Eye Tracking for Exploring Visual Communication Differences

Ayush Kumar*
Stony Brook University, New York, USA

Michael Burch†
Eindhoven University of Technology

Irma van den Brand
Eindhoven University of Technology

Laurens Castelijns
Eindhoven University of Technology

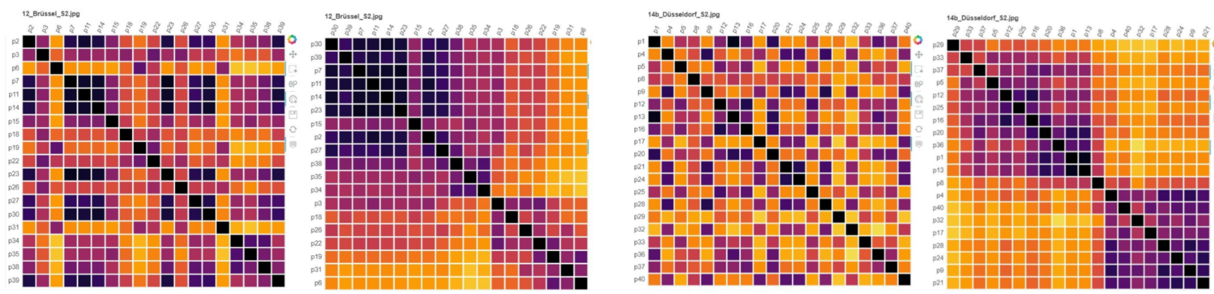
Frank Ritchi
Eindhoven University of Technology

Freek Rooks
Eindhoven University of Technology

Hero de Smeth
Eindhoven University of Technology

Neil Timmermans
Eindhoven University of Technology

Klaus Mueller‡
Stony Brook University, New York, USA



(a) (b)

Figure 1: Color coded adjacency matrices depicting the strengths of pairwise scanpath comparisons: The initial similarity matrix is further ordered by (a) dimensionality reduction or (b) spectral approaches.

Abstract

Interpreting and understanding visualizations can become a challenging task. Moreover, the task solution highly depends on the user experience and hence, can lead to different response times or accuracies. In traditional user experiments these dependent variables are valuable statistics in order to evaluate the strengths or weaknesses of how the visual stimulus communicates the contained information. However, on the negative side, these values do not tell anything about the solution strategies over space and time. Eye tracking is a technology suitable to record people's eye movements while they try to answer a given task, i.e., while the visualization communicates information or not and to what extent. In this paper we introduce eye movement comparison approaches depicted in ordered adjacency matrices to explore the visual communication differences of people while they solved route finding tasks in public transport maps.

Index Terms: Human-centered computing—Visualization—Visualization techniques; Human-centered computing—Visualization—Visualization design and evaluation methods

1 Introduction

Visual communication can be very different depending on the task that has to be answered [22] when people inspect a visual stimulus or scene. However, this phenomenon is known for many years, but due to the growing amount of spatio-temporal eye movement data [3] exploring such visual communication patterns is a challenging and tedious task.

For example, identifying similar eye movement behavior among a group of people [7] is difficult due to the fact that the eye movement data consists of several dimensions [1] while also several metrics might be derived for the similarity identification [11, 13]. Algorithmically and visually supporting the identification process is a good strategy since the data analysts might have a look at the eye movement data from several perspectives, deciding which one is suited best for their demands.

In this paper we describe the ordered eye movement similarity matrices that first algorithmically compare scanpaths on several comparison strategies, visualize the pairwise comparison values in a color coded adjacency matrix, and finally, allow several matrix reordering techniques in order to enhance the visual appearance of similarity groups. Complementary and linked views for eye movement data like visual attention maps or gaze plots support the identification of the found patterns in the spatial stimuli. Interactions can be applied to filter, navigate, browse, or aggregate the data while a repertoire of scanpath comparison techniques like bounding box-based or Jaccard coefficient-based ones and matrix reordering strategies [2] like dimensionality reduction, spectral approaches, or clustering and some more are provided.

We illustrate the usefulness of the approach by applying it to real-world eye movement data recorded in an experiment in which people had to find routes in public transport maps. In these scenarios the public transport maps are used to communicate information, e.g., how passengers can best plan their journeys in a foreign city. Hence, the role of visual communication is very important to understand if the designed metro maps are useful or not and scanpath comparison of various participants give an insight into it.

2 Related Work

There are lots of visualization techniques for eye movement data [4]. However, only a few concentrate on comparing eye movements among eye tracking study participants with the goal to detect similar visual communication patterns. For example, understanding communication patterns while data analysts explore different dynamic graph visualizations [8–10] can help to figure out which representation is performing better or worse than the others.

For example, visual attention maps [5, 6, 21] might be useful to see the differences and similarities of visual attention hot spots, but

*e-mail: aykumar@cs.stonybrook.edu
†e-mail: m.burch@tue.nl
‡e-mail: mueller@cs.stonybrook.edu

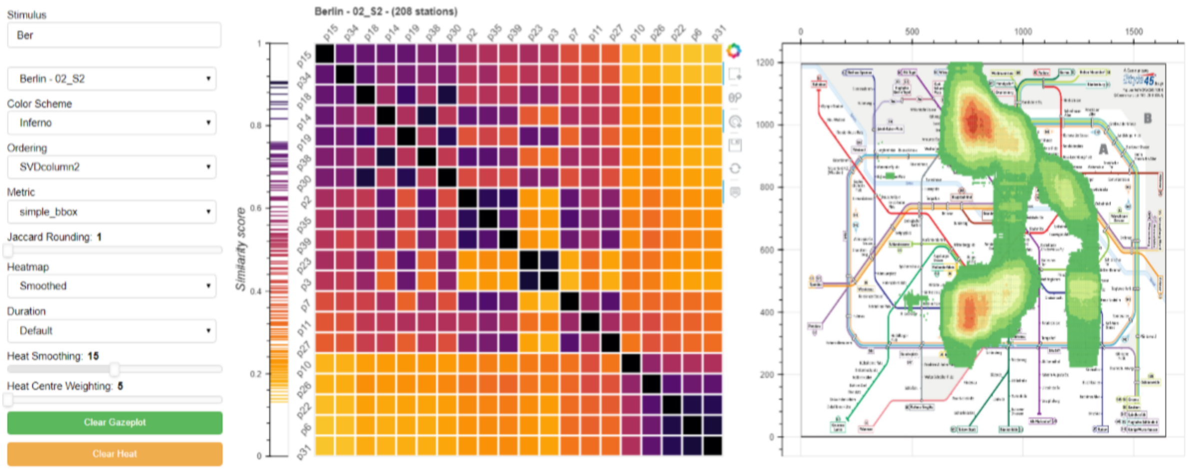


Figure 2: The graphical user interface consists of three views while those are interactive and linked: (a) The input panel. (b) The adjacency matrix panel. (c) The stimulus panel with visual attention maps and gaze plots.

the negative effect is that those heatmaps do not show the pairwise similarities between all the study participants. If those were drawn for each participant individually, the human observer would have problems in identifying different or similar visual patterns or even groups of people doing similar scanning strategies.

Using gaze plots instead to show individual eye movement scanpaths overdrawn on a visual stimulus brings new problems into play. If the scanpaths are temporally long and many people got eye tracked, those gaze plots [12] run into problems caused by visual clutter [20]. The visual communication patterns can consequently not be observed since the visual communication of the visual representation of the eye movements is not readable any more.

Andrienko et al. [1] survey visual analytics methodologies for eye movement data. Although many approaches exist already, there is (to the best of our knowledge) no visualization based on ordered adjacency matrices for comparing similarities between eye movement scanpaths. The work of Kumar et al. [14, 15], however, focuses on metric-based grouping of eye movements, but not on comparing the scanpaths by different options. Moreover, the ordering of the participants in the matrices is not computed in several ways as in our technique. Most of the existing approaches provide views on stacked scanpaths [7, 19] while sometimes the stacking order is computed by clustering techniques [16, 17].

In this paper, we propose a method for first comparing the eye movements by similarity measures in a pairwise manner, visually encode those comparison values in an adjacency matrix, reorder the matrix rows and columns, and provide interactions to link the similarity values with the actual stimulus and extra visual representations like visual attention maps and gaze plots.

3 Eye Movement Comparison and Visualization

Our visualization tool consists of several components for exploring eye movements for similar or dissimilar visual communication patterns. To reach our goal we have to first compare eye movements, visualize those comparisons, order them, and finally allow interactions and linkings to the original visual stimuli.

3.1 Design Criteria

Based on the above-mentioned objectives, five requirements for the tool have been formulated, after which it was assessed which visualization techniques were deemed necessary and/or most appropriate to meet these requirements:

- **Selecting stimuli and participants:** The user should be able to select the stimulus and also the participants.
- **Scanpath comparison:** The user should be able to compare two or more scanpaths.
- **Interactivity:** The tool should be interactive to give the user as much control as possible.
- **Clear and relevant output:** The output should be clear and relevant and help to discover new insights and to understand visual communication patterns.
- **Adaptability:** The user should be able to adapt the tool to their wishes.

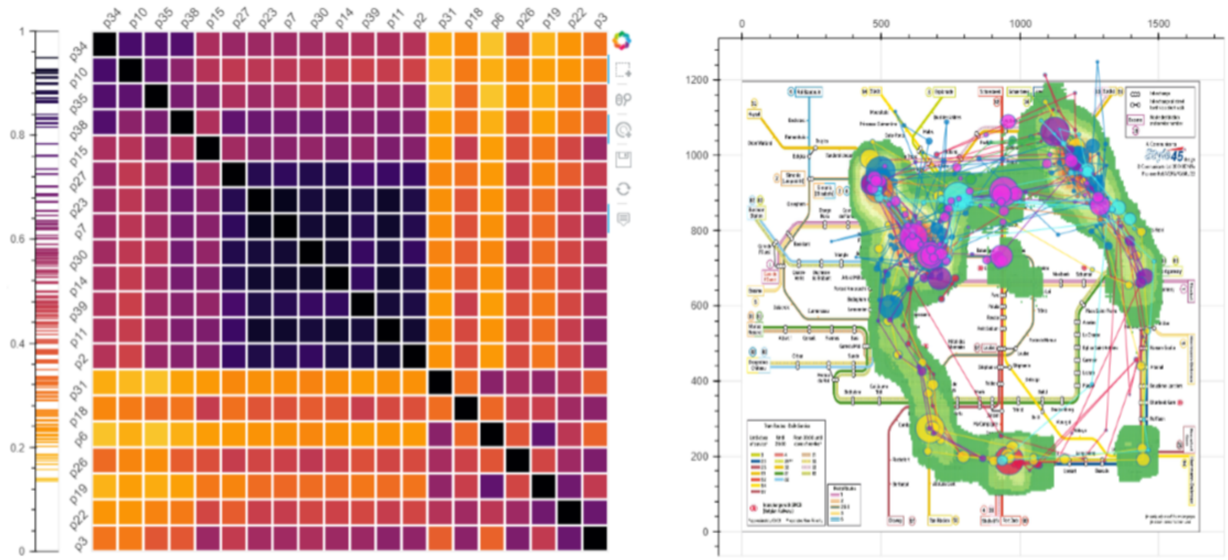
The graphical user interface of our visualization tool can be seen in Figure 2. It consists of three major views which are the input panel, the adjacency matrix panel with extra similarity value distributions, and the stimulus panel overdrawn with a visual attention map and a gaze plot.

3.2 Algorithmic and Visual Components

The settings to adjust the plots and transform the data can be found in the leftmost user input panel. The user can either select a stimulus or type in the first three letters of the city. The similarity values in the adjacency matrix for the selected stimulus will appear in the center view, and the metro map itself is given in the rightmost view.

Each scanpath from every user who has inspected the stimulus will be plotted on the map. In the background of the gaze plot, the visual attention map will be depicted to avoid the overdrawing of the lines. There are seven color schemes that can be selected. Depending on the input, some color schemes might be more suitable to clearly see some differences in similarity values (in the figures the color scheme 'inferno' is selected).

Beneath this setting one can select which ordering algorithm will be applied to the data. To this end there are seven algorithms in total based on the survey by Behrisch et al. [2]. The similarity results can be computed by either the bounding box method or the Jaccard similarity measurement. They are both quite simple and hence not very accurate, but do give some valuable insights in similarities between users. Since the Jaccard similarity values are computed by using the exact points from the data, the similarity values are almost zero (the chance of two users having the exact same fixation points



(a) (b)

Figure 3: Visualizations of the eye movement data: (a) An ordered and color coded adjacency matrix with additional information on the similarity value distribution. (b) The visual stimulus overdrawn with a visual attention map and a gaze plot.

is really low). To still make the Jaccard similarity measurement useful, the user can round the fixation points stepwise with step 5. The maximum rounding value is 50, in this case the similarity will be (too) high.

The last settings in the input box concern the heat map, the user can choose to either have the default heat map shown or the smoothened one. The smoothening of the heat map can also be chosen stepwise with step 5 and maximum value 50. This slider can give some valuable insights and makes it possible for the user to interactively choose the transformations done to the data. The last slider concerns the center weighting for the heat in the heat map. This slider has the same options as the previous ones (stepwise with step 5). Then there is also a color bar between the input box and adjacency matrix. This is a multi-purpose bar that serves as a legend, but one can also see the distribution of the similarity scores. Users can also select in-range from here, the corresponding squares in the adjacency matrix will be highlighted upon selecting similarity scores in a user defined range. By implementing all these techniques and interactivity, the user can easily explore the data. After completing a case or only wanting to see the gaze plot of the visual attention map, the user can choose to have the gaze plot or heat map (or both) cleared.

3.3 Implemented Features

Our visualization tool provides several features to this end. We are aware of the fact that this is still work-in-progress and plan to add many more of them in future to make it a useful tool for understanding visual communication patterns based on visual stimuli and eye movement patterns.

- **Adjacency matrix:** An adjacency matrix that displays the similarity between different scanpaths (see Figure 3 (a))
- **Selecting (subsets of) users:** Users can select two or more participants to compare scanpaths
- **Similarity metrics:** Two similarity functions (Jaccard and Bounding Box) to compare scanpaths
- **Matrix reordering:** The matrix can be reordered in several ways to help discover patterns.
- **Color scheme selection:** Users can choose between different color schemes for the adjacency matrix
- **Hovering:** Hovering over cells in the matrix shows the similarity value represented in these cells and the participant numbers
- **Zoomable gaze plot:** A zoomable map that displays the selected scanpaths over the stimulus
- **Smoothed heatmap:** A smoothed heatmap that shows where and how long participants fixated on (see Figure 3 (b))
- **Adaptable code and input data:** The tool supports that the code can be adapted for other input data

4 Application Example

The objective of the developed tool is to extract as much as possible relevant information from an eye-tracking dataset. This dataset contains data of the scanpaths of 40 participants whose goal was to find the most optimal route between two given points on different public transportation maps [18].

In total, these participants performed the task on 96 maps (stimuli) of 24 cities (2 routes per city, both in color and black/white). In total, the dataset contains over 118,000 fixation points. Gaining insight into such a dataset can be a complex and time-consuming process. By making it easy and intuitive to compare different (subsets of) users and stimuli, this process is simplified. We describe how the tool can show its strengths by mimicking two use cases in which we show similar and dissimilar visual communication patterns among eye tracking study participants.

4.1 Paris Metro Map

When looking at the ordered matrix in Figure 4 we can clearly see different clusters of scan paths, once they are seen with the gaze plot we observe they are different paths taken by users. This can be derived by these clusters having much resemblance with each

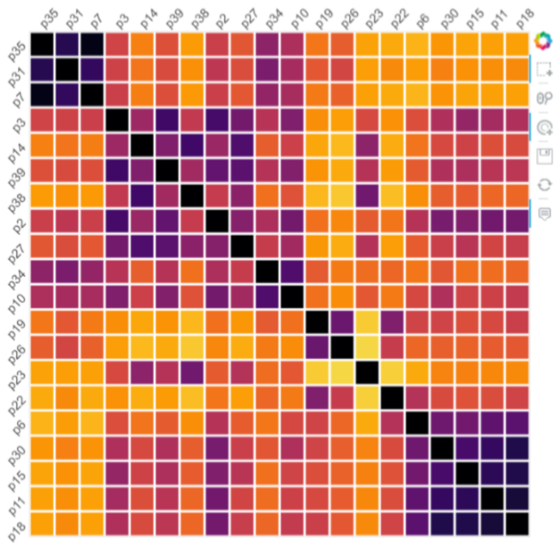


Figure 4: Comparing the similarities of eye movements when people answered route finding tasks in the Paris metro map. The similarity values are computed by a bounding box approach, color coded in an adjacency matrix, and ordered by a dimensionality reduction technique.

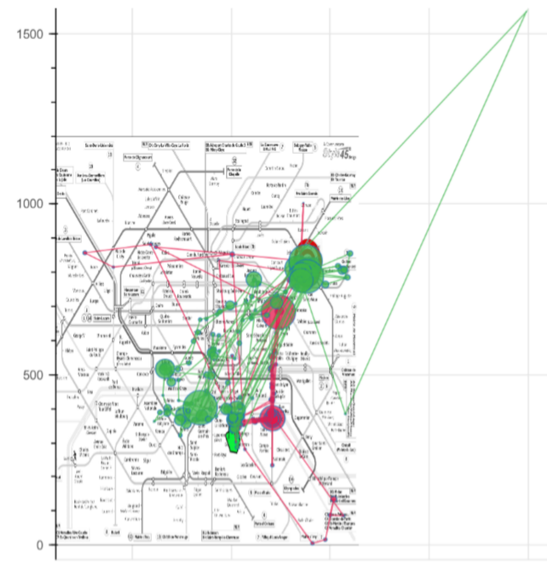


Figure 5: Sometimes the eye movements leave the visual stimulus, indicating a certain kind of problem or a distraction of the human observer.

other and not the other clusters, indicating that there is some sort of similarity between these groups (perhaps a choice of two or more different routes). However, this can depend on the ordering and similarity algorithms used but can be confirmed by selecting the different clusters and seeing them on the gaze plot while comparing the two (or more) groups and observing their differences.

The matrix can also be used to see outliers, and here in the case of the Paris map (see Figure 5) we can see two clear outliers with one of the users being distracted by looking outside of the metro map (green) and the other looking at the legend (red), this could give important feedback to the experiment and the map creators. Hence, the visual communication patterns when inspecting this metro map vary significantly.

4.2 Brussels Metro Map

A heat map such as in Figure 6 (a) shows the point on the metro map with the highest attention, these could be areas of interest (start, finish, other important/interesting points), or show potential bottlenecks where the users struggle to go through. Also, we can see when and where the paths split, and with adjusting the weights we can see where people just glance over and where users stop and loop at the stimulus giving it more attention (using the eye mind hypothesis we could even derive that there is some struggle in this area).

In the Brussels example we can clearly see two distinct routes. In Figure 6 (b) we adjust the heatmap for the fixation duration and we can see that some parts have less attention compared to the normal heatmap.

5 Discussion and Limitations

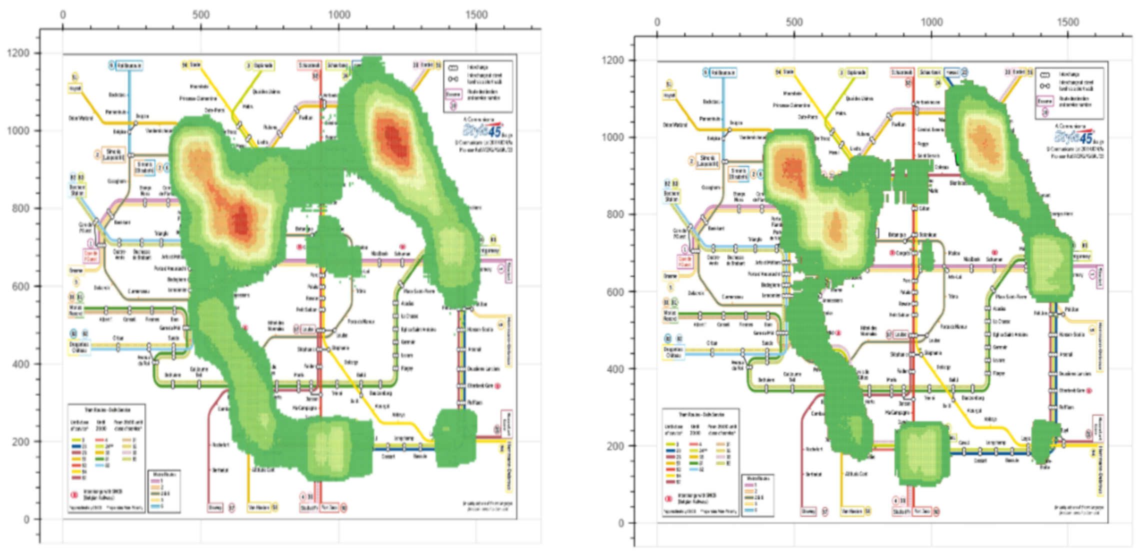
Indefinite selection and refinement within a single session allow the user to navigate through the provided data at any wished specificity. When the application is confronted with input of arbitrary size, this is no longer the case. Some of the described functionalities are suspect to scalability issues.

The tool is a web application, but not all of these issues can be solved by increasing the capacity of the server it is running on. A lot of tasks are performed by the user's browser, and rely on local (more limited) computational resources. For example, on an average notebook, plotting a too fine-grained heatmap will considerably decrease the applications' responsiveness. Even with the current data, plotting just the fixation points of a single stimulus could prove to be too demanding for a high-end smartphone. In such a situation the application would do well to decrease the interactivity of the plotted points, or withdraw other bells and whistles.

Note that if just the initial representation of inputs would be well-balanced with the available resources, these scalability issues are not severe as they may seem. The general structure of the tool is well-suited for both top-down and bottom-up research strategies. A more experienced user might navigate through large input by avoiding too demanding selections and parameter settings. Letting the availability of certain functionality depend on the available resources would greatly increase the scope of the tool.

6 Conclusion

We have created a tool in which a user can select a stimulus, compare scan paths while the comparison results are depicted both in an adjacency matrix and visually on a map with connected paths and/or a heatmap. The matrices can be ordered to find visual communication patterns, select and view a subset of users, apply different similarity metrics, and also adjust parameters in the heatmap. The objective was to create a tool to visually extract information from an eye-tracking dataset. The user has full control over the parameters, giving freedom to find patterns and information in ways we might not even expect. Developers are focusing their attention more and more towards web apps instead of native apps. Web apps can be run on the server from almost any browser, are not OS-dependent, and are becoming more and more powerful. This is why we have decided to create the tool as a web application. But the tool is not perfect at the moment, hence, various points remain for future work. There are several ways that could improve its value a lot, giving the user more options to visually extract information from any eye-tracking data set. One way is to simply add an extra box which gives additional descriptive statistics, because sometimes numbers say more than pictures. Secondly, it would be nice to be able to compare two maps against each other, and to select subsets of users and paths within both. Thirdly, implementing a time-aware metric would open up an



(a) (b)

Figure 6: The visual attention map for the metro map of Brussels in Belgium (a) without the fixation duration information and (b) with the fixation duration information.

extra dimension to the data. And finally, implementing a way for the users to upload their own data would be a future option to extend the tool.

ACKNOWLEDGMENTS

This research was supported by NSF grant IIS 1527200 & MSIT (Ministry of Science and ICT), Korea, under the ICTCCP (IITP-2017-R0346-16-1007) supervised by the IITP.

REFERENCES

[1] G. Andrienko, N. Andrienko, M. Burch, and D. Weiskopf. Visual analytics methodology for eye movement studies. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2889–2898, 2012.

[2] M. Behrisch, B. Bach, N. H. Riche, T. Schreck, and J. Fekete. Matrix reordering methods for table and network visualization. *Computer Graphics Forum*, 35(3):693–716, 2016.

[3] T. Blascheck, M. Burch, M. Raschke, and D. Weiskopf. Challenges and perspectives in big eye-movement data visual analytics. In *Proceedings of the 1st International Symposium on Big Data Visual Analytics*, pp. 17–24, 2015.

[4] T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl. Visualization of eye tracking data: A taxonomy and survey. *Computer Graphics Forum*, 2017.

[5] A. Bojko. Informative or misleading? Heatmaps deconstructed. In *Human-Computer Interaction – INTERACT*, pp. 30–39. Springer, 2009.

[6] M. Burch. Time-preserving visual attention maps. In *Proceedings of Conference on Intelligent Decision Technologies*, pp. 273–283, 2016.

[7] M. Burch, G. L. Andrienko, N. V. Andrienko, M. Höferlin, M. Raschke, and D. Weiskopf. Visual task solution strategies in tree diagrams. In *Proceedings of IEEE Pacific Visualization Symposium*, pp. 169–176, 2013.

[8] M. Burch, F. Beck, and D. Weiskopf. Radial edge splatting for visualizing dynamic directed graphs. In *Proceedings of the International Conference on Computer Graphics Theory and Applications (IVAPP)*, pp. 603–612, 2012.

[9] M. Burch, M. Fritz, F. Beck, and S. Diehl. Timespidertrees: A novel visual metaphor for dynamic compound graphs. In *Proceedings of the IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC*, pp. 168–175, 2010.

[10] M. Burch, B. Schmidt, and D. Weiskopf. A matrix-based visualization for exploring dynamic compound digraphs. In *Proceedings of the International Conference on Information Visualisation, IV*, pp. 66–73, 2013.

[11] A. T. Duchowski. *Eye Tracking Methodology - Theory and Practice*. Springer, 2003.

[12] J. H. Goldberg and J. I. Helfman. Visual scanpath representation. In *Proceedings of the Symposium on Eye-Tracking Research and Applications (ETRA)*, pp. 203–210, 2010.

[13] K. Holmqvist, M. Nyström, R. Andersson, R. Dewhurst, H. Jarodzka, and J. van de Weijer. *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford University Press, 2011.

[14] A. Kumar, R. Netzel, M. Burch, D. Weiskopf, and K. Mueller. Visual multi-metric grouping of eye tracking data. *Journal on Eye Movement Research*, 2018.

[15] A. Kumar, R. Netzel, M. Burch, D. Weiskopf, and K. Mueller2. Multi-similarity matrices of eye movement data. In *2016 IEEE Second Workshop on Eye Tracking and Visualization (ETVIS)*, pp. 26–30, Oct 2016. doi: 10.1109/ETVIS.2016.7851161

[16] K. Kurzhals, M. Hlawatsch, M. Burch, and D. Weiskopf. Fixation-image charts. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications, ETRA*, pp. 11–18, 2016.

[17] K. Kurzhals, M. Hlawatsch, F. Heimerl, M. Burch, T. Ertl, and D. Weiskopf. Gaze stripes: Image-based visualization of eye tracking data. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):1005–1014, 2016.

[18] R. Netzel, B. Ohlhausen, K. Kurzhals, R. Woods, M. Burch, and D. Weiskopf. User performance and reading strategies for metro maps: An eye tracking study. *Spatial Cognition & Computation*, 17(1–2):39–64, 2017.

[19] M. Raschke, X. Chen, and T. Ertl. Parallel scan-path visualization. In *Proceedings of the 2012 Symposium on Eye-Tracking Research and Applications, ETRA*, pp. 165–168, 2012.

[20] R. Rosenholtz, Y. Li, J. Mansfield, and Z. Jin. Feature congestion: A measure of display clutter. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 761–770. ACM, 2005.

[21] O. Spakov and D. Miniotas. Visualization of eye gaze data using heat maps. *Electronics and Electrical Engineering*, 2(74):55–58, 2007.

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