlapping area of both as a measure for the similarity of two scanpaths. We normalize the computed overlap area value by dividing it by the value of the entire area that can be covered by a bounding box. A value of 1 indicates complete overlap while a value of 0 indicates no overlap at all.
- **Jaccard index:** We calculate the similarity between two sets, following the idea created by Paul Jaccard in 1901. It is defined as the size of the intersection (number of common elements in the two sets) divided by the size of the union of the sample sets (number of unique elements over both sets). In our approach we interpret the fixation coordinates of two scanpaths as two different fixation sets. We use a grid-based AOI subdivision to build a set of fixations by first identifying in which AOI each fixation is located. The grid cell sizes can be varied interactively.

The tool is coded using Python 3.6 because it is easy to extend the functionality of the program, for example, by additional comparison functions and it supports analysis libraries.

4 APPLICATION EXAMPLE

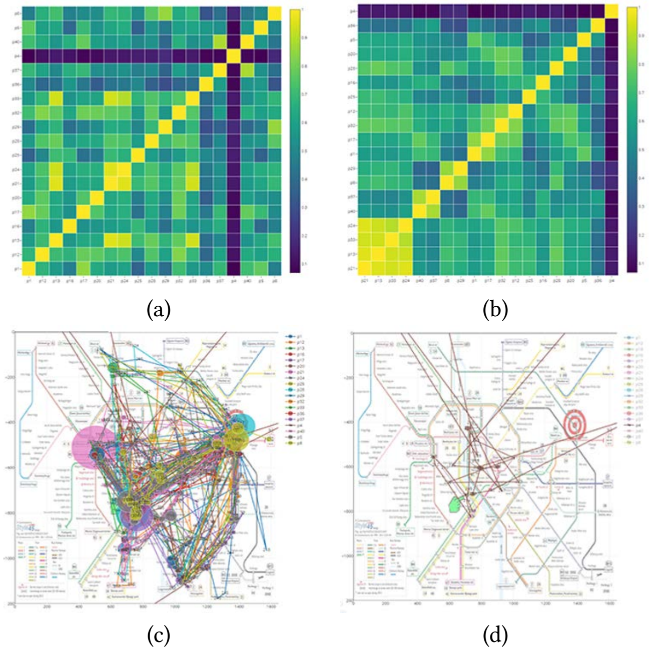
We applied our technique to an eye tracking dataset from a formerly conducted eye tracking experiment exploring route finding tasks in public transport maps [Netzel et al. 2017] (see Figure 3). We first upload the data to our server and preprocess it to make it ready for the interactive visualization tool.

4.1 Scanpath Data and Preprocessing

The analyzed dataset contained 96 different stimuli and eye movement data corresponding to these stimuli. For the eye movement data, there were 188,126 observations (number of rows) in total and 8 variables being timestamp, stimulus name, fixation index, fixation duration, mapped fixation point X, mapped fixation point Y, user, and description.

We decided to drop the fixation index, because it was highly correlated with the timestamp (p=0.97601) so we could be more efficient with memory space. Therefore, a final data row would consist of a timestamp, the name of the city, which included whether it was stimuli group 1 or stimuli group 2 of that city and whether it was colored or not, a fixation duration, which indicated how long the user looked at the stimuli, an x- and y-coordinate, the user, and a description, which also indicated whether the corresponding stimulus was colored or not.

We adapted fixation coordinates that are below 0, or higher than the resolution of the stimulus. This means that the person in the experiment was fixating somewhere else than to the monitor on which the stimulus was displayed. We changed these values by mapping them to the corresponding edge of the stimulus. We could have also deleted these data points, but that would result in non-logical gaze plots, and non-smooth scanpaths. By keeping these data points, we keep the complete scanpath of the user, which



(a) (b)

(c) (d)

Figure 3: A comparison matrix (a) can be ordered (b) to identify outliers. Here we only have one outlier scanpath. All of the scanpaths can be visualized (c) while a special focus on the outlier scanpath (d) may be interesting.

makes an analysis based on a visual attention map, and the comparison between scanpaths smoother and easier. After all the data preprocessing, the final dataset has a size of 8.333KB causing no scalability challenges for our tool.

4.2 Finding the Outliers

For the application example, we use the combination of stimulus and the users as seen in Figure 3. The corresponding comparison matrix is built by applying the Jaccard index to a grid-based AOI subdivision of 20 times 20 pixels to the scanpaths from 19 participants that took part in the eye tracking study solving route finding tasks for this public transport map (shown in Figure 3 (a)). This comparison matrix immediately suggests similarity between the different scanpaths. This gets even more obvious if we apply Robinsonian reordering. The matrix after reordering is depicted in Figure 3 (b).

We see that the participant in row 4 and column 16 in Figure 3 (a) does not behave like the other persons do. To see more details we can take a look at the gaze plot (see Figure 3 (c)) while this is very crowded and produces a lot of visual clutter, hence we could interactively isolate this participant from the gaze plot by using the comparison matrix as interaction medium (see Figure 3 (d)).

As we can see, this is really a strange behavior. The participant never takes a look at the target and focuses out of the map several times. For a researcher, this might be really valuable information to consider during analysis. This example shows how we can interpret the results provided by our visualization tool and how we see anomalies in the data really quickly while setting them in context to other visualizations and the stimulus.

5 DISCUSSION AND LIMITATIONS

Our approach also has some limitations. One limitation is that it is not possible to select specific users in the heatmap due to the aggregation effect. It is possible to show the heatmap for as many users as you want (so also for one specific user), but when multiple users are selected it is not possible to select a specific user.

Our tool also has some visual scalability issues. The user of the tool needs to scroll down if too many scanpaths are compared, i.e., if the corresponding comparison matrix gets too large, consisting of too many rows and columns. Also, the figures are constantly generated, even after very small changes, meaning these do not change the visualizations in an instant.

There were also some algorithmic limitations. If we computed similarity values while taking the order of the fixation points into account, we would get very low similarity values, almost unusable. Also, the running time of our visualizations can be problematic for really many comparisons resulting in large matrices. This has also an impact on the interaction techniques and their responsiveness.

From a perceptual perspective it is important to not use that many colors and to not overdraw too many scanpaths if a gazeplot is used as an additional view on the scanpath data. Moreover, if the comparison matrices are too large and scaled down to pixel size, it might be important to apply an appropriate color coding to the pixels to still be able to identify the outliers.

6 CONCLUSION AND FUTURE WORK

In this paper we described an interactive visualization tool based on scanpath comparisons with the goal to detect outliers in scanpath data. To reach this goal we first compared the scanpaths looking for similar properties like overlaps of bounding boxes or common fixations in both scanpaths, further reduced into percentage numbers by applying the Jaccard coefficient. The comparison values are depicted in a color coded adjacency matrix that is reordered to better identify outliers. Additional views like visual attention maps and gaze plots show contextual information as well as the visual stimulus. For future work we plan to add more comparison techniques while also further matrix reordering techniques [Behrisch et al. 2016] might be applicable. Also a user evaluation could give further insights into the usefulness of our visualization tool. This could also be done by an eye tracking experiment, using our tool to identify outliers in the scanpath data. Apart from adjacency matrix visualization, we will plan to experiment with node-link diagrams in different layouts to show the pairwise comparisons. This may help to identify clusters and as in the case of this research, also the outlier scanpaths.

ACKNOWLEDGMENTS

This research was partially supported by NSF grant IIS 1527200 and MSIT, Korea, under the ICT Consilience Creative program (IITP-2019-H8601-15-1011) supervised by the IITP. We would like to thank Robin Woods from Communicarta Ltd. to provide the public transport maps for our eye tracking study.

REFERENCES

Gennady Andrienko, Natalia Andrienko, Michael Burch, and Daniel Weiskopf. 2012. Visual Analytics Methodology for Eye Movement Studies. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (2012), 2889–2898.

Michael Behrisch, Benjamin Bach, Nathalie Henry Riche, Tobias Schreck, and Jean-Daniel Fekete. 2016. Matrix Reordering Methods for Table and Network Visualization. *Computer Graphics Forum* 35, 3 (2016), 693–716.

Tanja Blascheck, Michael Burch, Michael Raschke, and Daniel Weiskopf. 2015. Challenges and Perspectives in Big Eye-Movement Data Visual Analytics. In *Proceedings of the 1st International Symposium on Big Data Visual Analytics*. 17–24.

Tanja Blascheck, Kuno Kurzhals, Michael Raschke, Michael Burch, Daniel Weiskopf, and Thomas Ertl. 2017. Visualization of Eye Tracking Data: A Taxonomy and Survey. *Computer Graphics Forum* 36, 8 (2017), 260–284.

Agnieszka Bojko. 2009. Informative or Misleading? Heatmaps Deconstructed. In *Human-Computer Interaction – INTERACT*. Springer, 30–39.

Michael Burch. 2016. Time-Preserving Visual Attention Maps. In *Proceedings of Conference on Intelligent Decision Technologies*. 273–283.

Michael Burch. 2018. Identifying Similar Eye Movement Patterns with t-SNE. In *Proceedings of Symposium on Vision, Modeling, and Visualization*. 111–118.

Michael Burch, Ayush Kumar, and Klaus Mueller. 2018a. The hierarchical flow of eye movements. In *Proceedings of the 3rd Workshop on Eye Tracking and Visualization, ETVIS@ETRA*. 1–5.

Michael Burch, Kuno Kurzhals, Niklas Kleinhans, and Daniel Weiskopf. 2018b. EyeMSA: exploring eye movement data with pairwise and multiple sequence alignment. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications, ETRA*. 1–5.

Andrew T. Duchowski. 2003. *Eye Tracking Methodology - Theory and Practice*. Springer.

Joseph H. Goldberg and Jonathan I. Helfman. 2010. Visual scanpath representation. In *Proceedings of the Symposium on Eye-Tracking Research and Applications (ETRA)*. 203–210.

Christopher G. Healey and James T. Enns. 2012. Attention and Visual Memory in Visualization and Computer Graphics. *IEEE Transactions on Visualization and Computer Graphics* 18, 7 (2012), 1170–1188.

Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost van de Weijer. 2011. *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford University Press.

Daniel A. Keim. 2012. Solving Problems with Visual Analytics: Challenges and Applications. In *Proceedings of Machine Learning and Knowledge Discovery in Databases - European Conference*. 5–6.

Ayush Kumar, Michael Burch, Isaiah Brand, Laurens Castelijns, Frank Ritchi, Freek Rooks, Neil Timmermans, and Klaus Mueller. 2018a. Eye Tracking for Exploring Visual Communication Differences.

Ayush Kumar, Rudolf Netzel, Michael Burch, Daniel Weiskopf, and Klaus Mueller. 2018b. Visual Multi-Metric Grouping of Eye-Tracking Data. *Journal of Eye Movement Research* 10, 5 (2018), 247–262. https://bop.unibe.ch/JEMR/article/view/4225

Kuno Kurzhals, Brian D. Fisher, Michael Burch, and Daniel Weiskopf. 2014. Evaluating Visual Analytics with Eye Tracking. In *Proceedings of the Fifth Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization, BELIV*. 61–69.

Kuno Kurzhals, Marcel Hlawatsch, Florian Heimerl, Michael Burch, Thomas Ertl, and Daniel Weiskopf. 2016. Gaze Stripes: Image-Based Visualization of Eye Tracking Data. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (2016), 1005–1014.

Rudolf Netzel, Bettina Ohlhausen, Kuno Kurzhals, Robin Woods, Michael Burch, and Daniel Weiskopf. 2017. User performance and reading strategies for metro maps: An eye tracking study. *Spatial Cognition & Computation* 17, 1–2 (2017), 39–64.

Ruth Rosenholtz, Yuanzhen Li, Jonathan Mansfield, and Zhenlan Jin. 2005. Feature Congestion: A Measure of Display Clutter. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 761–770.

Colin Ware. 2004. *Information Visualization: Perception for Design*. Morgan Kaufmann.

Colin Ware. 2008. *Visual Thinking: for Design*. Morgan Kaufmann Series in Interactive Technologies, Paperback.

Alfred L. Yarbus. 1967. *Eye Movements and Vision (Translated from Russian by Basil Haigh. Original Russian edition published in Moscow in 1965.)*. New York: Plenum Press.