A Tool to Simplify the Visualization of Eye Tracking Data

Eric Abraham, Lukas Dobbelsteen, Milou Henzen, Chiara Liotta, Juul Peters, and Tristan Tummers

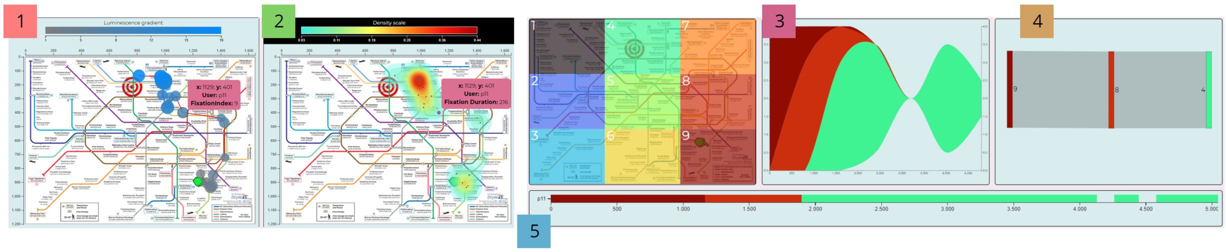


Fig. 1: A linked visualization offered by the webtool featuring a scanpath (1), a heatmap (2), a ThemeRiver (3), a Sankey diagram (4), and a scarf plot (5). The figures visualize eye tracking data over the selected stimulus: a colored metro map of Moscow. A specific study subject (p11) was chosen and the point-based visualization linking between the dots representing the fixation points is shown.

Abstract—The popularity of eye tracking for the analysis of human viewing behavior has seen a remarkable rise over the last years, leading to an increase in applications in numerous fields of research. Data from these eye tracking experiments can be analyzed in order to gain insight into its internal structure and discover potential causal relationships present in it. This paper describes an approach to eye tracking data analysis that uses various linked visualization techniques, each having its own functionalities. These visualizations will be accessible using a web-based tool; therefore, unlike many others, they will not require any programming experience or downloadable software. Furthermore, the data and results of a visualization can be easily shared with other people working on the same topic. To exemplify the practicality of this web-based visualization tool, it will be applied to a formerly conducted real-world eye tracking experiment.

1 Introduction

The use of eye tracking technology has become more widespread as it is becoming cheaper and easier to use. Numerous fields of study, such as marketing, neuroscience, and human-computer interaction [8], implement eye tracking technology in order to analyze human viewing behavior. One of the advantages of an eye tracking analysis over other methods, e.g. the mere comparison of completion times, is represented by its ability to provide additional information on how visual attention is distributed overall and how it changes per study subject for a presented stimulus. Eye tracking devices record gaze points indicating where a participant is looking on a stimulus and then store them as raw data. This data can be further processed into, among other things, fixations and saccades.

The processed data can be used by visualization tools in order to create visualizations that can, for example, uncover certain relationships within the data. Due to the growing use of eye tracking technologies, the demand for these visualization tools has increased as well [13]. However, we noticed that programming experience or downloadable software is often required in order to fully benefit from the already existing ones. This requirement might pose a problem for researchers in fields of study where this experience is not common, such as marketing.

In this paper, we introduce a web-based visualization tool that allows eye tracking data to be effortlessly visualized by anyone. The tool provides five visualization techniques that offer several insights into a dataset. Those can be easily uploaded, but they need to conform to a predefined format. Since it is a web-based environment, users can share their data and results with colleagues without difficulty.

- *Eric Abraham, TU Eindhoven, e.abraham@student.tue.nl.*
- *Lukas Dobbelsteen, TU Eindhoven, l.a.a.dobbelsteen@student.tue.nl.*
- *Milou Henzen, TU Eindhoven, m.m.l.s.henzen@student.tue.nl.*
- *Chiara Liotta, TU Eindhoven, c.liotta@student.tue.nl.*
- *Juul Peters, TU Eindhoven, j.l.peters@student.tue.nl.*
- *Tristan Tummers, TU Eindhoven, t.tummers@student.tue.nl.*

We illustrate the functionalities of our visualization tool by applying it to the dataset of a real-world eye tracking experiment conducted in 2016, where participants were presented with route-finding tasks in public transport maps [16].

2 Related Work

After conducting an eye tracking experiment, an easy-to-use but detailed analysis software is needed to gain insights into the data collected. This tool should not only provide the data analysts with user-friendly upload and managing functionalities, but also with interactive visualization techniques. Heatmaps and scanpaths are two of the most used types of visualizations. However, both solutions only allow a limited and one-sided analysis of the data, which is why several distinct perspectives on the same dataset should be available, and linked coordinated views should be offered [19]. Our visualization tool represents an attempt at achieving just that - a multiplicity of connected views and different angles on the data.

Scanpaths offer a good understanding of how study subjects interact with a given stimulus [10], however, this type of visualization can quickly produce clutter if several different scanpaths are shown together [1]. Such a visual clutter problem is exemplified in Figure 3.

Heatmaps provide an intuitive overview of the points that grabbed the study subjects' attention the most [3], but they do not come without disadvantages [2].

Other visualizations that provide different angles of analysis are represented by ThemeRivers, scarfplots, and Sankey diagrams. ThemeRivers allow users to get an overview of the distribution of attention over time per area of interest [15], but fail to show the hierarchical flow of the eye movements. Scarfplots offer the same paradigm, but allow study subjects to be compared individually. Sankey diagrams are a powerful method of visualizing the relationship between different data attributes. In this case they focus on the flow of transitions between different areas of interest (AOIs); however, they can quickly produce visual clutter if an excessive number of areas of interest are displayed [5].

By integrating these different visualization techniques, the data can be viewed from several distinct perspectives. This can lead to the un-

	Timestamp [PK] integer	StimuliName [PK] text	FixationIndex [PK] integer	FixationDuration integer	MappedFixationPointX integer	MappedFixationPointY integer	user [PK] text	description text
1	2586	01_Antwerpen_S1...	9	250	1151	458	p1	color
2	2836	01_Antwerpen_S1...	10	150	1371	316	p1	color
3	2986	01_Antwerpen_S1...	11	283	1342	287	p1	color
4	3269	01_Antwerpen_S1...	12	433	762	303	p1	color
5	3702	01_Antwerpen_S1...	13	183	624	297	p1	color
6	3885	01_Antwerpen_S1...	14	333	712	303	p1	color
7	4218	01_Antwerpen_S1...	15	300	753	293	p1	color
8	4518	01_Antwerpen_S1...	16	516	804	284	p1	color

Fig. 2: Example taken from the metro map dataset table with the seven attributes, namely Timestamp, StimuliName, FixationIndex, FixationDuration, MappedFixationPointX, MappedFixationPointY, user, and description. It is to be noted that the last attribute is not a requirement in order to use the web tool.

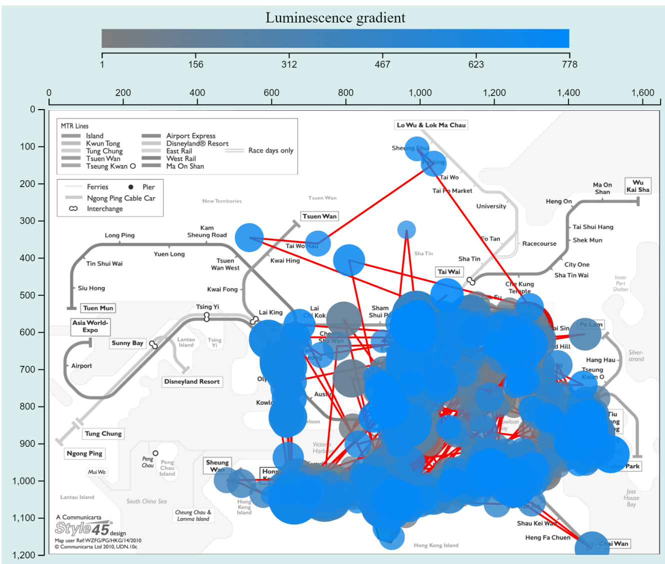


Fig. 3: Example of the visual clutter problem in the scanpath visualization displaying the eye tracking data for all the study participants over the selected stimulus: a grayscale metro map of Hong Kong.

covering of many different insights that can be interactively combined by the human analyst, thereby integrating the analytical capabilities of the computer with the unique human ability to find patterns [14]. Moreover, the decision to make this tool web-based guarantees a higher accessibility level.

3 Data Model

This section will briefly introduce the type of data that is used in the project. Section 4.3 explains in detail how the data is being read and handled.

3.1 Data Type

The following data comes in a '.csv' format, which stands for comma-separated values, however, we are using tab-separated values. The file contains seven attributes, which have the following meaning:

1. **Timestamp**: refers to the time - measured in ms - elapsed since the recording of the data began;
2. **StimuliName**: showcases the specific metro map that the study subject looked at;
3. **FixationIndex**: describes the order of the fixation points;
4. **MappedFixationPointX**: refers to the x-axis coordinate of the point that the study subject looked at;
5. **MappedFixationPointX**: refers to the y-axis coordinate of the point that the study subject looked at;
6. **user**: defines the ID of the study subject;
7. **description**: category that defines if the metro map was in grayscale or in full color.

An example of this dataset is shown in Figure 2.

3.2 Specifications

This subsection will address some data specifications, in order to provide more insight into what the data entries mean.

Firstly, the x and y coordinate attributes lie on a plane whose axes start from the top-left, meaning that the origin can be found in that corner.

It is possible that, on some stimuli, certain data points are found outside of the range of the metro image given, an occurrence can be due to malfunctioning hardware, or the test subjects simply going over the border of the images. Our project does not discard such points. Furthermore, the points whose coordinates have been identified are fixation points, that is, aggregated gaze points on a specified area and time span [1].

The collection of metro maps that were studied are defined as stimuli for the experiment. The list of maps is presented in a folder containing '.jpg', '.png', or '.jpeg' pictures. Moreover, each map comes in a colored version and a grayscale one.

4 Visualization Tool

The purpose of this section is to illustrate how the tool was set up, how it can be used, and what visualization types it supports.

4.1 Graphical User Interface (GUI)

The tool's graphical user interface features three main components that the user can interact with: an introductory section, a navigation menu, and a visualization container. An annotated view of the GUI can be observed in Figure 5.

4.1.1 Introductory Section

The main purpose of this component is to get the users acquainted with the basic functionalities of the tool and to allow them to upload a dataset in the described predetermined format. Moreover, explanations and examples of all the visualizations types are available, in order to allow even inexperienced users to be able to take full advantage of the tool's potential.

4.1.2 Navigation Menu

The drop-down menu, situated on the lower left-hand corner of the header, contains six different collapsible subsections that start out hidden. Furthermore, the menu, if open, moves as the user scrolls up or down in order to guarantee its user-friendliness throughout the whole experience.

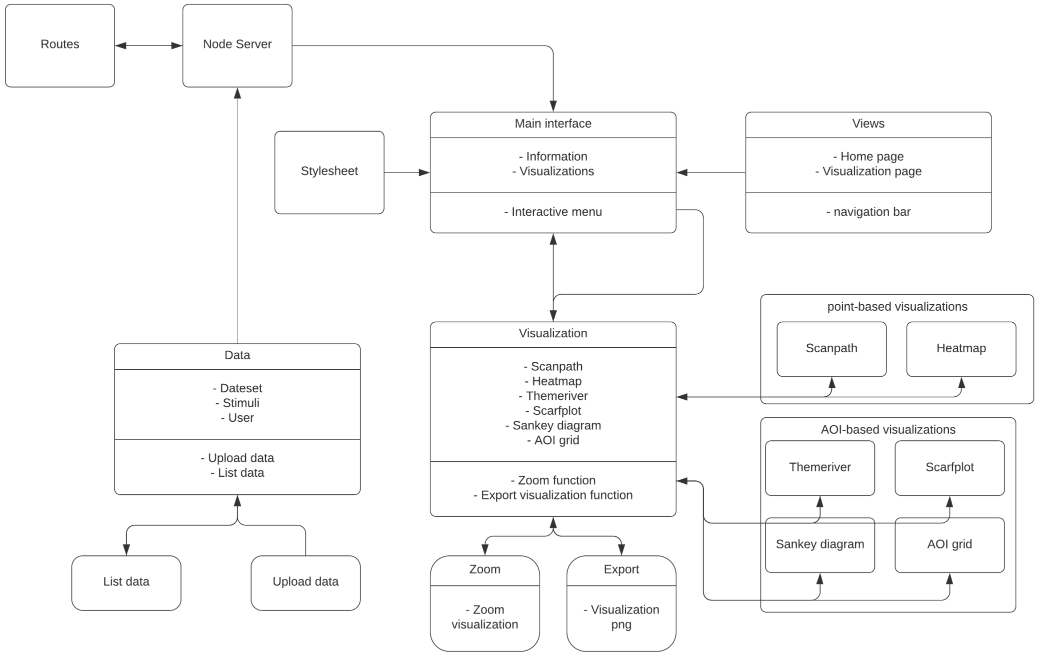


Fig. 4: The Unified Modeling Language diagram (UML diagram) of our visualization tool. This diagram shows all the software components and how they are linked up together.

The first menu subsection allows the user to choose the data and initialize the linked view, the second deals with zoom resetting, the third and fourth present the specific options for scanpaths and heatmaps respectively, the fifth concerns the AOI-based visualizations, while the last one allows the user to export each visualization. The ”Home” button allows the user to go back to the homepage in order to upload a new dataset or read the visualization descriptions.

4.1.3 Visualization Container

At the center of the web page, a container is displayed once the linked visualizations are initialized via an ”Initialize” button under the inputs for the choice of dataset, stimulus, and study participant(s). Since our web tool only features one actual web page, clicking this button hides the elements belonging to the homepage and displays the container with both the point-based and AOI-based visualizations. This type of behavior is achieved through the use of jQuery and allows for faster loading overall. The two point-based visualizations appear next to each other in a grid and are fully linked via zooming and pop-up behavior. The AOI-based visualizations are present underneath them and offer a linked highlighting behavior on hover based on areas of interest.

4.2 Interaction Techniques

This section will present the different interactions implemented in our visualization tool:

- **Re-adjust and resize**: Users are given the freedom to zoom and pan scanpaths, heatmaps, and ThemeRivers. Zooming allows the image to be resized using a mouse wheel or touchpad. Panning lets the user click on the image and drag it across its container. Moreover, ”Reset” buttons are present under the ”Zoom” menu subsection in order to reset them to their original scale. All visualizations also scale with the window.

- **Data filtering**: For each stimulus, the tool is capable of filtering data based on the individuals whose eye movement data was tracked and recorded. This option provides the user with a separate overview for each study subject, with the option to visualize all data at once, without distinguishing between different participants.

- **Hover pop-up**: In all visualizations the user can hover over a fixation point, area of interest, grid, or density contour with the cursor, triggering the relevant information to appear in a pop-up box. This information includes coordinates, users, density, AOI name, timestamp, etc. This feature allows users to uniquely observe and compare all aspects of the visualizations.

- **Color**: Scanpath saccades and fixation points can change color in order to be more accessible - in case of a color vision deficiency - or to provide a clearer overview. The same applies to the heatmap and the AOI-based visualizations, both offering three different color gradients.

- **Coordinate axes**: Scanpaths, heatmaps, and ThemeRivers implement coordinate axes for the points, which can be scaled and translated, thus providing a clear overview of the location of all points or AOIs on the plane. Scarf plots, on the other hand, offer a static set of axes.

- **Sliders**: Both the heatmap and ThemeRiver have sliders to allow several interactive options. The heatmap includes three sliders that allow the user to choose the circle radius, bandwidth (standard deviation), and contour opacity. These sliders can be reset to the original parameters through a ”Reset sliders” button. For the ThemeRiver, the slider adjusts the length of the time intervals.

- **AOI grid size**: The size of the grid that defines the areas of interest can be specified in the grid size selector, with a minimum of a 2x2 grid and a maximum of 6x6.

On top of having a common dataset, stimulus, and study participant menu, the tool also features some linking and brushing related interactions:

- **Point-based visualizations:** For all point-based visualizations, the zooming behavior is completely linked. Moreover, linked pop-ups are present and offer the possibility to easily compare the fixation duration of a determined point in an area of lower or higher attention with its position in the ordered sequence of

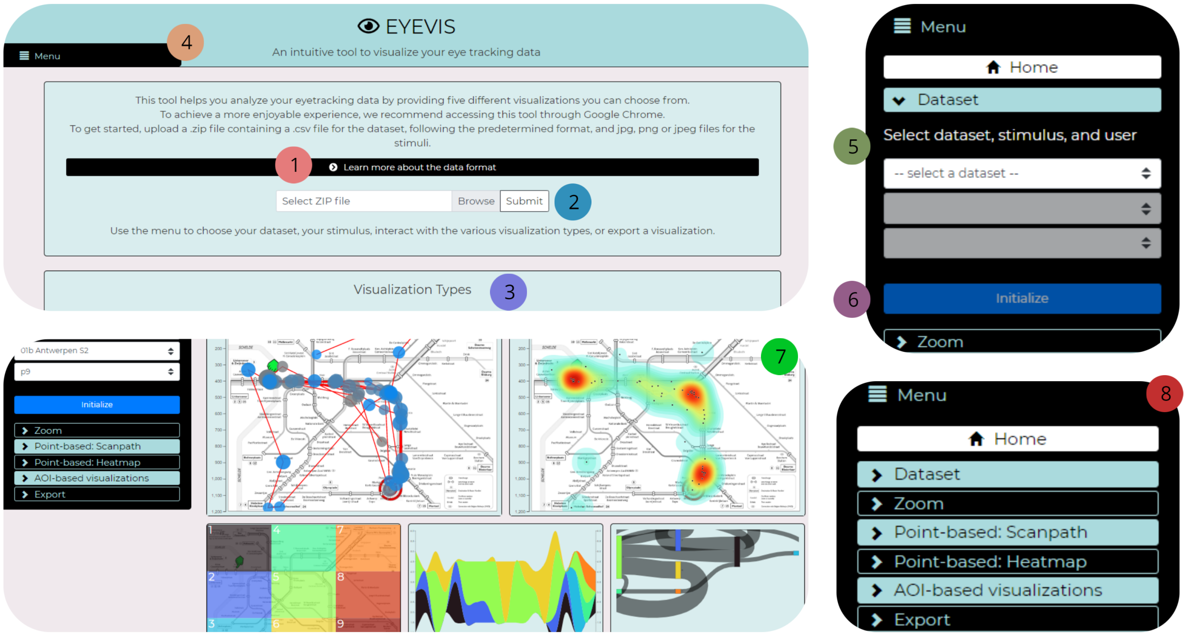


Fig. 5: The GUI layout, including further information about the accepted data format (1), the field to upload a new dataset (2), the description of the visualization types offered (3), the menu (4) from which the user can choose the dataset, stimulus, and study subject(s) (5), the button (6) to initialize the linked view of the visualizations (7), and the drop-down buttons to uncover visualization-specific options (8).

gaze points. This linking and brushing technique also offers a highlight of the point being hovered on in both visualizations and gives a partial solution to the scanpath's clutter by offering the user a way to trigger the pop-ups of potentially covered circles.

- **AOI-based visualizations:** All three visualizations respond to the same grid size selector and are altered accordingly. There are three common color gradients to choose from that apply to all AOI-based visualizations. Hovering over a visualization, or over the color grid, will highlight the hovered-over area of interest in all visualizations and in the color grid. This feature is particularly useful to distinguish possibly similarly-colored areas.

All these interactions combined provide users with enough options to customize and analyze the visualizations, with the purpose of offering a more thorough understanding of the data.

4.3 Implementation Details

The programming language used for the tool is JavaScript. This language is used for both the back-end and the front-end, thanks to Node.js. This runtime environment allows JavaScript code to be run outside of a web browser, where it is normally used, thereby eliminating the need for different languages for server- and client-side scripts. Using a single language simplifies development and increases the readability of the code. Node.js, often abbreviated to Node, has a rich repository of packages built by developers, which allows quick development while guaranteeing reliability and stability. Node supports asynchronous/non-blocking programming as well, allowing the server to remain responsive while handling multiple users. These advantages outweigh those of the alternatives, namely PHP and Python frameworks.

The Node back-end allows for uploading '.zip' files. These files should contain a '.csv' file for the raw data and '.jpg', '.png', or '.jpeg' files for the corresponding stimuli. The back-end handles unzipping and storing the files into their respective directories on the file system. A client can then easily request these datasets and stimuli through HTTP requests. Likewise, a list of the available datasets and stimuli can be requested. The back-end uses the following libraries to achieve these functionalities:

- Express, a web application framework;
- Express-fileupload, a 'middleware' enabling uploading of files;
- EJS, a templating engine for reusing and modularizing HTML, CSS, and JavaScript;
- ADM-ZIP, a tool for reading zip files.

An input form that only accepts '.zip' files is displayed to upload and submit the data, which is then read in the browser. Using D3's ability to parse '.csv' files, the dataset file is read from the server and parsed so that it can be used directly in JavaScript. The library used for visualizing the data is D3, one of the most popular JavaScript libraries for creating dynamic visualizations. All of the visualizations use this library. The tool also employs the jQuery library, which increases readability and allows posting AJAX requests easily. The user interface is built using Bootstrap.

The tool can be deployed like any other Node application. The project includes a Dockerfile for building and running in Docker, a container platform. To achieve a secure, encrypted connection, the application is run behind a reverse proxy using Caddy, a web server. It can be publicly accessed at the following address: `https://eyevis.dobbel.dev`.

It is to be noted that the tool does not work properly in pre-chromium Microsoft Edge and that the visualization interactions run less smoothly in Mozilla Firefox than they do in Google Chrome, with the former also not displaying the scanpath's saccades. Therefore, the use of the latter is highly recommended.

4.4 Visualization Techniques

This section describes the details and features of the visualization techniques present in our tool, explaining the strengths and shortcomings of each one.

4.4.1 Scanpath

To reiterate the definitions mentioned in [1], the word 'scanpath' is used to describe a sequence of fixation points and saccades, visualized over a

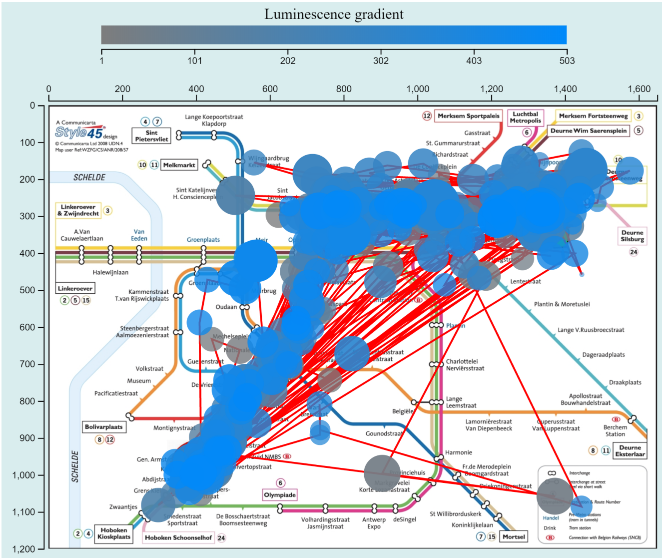


Fig. 6: Scanpath visualization showing all study subject fixation points and saccades for a selected stimulus: a colored metro map of Antwerp. It also features a color gradient and coordinate axes.

static stimulus [13]. A fixation point is defined as an aggregation of gaze points, based on a specified area and time span, usually between 200 and 300 ms [13]. Between consecutive fixation points, in chronological order, there are saccades, which describe rapid eye movement from one fixation point to another.

Scanpath visualizations are useful because they provide a spatio-temporal overview of the eye tracking data, making it easy to identify patterns and paths taken by different study subjects. However, as observed by [20], one great drawback of scanpaths is their tendency to have too much visual clutter (Figure 3), therefore, to offset such a problem, a per-study-subject view has been added.

As the paper indicates [1], there are various approaches meant to solve the clutter problem, such as averaging scanpaths, or removing circles in favor of displaying different fixation durations through line thickness or color.

Our visualization takes the standard approach of having each circle represent a fixation point, where the radius corresponds to the fixation duration and lines represent saccades between fixations [1]. It is also possible to only focus on one study subject at a time, in order to minimize clutter.

In order to offset the possible drawback of having a crowded set of points, which causes fixation durations to not be easily distinguished, the visualization has been linked with a heatmap. This linked view was developed because the heatmap clearly displays, through the calculation of densities, which areas have been observed the longest (relative gaze duration heatmap) or which have been observed the most often (fixation count heatmap) overall. Moreover, the heatmap shows fixation points of a constant and small size, making it easier for them to be hovered over and trigger the pop-ups displaying the fixation duration and fixation index. Both the scanpath and the heatmap work in parallel, allowing users to easily and fully benefit from the advantages of both visualizations, while also reducing the drawbacks that come from both.

The visualization was created using the D3 library by adding circles with radii that scale logarithmically with the fixation duration. Because fixation durations vary vastly, such a logarithmic function is appropriate, as the difference in circle radius can still be easily observed, while maintaining a manageable size. Therefore, this approach gives a clear and easy overview of which points have been observed longest. Saccades are just lines that follow the circles in chronological order, connecting the origin of 2 consecutive fixation points. Upon hovering over on a point, a pop-up displays the coordinates on the XY-plane and the fixation duration of the point. Furthermore, our version of the scanpath visualization comes with XY-coordinate axes, to easily locate and identify points on the plane. In addition, a luminescence gradient is provided for the fixation points, spanning from the lowest to the highest possible luminescence, indicating the start and finish of the observations respectively, thereby also offering a temporal representation.

4.4.2 Heatmap

Heatmaps, also called attention maps, are a great tool to intuitively display the points that were observed the longest. This visualization technique usually aggregates fixation points over time in one static color-coded density field which highlights the hot-spots of attention [4]. As previous research also indicates [3], the usefulness and convenience of heatmaps lies in the little effort required to read and understand it. The intuitive color scale and the positioning of the data over the stimulus allow the user to identify the most commonly gazed points in just one glance.

The ability of heatmaps to provide a clean depiction of aggregated gaze fixations, even when visualizing multiple study subjects, represents one of its main advantages over scanpaths [9]. However, it is to be noted that heatmaps sacrifice displaying the order of the fixation points. This issue is the reason why our tool offers a linked and simultaneous view that includes scanpaths and other temporal-based visualizations.

Despite their helpful properties when visualizing aggregated gaze fixations, heatmaps also have some disadvantages. First of all, if the difference in time between areas with little attention and areas with much attention is large, the areas with little attention might not be colored clearly enough, or at all. Secondly, the visual span that participants can observe with their peripheral view is unknown. Thirdly, the transitions from one color to the next are not sharp, so it is difficult to interpret the colors in terms of specific values [2].

In order to maximize the value of this kind of visualization, a decision was made to let the user switch between two types of heatmap: relative gaze duration and fixation count. The former calculates the fixation-point density in a given area by weighing the points based on their duration, whereas the latter assigns the same weight to all points regardless. Furthermore, in both types, the gradient is programmed in such a way that the lower end assumes the minimum value among all the densities calculated, while the higher end assumes the maximum value. This decision was taken because having a fixed linear gradient scale would drastically decrease the color variety within individual visualizations. Moreover, the adoption of a gradient-density legend allows the users to uncover disparities between the relative scales of different data.

In an attempt to solve the problem related to the neglecting of low attention areas, the heatmap features fixation points displayed as dots. These re-sizeable circles offer the user the opportunity to uncover the fixation points left out by the heatmap overlay due to a low-density area. Moreover, information about the fixation duration is available on hover, together with XY-coordinates and a specific study subject. This feature was developed as a partial solution to our inability to create a linear relationship between the gradient and the upper bound of the fixation duration in each generated density path, as advised by previous research [3]. Furthermore, to solve the issue related to the color transitions not being sharp enough, a hover-over pop-up paired with the highlighting of the hovered-over path displays the exact density value of it.

The visualization was obtained using D3 in combination with the d3-contour plug-in, which facilitates the calculation of the fixation-point density in different areas and allows for the generation of different-colored paths based on the results. Since this plug-in allows for the setting of the weight for each data point, this meant we were able to offer both a fixation count and a relative gaze duration heatmap. For the former, we assigned all points a weight of 500 in order to get a larger density range that could be read more easily. Concerning the latter, the weight of each gaze point was directly assigned based on the "FixationDuration" attribute.

A drawback of d3-contour is that the path densities calculated are based on the size of the svg element, meaning it was necessary to decide on a predetermined size for the visualization, which then gets scaled based on the window width. This solution allows users with different sized windows to see the same exact heatmap if they choose the same parameters, but could also make it harder to look at the numbers on the axes in certain cases.

Although the result obtained cannot be considered perfect, this type of visualization can still be regarded as useful to see the "bigger picture" in the data and identify general patterns or trends [3]. An example of this is shown in Figure 7, from which it is immediately clear that two particular intersections received generally more attention from the study participants during their quest for the right path from the starting point to the target.

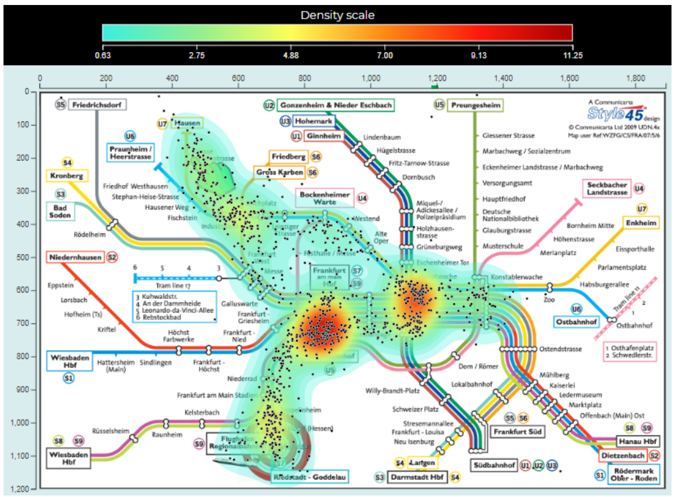


Fig. 7: A relative gaze heatmap that visualizes eye tracking data belonging to every participant in the study over the used stimulus: a colored metro map of Frankfurt. The visualization also displays single fixation points and gives a clear overview of the two intersections that received the most attention from the study subjects.

4.4.3 ThemeRiver

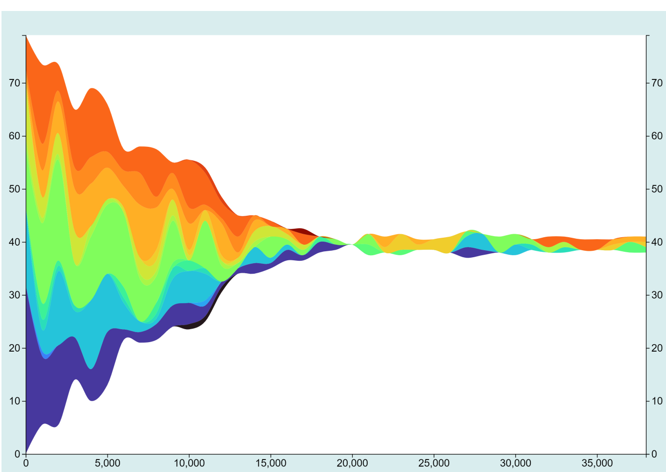
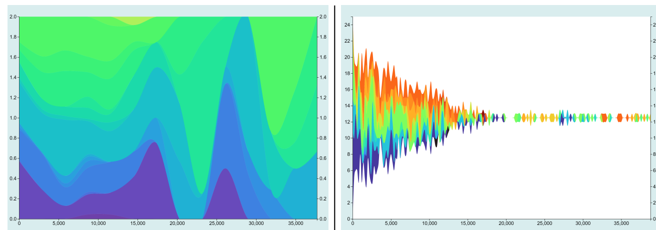


Fig. 8: A ThemeRiver Visualization based on the metro map data set. This example uses the "01 Antwerpen S2" stimulus with a 5x4 grid size and a time interval length of 1000 ms.

ThemeRivers, sometimes called streamgraphs [7], are stacked graphs that portray thematic change over time [11]. On the topic of eye tracking visualization, a ThemeRiver visualization gives information on areas of interest in relation to time. The streams of the graph show the number of fixations within each AOI in a defined time interval, allowing users to observe the flow of the relative proportions of each area of interest and to search for trends, patterns, and relationships [12]. Moreover, up to a certain degree ThemeRivers can also give insights into the mean time taken by users to complete the task when there was no strict time limit given. This information can be drawn from the graph, as more fixations will be present within each AOI within a certain time span. An example of this feature can be observed in Figure 8, where there seems to be a significantly higher number of fixations within the first half of the graph with respect to the second half, suggesting that some users finished earlier than others.

One of the challenges is finding a good predetermined time interval length, for when the user has not yet adjusted the slider. While data with all study subjects benefits more from a higher time interval length, the single participants are able to display more information when given a shorter time interval length. A predetermined time interval length too short could result in an all-study-participants-graph that is impossible to read, as shown in Figure 9b.



(a) Normalized ThemeRiver of "01 Antwerpen S2" stimulus from the metro map data set.

(b) Example of ThemeRiver with time length interval 100, which shows it is hard to read.

Fig. 9: Examples of ThemeRiver types to illustrate interactive possibilities and issues mentioned in the text.

Several interactive possibilities were added to improve user interactivity. For one, users are able to decide on the grid size to select the number of AOIs based on which the stimulus will be divided. The slider option is available to adjust the length of the time intervals used to calculate the number of fixations within each area of interest. Another user-selectable option is to change the graph from standard to normalized as shown in Figure 9a. The "silhouette" offset is what distinguishes ThemeRivers from other streamgraphs, which use a "wiggle" offset [6]. Lastly, the graph responds to hovering, zooming, and panning. Hovering over a river will lower the opacity of the other rivers and display a tooltip with the name of the AOI, the number of fixations in it at that time, and the timestamp.

The current visualization, with all the interaction possibilities, already conveys the necessary information. However, slight changes could be implemented to improve it further. For instance, in this web tool a time interval length of 800 milliseconds was chosen as a middle ground, but in future work this could perhaps be changed. A possible solution to the disadvantage described above could be to have separate predetermined lengths, one for all study subjects and one for single participants.

4.4.4 Scarf Plot

Scarf plots offer a way to visualize the AOIs of a study subject over time. It shows a timeline of gazes inside an AOI by color-coding every AOI and changing color whenever the gaze of a study participant enters a different AOI.

The strength of the scarf plot is that, for a single participant, it has a single dimension based on time, thus it is possible to add multiple study subjects vertically as a second dimension. This provides an intuitive and clear way of comparing different participants based on the areas of interest they observed over time. As described in [1], larger areas of interest are desired, as smaller areas increase the number of transitions from one AOI to the other, thereby decreasing readability.

Compared to the ThemeRiver visualization, which offers similar insights, the scarf plots allow identification of single participants: while the ThemeRiver groups study subjects together, the scarf plot keeps them separate.

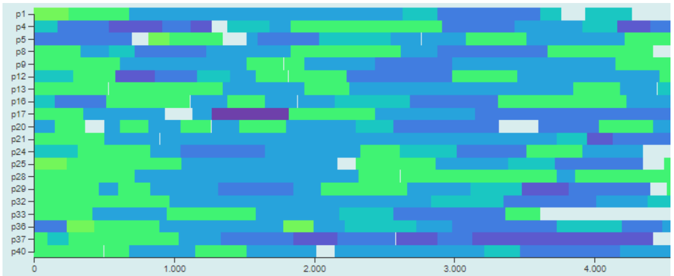


Fig. 10: An example of the scarf plot visualization technique. Shown is half the scarf plot for the colored Antwerp metro map. On the y-axis are the study subjects and on the x-axis the timestamp relative to the start of the measurement. This particular scarf plot uses a 3x3 grid, meaning that the stimulus is split in a 3x3 grid of squares, each with their own color.

4.4.5 Sankey Diagram

Sankey diagrams [18] have been used for a long time to illustrate the flow of people or objects. For example, the Minard map [17] shows the movement of soldiers in an army, encoding the number as differently thick rivers and subrivers. The movement direction and the splitting and merging behavior is indicated by typical intuitive visual variables [5].

A Sankey diagram offers a clear visualization of the transitioning of areas of interest (AOIs), which are automatically computed by applying a grid onto the stimulus, with parameters specified by the user, and are then represented as color-coded vertical bars. In the case that multiple AOIs are located on the same layer, these will be stacked vertically on each other. Moreover, the user is granted the ability to move the vertical bars around the visualization container. The thickness of a link between two AOIs visually encodes the number of transitions between them [5].

This visualization is quite powerful since it can clearly show people's attention behavior specifically regarding the transition between areas of interest. However, there exists a limit to the number of areas of interest that can be displayed, as an excessive amount can quickly produce a lot of visual clutter [5]. An example of a Sankey diagram can be observed in Figure 11.

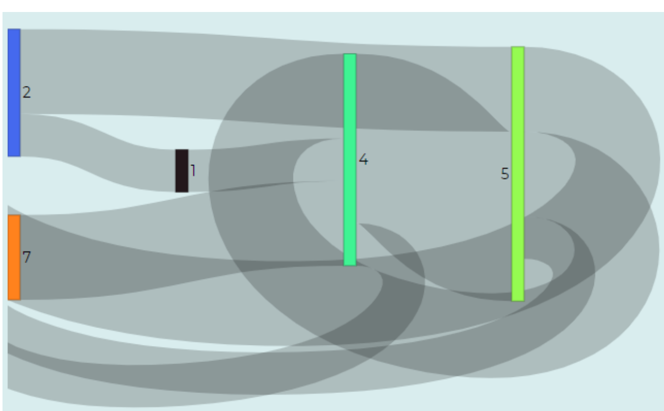
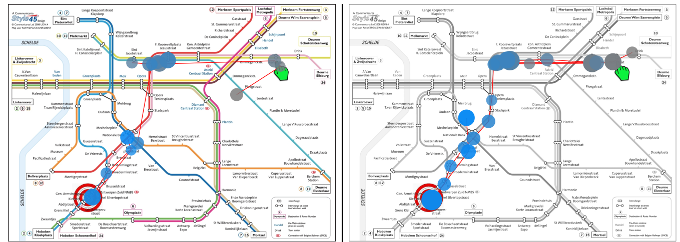


Fig. 11: Example of Sankey diagram showing the AOI transitions.

4.5 Application Example

To illustrate the usefulness and functionality of our tool, we will partially repeat the analysis of a formerly conducted real-world eye tracking experiment, where participants were presented with route-finding tasks in public transport maps [16]. We will conduct this partial analysis by considering some of the hypotheses and research questions that were described in the original paper and giving visualizations that exemplify and give an indication to these hypotheses.



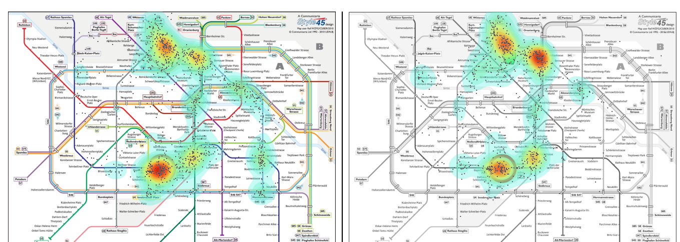
(a) A scanpath on the colored metro map of Antwerp with long saccades and short fixation durations, as seen by the red lines connecting the dots and the radius of the dots respectively.

(b) A scanpath on the grayscaled metro map of Antwerp with relatively short saccades and longer fixation durations, as seen by the red lines connecting the dots and the radius of the dots respectively.

Fig. 12: A scanpath comparison between colored and grayscaled metro maps.

H1: "In color-coded metro maps, the saccades will be longer and the fixation durations will be shorter than in grayscale maps. The colored lines will help the participants identify a desired line faster (i.e., with lower fixation duration) even if the participants will cover a long distance with eye movements (i.e., longer saccade length)."

As seen in Figure 12, there is an indication that this hypothesis is true. Figure 12a shows a scanpath on the colored metro map of Antwerp and Figure 12b shows a scanpath on the same map, but now grayscaled. There is a clear difference between both saccade length and fixation duration. The colored lines are more easily distinguished, leading to an increase in saccade length, as seen by the fact that almost all fixation points are located near an interchange point and not on the lines themselves. This is not the case on the grayscaled map, where participants likely had more trouble following a line. Also, an increase in radius size of the fixation points on the grayscaled map indicates longer fixation durations.



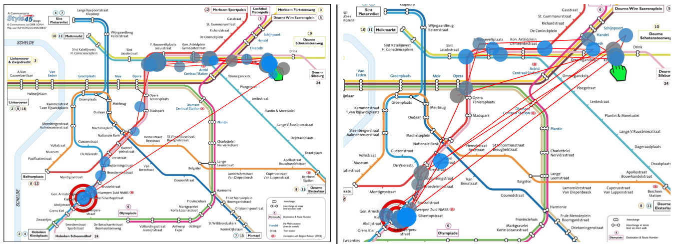
(a) A heatmap on the colored metro map of Berlin, indicating concentrations of fixation points by study subjects.

(b) A heatmap on the grayscaled metro map of Berlin, indicating concentrations of fixation points by study subjects.

Fig. 13: A heatmap comparison between colored and grayscaled metro maps, indicating a difference in reading strategy based on the use of colors.

H2: "There will be distinguishable reading strategies that can be separated clearly."

Although the hypothesis does not mention a difference between colored and grayscaled maps, we found an interesting example of how this might influence reading strategy. Figure 13 shows heatmaps of the metro map of Berlin, with Figure 13a displaying a colored version and Figure 13b displaying a grayscaled version. As can be seen, participants that were presented the colored version often chose a path that participants that were presented the grayscaled version did not, namely the path on the far right. This path required the least amount of transfers, but the lack of color apparently influenced participants to not choose this path.



(a) A scanpath on the colored metro map of Antwerp, where a user verifies their result by going from start to end and backwards only once.

(b) A scanpath on the colored metro map of Antwerp, where a user verifies their result by going from start to end and backwards repeatedly.

Fig. 14: A scanpath comparison visualizing two of the three main visual task solution strategies.

H3: "There are three main visual task solution strategies: (1) find a route directly from a start to an end location, (2) go from start to end and backwards verifying the result, (3) use multiple repetitions of going back and forth."

Figure 14 shows examples that indicate the truth of this hypothesis by using the scanpath visualization. The saccade between consecutive fixation points nicely shows the users verifying the results by going from start to end once, as seen in Figure 14a, or multiple times, as seen in Figure 14b.

5 Discussion and Limitations

The consideration of the possible limitations or shortcomings users may experience when working with our tool is always necessary.

Firstly, our tool allows the user to select and visualize the data of either a single or all participants. However, a user might want to compare a subset of all participants in order to gain insights into their data, e.g. compare the scanpath of multiple participants to easily recognize different reading strategies.

Secondly, our tool supports linking between both point-based visualizations and AOI-based visualizations separately. Although these are the two main visualization categories in our tool, a user might be interested in having the ability to get a linking feature between the two categories in order to comprehend the flow of single fixation points.

Finally, our tool defines AOIs based on a customizable grid. The possibility for a user to define AOIs themselves or based on fixation points is something that could vastly improve the practicality of our visualization tool.

6 Conclusion and Future Work

In this paper, we have presented a web-based project designed to reliably and accurately visualize eye tracking data. The tool can be accessed by visiting a web page and does not require the installation of additional software. Furthermore, the user interface makes it easy for data to be uploaded, thereby providing researchers with a quick and straightforward resource to gain insights into their data. This tool supports the option to upload different data sets, allows the user to benefit from multiple linked visualization types, and provides different interaction techniques.

Potential future work includes the option to select a subset of study participants to visualize, giving the users the ability to select the areas of interests or having a clustering algorithm select them, and offering the option to filter the dataset by time intervals.

Acknowledgments

We would like to thank all the group members for contributing, our tutor for providing helpful feedback, and Michael Burch for responding to emails.

References

[1] T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl. Visualization of eye tracking data: A taxonomy and survey. *Comput. Graph. Forum*, 36(8):260–284, 2017. doi: 10.1111/cgf.13079

[2] P. Blignaut. Visual span and other parameters for the generation of heatmaps. *Proceedings of the Symposium on Eye Tracking Research & Applications*, 2010. doi: 10.1145/1743666.1743697

[3] A. Bojko. Informative or misleading? heatmaps deconstructed. *Proceedings of the 13th International Conference on Human-Computer Interaction. Part I: New Trends*, 2009. doi: 10.1007/978-3-642-02574-7_4

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

[5] M. Burch and N. Timmermans. Sankeye: A visualization technique for aoi transitions. 04 2020. doi: 10.1145/3379156.3391833

[6] L. Byron and M. Wattenberg. Stacked graphs – geometry aesthetics. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1245–1252, 2008.

[7] E. Cuenca, A. Sallaberry, F. Y. Wang, and P. Poncelet. Multistream: A multiresolution streamgraph approach to explore hierarchical time series. *IEEE Transactions on Visualization and Computer Graphics*, 24(12):3160–3173, 2018.

[8] A. T. Duchowski. A breadth-first survey of eye tracking applications. *Behavior Research Methods, Instruments, Computers*, 34(4):455–470, 2002. doi: 10.3758/bf03195475

[9] A. T. Duchowski, M. M. Price, M. Meyer, and P. Orero. Aggregate gaze visualization with real-time heatmaps. *Proceedings of the Symposium on Eye Tracking Research and Applications*, 2012. doi: 10.1145/2168556.2168558

[10] S. Eraslan, S. Karabulut, M. C. Atalay, and Y. Yesilada. Vista: Visualisation of scanpath trend analysis (sta). In *Proceedings of Turkish National Software Engineering Symposium (UYMS)*, 2018.

[11] S. Havre, B. Hetzler, and L. Nowell. Themeriver: visualizing theme changes over time. In *IEEE Symposium on Information Visualization 2000. INFOVIS 2000. Proceedings*, pp. 115–123, Oct 2000. doi: 10.1109/INFVIS.2000.885098

[12] S. L. Havre, B. Hetzler, and L. T. Nowell. Themeriver*: In search of trends, patterns, and relationships. 1999.

[13] K. Holmqvist. *Eye tracking: a comprehensive guide to methods and measures*. Oxford University Press, 2011.

[14] D. Keim and L. Zhang. Solving problems with visual analytics: Challenges and applications. In *Proceedings of the 11th International Conference on Knowledge Management and Knowledge Technologies*, i-KNOW '11. Association for Computing Machinery, New York, NY, USA, 2011. doi: 10.1145/2024288.2024290

[15] 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 '16, p. 11–18. Association for Computing Machinery, New York, NY, USA, 2016. doi: 10.1145/2857491.2857507

[16] 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. doi: 10.1080/13875868.2016.1226839

[17] G. Pálsson. A review of the minard system: The complete statistical graphics of charles-joseph minard. 04 2019.

[18] P. Riehmann, M. Hanfler, and B. Froehlich. Interactive sankey diagrams. In *Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, INFOVIS '05, p. 31. IEEE Computer Society, USA, 2005. doi: 10.1109/INFOVIS.2005.21

[19] J. C. Roberts. Exploratory visualization with multiple linked views. In A. M. MacEachren, M.-J. Kraak, and J. Dykes, eds., *Exploring Geovisualization*. Amsterdam: Elseviers, December 2004.

[20] R. Rosenholtz, Y. Li, J. Mansfield, and Z. Jin. Feature congestion: a measure of display clutter. In G. C. van der Veer and C. Gale, eds., *Proceedings of the 2005 Conference on Human Factors in Computing Systems, CHI 2005, Portland, Oregon, USA, April 2-7, 2005*, pp. 761–770. ACM, 2005. doi: 10.1145/1054972.1055078