Blickshift Analytics

Manual

Version 1.0.11.1

Blickshift GmbH
Schelmenwasenstr. 34
70567 Stuttgart

Tel.: +49 711/184 245 26
support@blickshift.de

Contents

CHAPTER 1

Introduction

1.1 Concepts

This chapter defines and explains some terms that are used throughout this documentation. It is not necessary to read through the entire chapter. Instead, when one of the terms is used, a link back to the fitting section of this chapter is given, allowing you to quickly look up the meaning.

In this chapter, we assume that you are analyzing data from a classic user study, with participants and tasks (or scenarios) that the participants performed. This assumption is mainly used to make the nomenclature easier to understand. If the data you want to analyze is structured differently, you can try to map your data onto the nomenclature used here.

1.1.1 Data

In order to fully understand all the possibilities of analyzing data with Blickshift Analytics, it is helpful to understand how the data is structured. Figure 1.1 shows how data sets, scenarios, participants and data columns relate to each other.

1. Data Set A data set is the topmost data structure in Blickshift Analytics. A data set contains a number of data tables, all with the same data columns. In many cases, you will only ever have one source data set for your project.

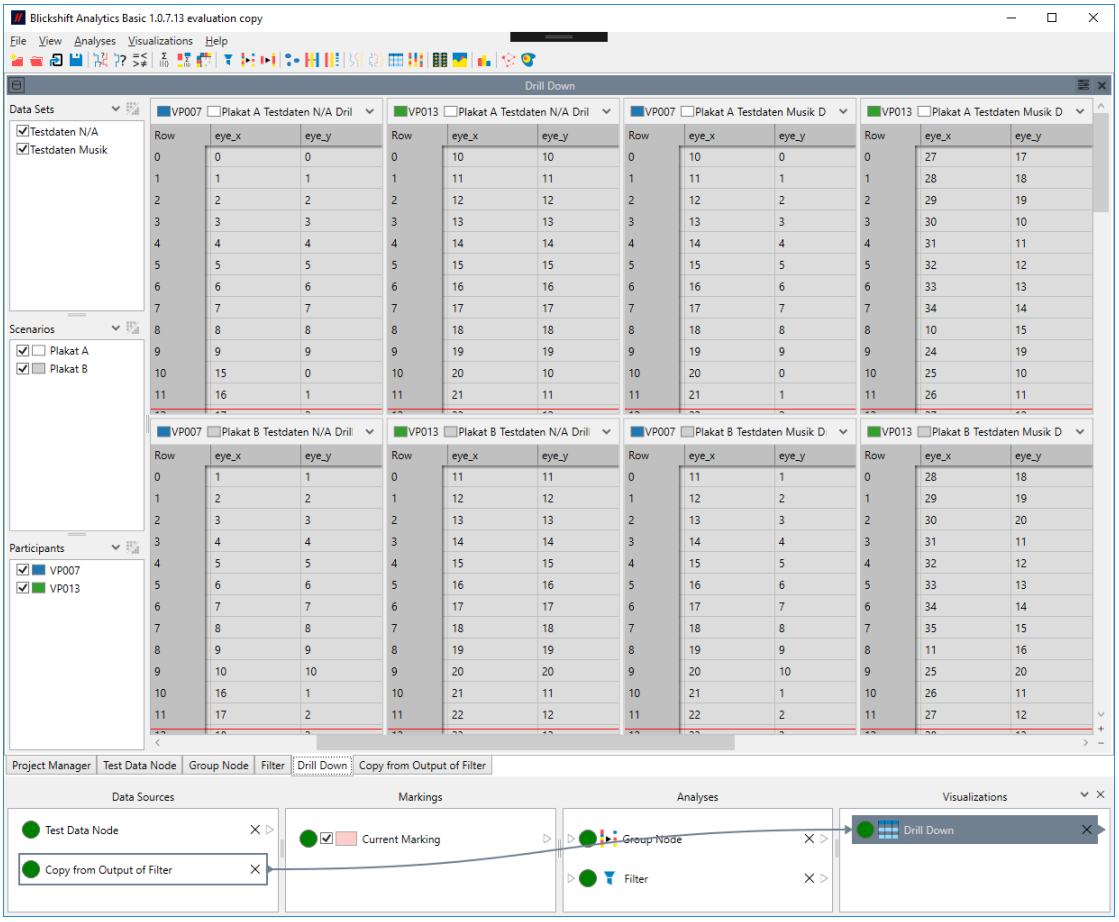

Figure 1.1 — Two data sets, two scenarios and two participants result in eight data tables. For each data table, two data columns are visible.

Other data sets may be produced by analyses. Most analyses produce one output data set per input data set. Some analyses, such as the Group Node, produce many output data sets, which gives you the opportunity to organize your data according to specific criteria.

2. Data Table A data table is a unit of data, with many data columns. In each data set, there is (at most) one data table per scenario/participant combination, which contains the data pertaining to one particular participant in one particular scenario.

3. Scenario A scenario is a specific task or other unit that was performed by your participants. In a classic eye tracking study for advertisements, each different advertisement would be one scenario.

4. Participant A participant is one person that participated in your study.

5. Data Column A data column is all the data in one data table that belongs to the same data dimension, e.g. one coordinate of your eye tracking data, the speed of a test vehicle, the temperature, etc. Most analyses and visualization operate on only one or two data columns, and you have to select the data column that you want to get a result for.

    a) Enumerable Column Enumerable Columns are data columns that contain only a low number of distinct data points, as opposed to data columns that contain continuous data, like e.g. the vehicle speed. The prime example of enumerable columns are AOIs. Some analyses and visualizations only work on enumerable columns.

6. Data Node A data node is a node that hosts data and is used as a source for analyses and visualizations. Its window shows a preview of the data, if possible, and its Settings pane allows to set some metadata for the data columns. The importer tries to guess these metadata, but with the variety of data formats, those guesses are not always correct. Setting up the metadata correctly helps analyses and visualizations to often correctly guess the required columns for specific purposes.

    a) Settings

        Eye X Coordinates: The raw eye coordinates (x direction). The selected column will be used by default by the Fixation Computation.

        Eye Y Coordinates: The raw eye coordinates (y direction). The selected column will be used by default by the Fixation Computation.

        Fixation X Coordinates: The fixation coordinates (x direction). The selected column will be used by e.g. the Scan Path and the Heat Map.

        Fixation Y Coordinates: The fixation coordinates (y direction). The selected column will be used by e.g. the Scan Path and the Heat Map.

        Fixation Duration: The duration column of the fixations. If your fixation computation algorithm has computed durations for each fixation, select this column here. It can then be used in the Scan Path.

            Fixation Duration Unit: The unit of the fixation duration. Set this to the time unit of the column you have selected. If the

time unit is of type "TimeSpan" or "DateTime", this setting is fixed.

Time: The main time column. This will be used by nodes that compute values based on time, such as the gaze durations statistics. It is also used as the default time axis in time-based visualizations. You can only select columns with monotonically increasing values here, as others would result in incorrect results for the computations.

Time Unit: The unit of the time column. Set this value so that computation nodes that use the time column know about the time unit of the column they are computing on.

Video Time: The video time column. The selected column is used by default by the film strip and stimulus visualization.

Video Time Unit: The time unit of the video time column. Setting this column to the correct time unit helps the film strip and stimulus visualization to display the correct frame with the default settings.

AOI: The AOI column. The selected column is used by default by the Parallel Scan Path, the Gaze Duration Statistics, the Transition Matrix, the Sequence Search, and the Sequence Analysis.

Stimulus Name: If you have data, that associates different stimuli within the same data table (i.e. scenario/participant combination) and a column that contains stimulus names, you can select this column here. This column will then be the default stimulus column, if you set stimulus selection of the stimulus visualization to "Column".

1.1.2 Markings

A marking is a reference to a specific subset of the data that exists anywhere in the current project. In the simplest case, a marking references a subset of the base data that you want to analyze.

The marking itself behaves just like any other data set. This means that you can use a marking as an input for visualizations and analyses, just like you can do with any other data. If you use a marking, only the subset that is referenced by the marking will be visualized and analyzed. Figure 1.2 shows a marking on the base data set that is displayed on a line graph, and the same marking visualized by another line graph.

Markings currently have three different sources:

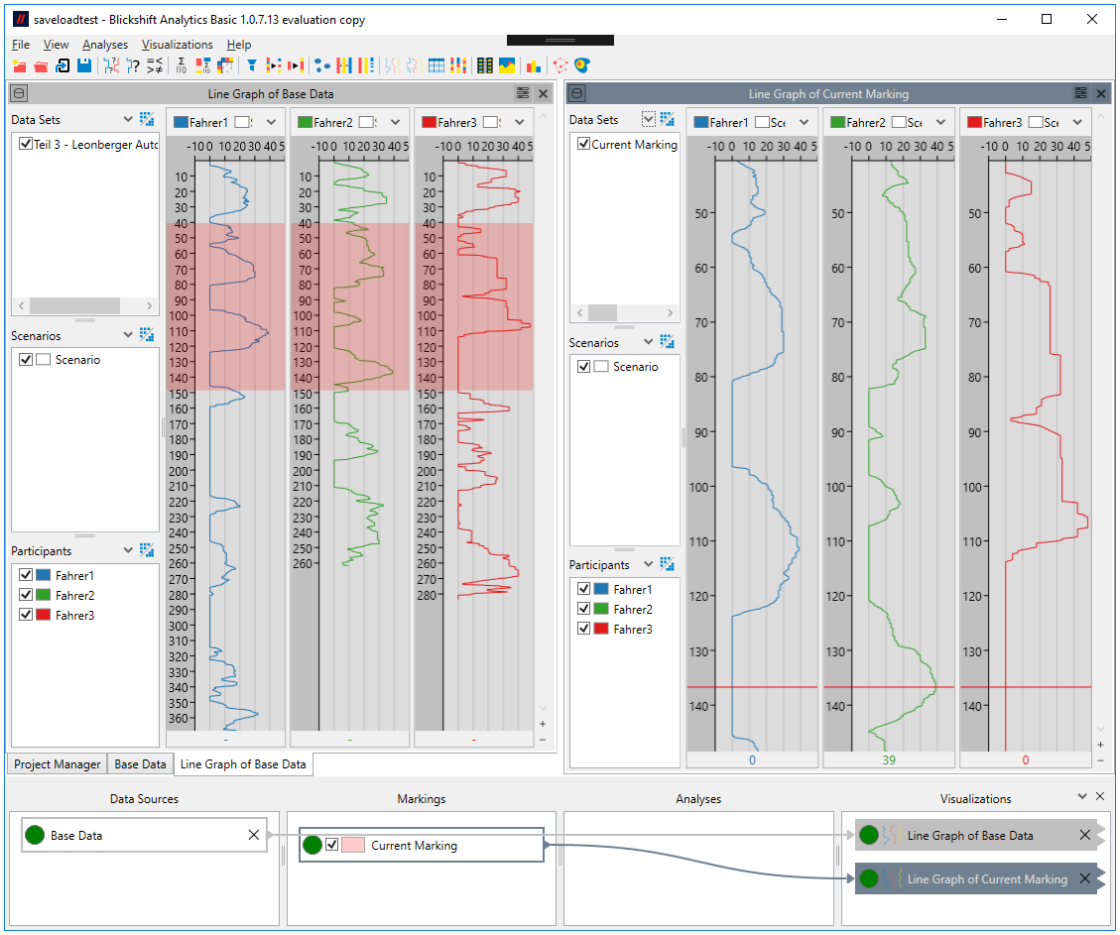

Figure 1.2 — A marking displayed on a line graph, and the same marking visualized by another line graph.

Current Marking: The current marking can be manipulated by the user, normally via the mouse cursor in time-based visualizations.

Analysis Result Markings: The marking is the result of an analysis, e.g. a Value Search or a Sequence Search. The marking references those parts of the data that fulfill the criteria of the analysis.

Stored Markings: As the current marking and analysis result markings can change rapidly, you can make any marking permanent by duplicating it. Once a marking has been duplicated (stored) it will no longer change, even if the original marking is changed or deleted.

1. Split Marking A split marking is a marking that references data that is not continuous. Figure 1.3 shows a split marking, in contrast to the continuous

marking in figure 1.2. The line graph that visualizes the split marking displays split markers that indicate the discontinuities in its data.

Split markings can be used just as any other marking, but there are some analysis options that do not work (or only work) with split markings. The documentation of the analyses mentions whenever such cases exist.

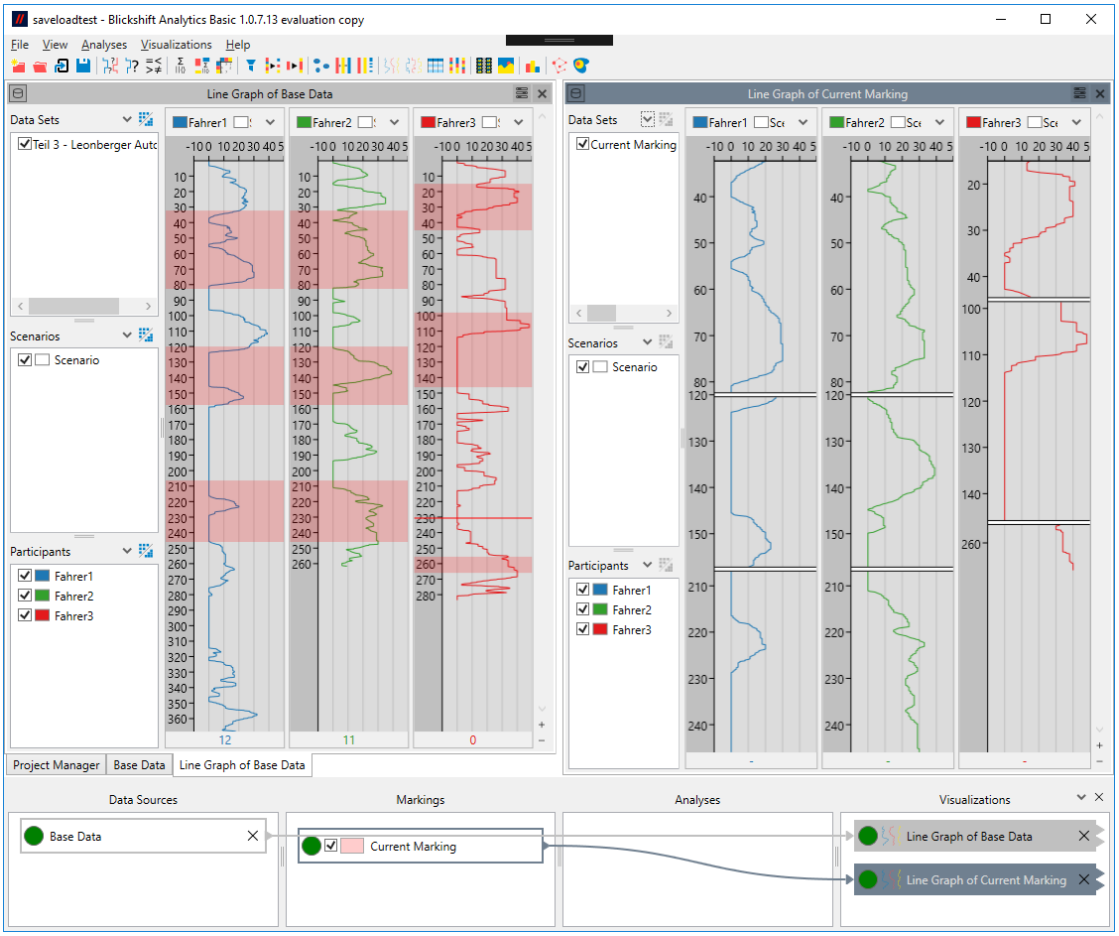

Figure 1.3 — A split marking shown on a line graph, and the same split marking visualized by another line graph.

1.1.3 Analyses

An analysis is a component that receives input data, performs a computation (often influenced by parameters) and creates output data. Most analyses do not have their own visual representation of the data. Some analyses offer a preview of the result similar to a Drill Down visualization. Some analyses that output markings (e.g. Value Search and Sequence Search) do not offer previews for the

data. Instead they can be added to the same window as a visualization of their base data, where the result marking will be highlighted.

1.1.4 Visualizations

Visualizations are components that receive input data and display the data in a visual manner. Visualizations do not normally output any data.

Time-based visualizations are visualizations that display time along one axis. For more information, see time-based visualizations.

CHAPTER 2

The Blickshift GUI

2.1 Project Manager

In the Project Manager general settings that pertain to the entire project can be changed. The scenarios and participants and their names and colors, and media files and how they are connected to scenarios and participants are handled here. The Project Manager is shown as one of the tabs in the main view. If you have closed it, you can reopen it via "View - Project Manager" in the main menu.

2.1.1 Scenarios

The Project Manager lists all scenarios that are currently in use in the project on the left side. When you select a scenario, its properties are shown on the right. At the moment, you can change the name and the saturation of the scenario. The saturation is used in conjunction with the color of the participants to define colors for scenario/participant combinations, which are used in visualizations.

On the second tab on the right, the association with stimuli for all participants in the project is shown. You can also change the association here. In most cases, however, it will be easier to change the association on the Scenario/Participant tab of the Stimulus.

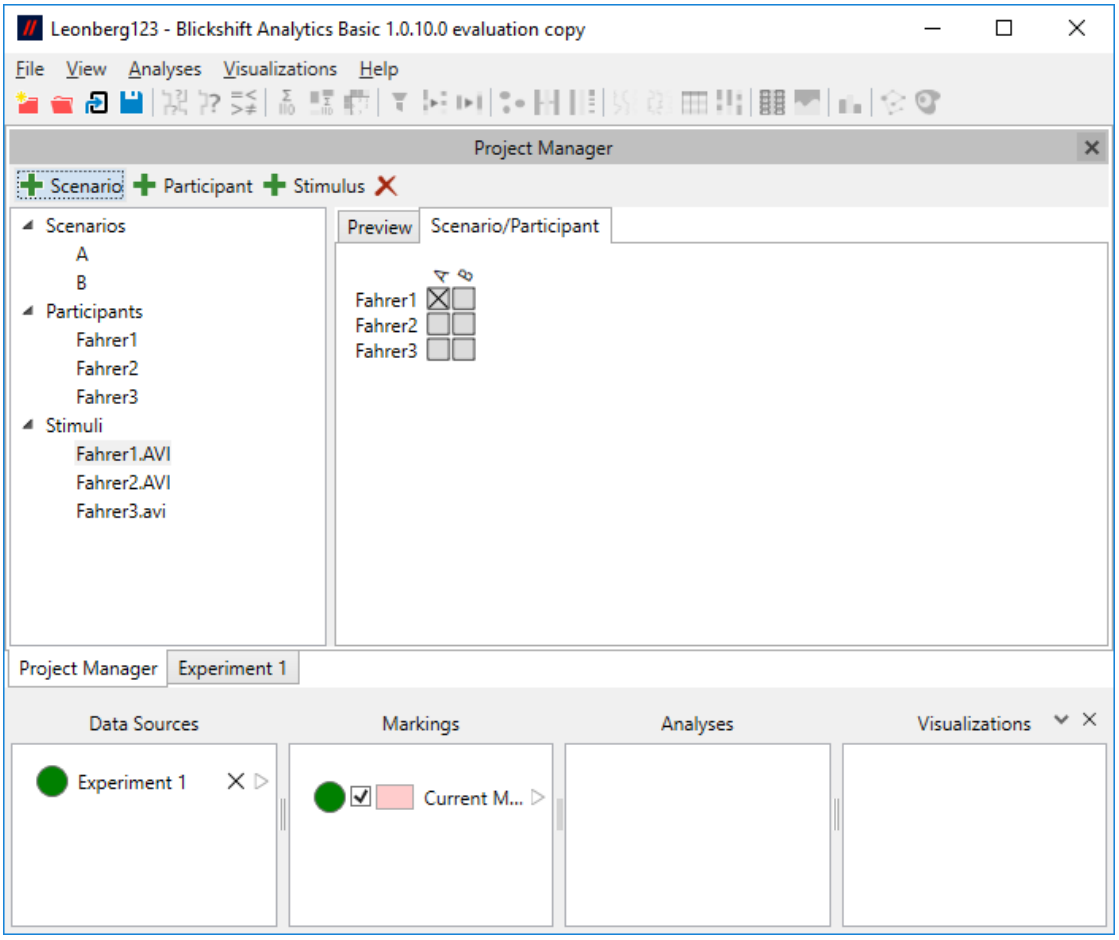

Figure 2.1 — The project manager

2.1.2 Participants

Similar to the scenarios, the participants are also listed on the left side. When a participant is selected, you can change its name and its color.

2.1.3 Stimuli

The third kind of data that can be manipulated in the project manager is stimuli. If you have imported the stimuli during the import process successfully, the stimuli will be listed on the left side, below scenarios and participants. You can also import stimuli in the project manager. To do so, click the "Add Stimulus" button in the tool bar of the project manager. You can then select one or many files and import them into the project. If you import the stimuli with this method, you will need to associate them with scenarios and participants

manually (see below).

If you select a stimulus in the list, the right side will show two tabs. The "Preview" tab shows a preview of the selected stimulus. On the scenario/participant tab, you can determine the association of the stimulus with scenarios and participants. It shows a matrix of all scenarios and participants currently in the project. The boxes in the matrix are ticked, if the currently selected stimulus is associated with the corresponding scenario/participant combination. You can tick/untick the boxes manually, or you can also associate stimuli with entire columns of scenarios (or rows of participants) by clicking on the name of the scenario (or participant). This is especially useful, in cases where scenarios correspond directly with the stimuli that were shown to participants.

Note that you can also import stimuli and not associate them with scenarios and participants. Those will never be shown in visualizations by default, but some visualizations (e.g. stimulus visualization and film strip) allow other methods of selecting media files, as long as they have been added in the project manager.

2.2 Workflow Explorer

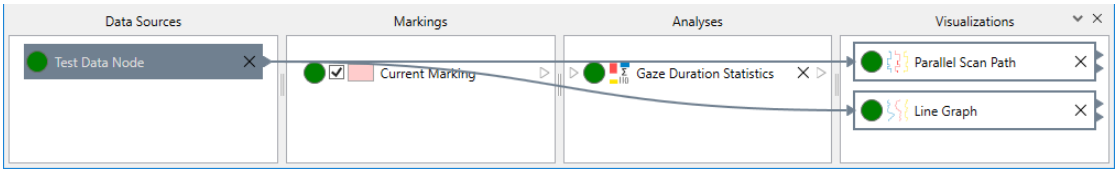

Figure 2.2 — The workflow explorer

The workflow explorer is displayed at the bottom of the window. If you have closed it via the close the button in its top right corner, you can reopen it via "View - Workflow Explorer" in the main menu.

2.2.1 Concepts

The workflow explorer shows how the different nodes form the current workflow. Nodes can receive data via one or more inputs on their left side and output data via one ore more outputs on their right side (figure 2.3). Each node represents one step in a workflow. By chaining different analyses and visualizations it becomes possible to build a workflow graph that is fitting for a specific task or question.

Figure 2.3 — The sequence analysis has one input and two outputs

The state of a node can be seen by the colored circle that is displayed with each node:

Processing (Light Green) The node is currently computing, e.g. because its input or the value of one of its parameters has been changed. Once the computation has finished successfully, the nodes that succeed it in the graph will start their computation. Note that while a node is computing its output can not be considered valid. Especially visualizations might still display the data as it was before the new computation was started. This allows a better comparison between the old and the new state, once the computation is finished.

Valid (Dark Green) The node was able to compute its last task successfully and its outputs are valid.

Failed (Red) There was an error during the last computation by this node. Such an error can occur e.g. because the input data is not fitting for the node or because some parameters in its settings pane do not allow a successful computation. Hover over the red circle to receive more information about the cause of the error.

Invalid (Gray) The node's output data is outdated or otherwise invalid. The main reason for this state is that no valid input data exists for this node and therefore no computation can be done. This might be because not all inputs of this node are connected, or because an failed state (rede) exists in a preceding node.

All nodes can be deleted by clicking the ✕ symbol on the node.

2.2.2 Workflow Explorer Columns

The main Area of the Analysis Explorer displays the following Columns:

1. Data Sources This lists the nodes that hold data. In many cases, this is just one node that gets created when you import data. However, you can also store intermediate results that have been computed as new data nodes. This ensures that the data of the intermediate result is not lost, no matter

what settings you change. Nodes displayed in this column only have outputs as they do nothing but provide data that should be analyzed.

2. Markings Displays all current marking, which cannot be deleted. It can be changed interactively in many visualizations, e.g. line graph or parallel scan path.

   Additionally, this column is used to display stored markings. You can store markings by selecting "Duplicate Marking" in the context menu of any node that generates a marking, or in the context menu of a visualization that displays this marking. Stored markings cannot be changed, but it is possible to change the current marking so that it equals a stored marking, and then manipulate the current marking. Select "Set as Current Marking" in the context menu of a stored marking node.

   Marking nodes have no input. They output a marking, i.e. a reference to certain areas in the data. This means that there are two ways to visualize a marking:

   - Visualize the data of the marked area, e.g. show a line graph of a certain segment of the data only.
   - Visualize the entire base data and highlight the area of the marking, e.g. show a line graph of the complete data and highlight the segment referenced by the marking.

   Accordingly, marking nodes provide two ways of manipulating how the marking is displayed.

   The first one is the output. When connected to another node, that other node only "sees" those areas, not the whole data. Thus, e.g. a heatmap connected to the current marking will only display a heatmap of whatever is currently marked by the user. Of course, it is also possible to connect analyses instead of visualizations.

   The second one is the checkbox and the color selector that sets marking nodes apart from other nodes. These two elements determine how visualizations that can highlight markings in the visualization of the complete base data handle the marking of the node. The checkbox determines whether the marking is highlighted at all, and the color selector determines the color of the highlight. Both affect all visualizations that display data which is referenced by the marking.

3. Analyses Displays all analyses in the current graph. Analyses get data as an input and output the result of the analysis as data. The result can either be displayed in a visualization or further analyzed by another analysis.

Analyses can either ouput plain data (similar to a data node in the "Data Sources" column), or a marking (similar to the marking nodes in the "Markings" column). Analysis nodes that output markings can be distinguished by the checkbox and the color selector that work in the same way as for marking nodes.

If an analysis node is selected, the window that contains the settings for that analysis is activated.

4. Visualizations Displays all visualizations in the current graph. Visualizations get data as an input and display a graphical view of that data.

   If a visualization node is selected, the window that contains the visualization is activated.

2.2.3 Workflow Explorer Graph Display

The workflow explorer displays the data flow and how the different nodes are connected. When a node is hovered over with the mouse, it is highlit in a light blue. The connections to other nodes are shown in the same color, and all nodes that either influence it (via input) or are influenced by it (via output) are shown with a box in the same color.

In a similar manner the part of the graph influenced by or influencing other nodes can be shown. This can be set via the workflow explorer options accessible in its top right corner.

Highlight graph of nodes in active window: Show the graph of all nodes displayed in the active window (or of the active node, if the active node does not display a window). This graph is displayed in the same color as the header of the active window.

Highlight graph of nodes in visible windows: Show the graphs of all nodes displayed in currently visible windows. This graph is displayed in the same color as the header of non-active visible windows.

Show all connections: Show all connections in the graph, regardless of whether the nodes are displayed in a visible window or not.

Hide nodes that are not part of highlighted graphs: Hides all nodes completely that are not part of currently displayed graphs. Which nodes exactly will be hidden depends on the first three options.

2.2.4 Changing the Flow of Data

It can be very useful to change the flow of data in the workflow explorer, e.g. if you have created a workflow that answers a specific question and want to apply that workflow to another source of data. There are three ways of influencing the workflow: Creating new connections, changing existing connections, and deleting connections..

To create a new connection, click on the little arrow that denotes an output, hold the mouse button and start dragging. You can then drop the connection onto any node that has an input. The connection will be created between the output you started dragging on and the first input of the node where you released the mouse button. If you need to connect to an input other than the first, you need to release the mouse button while hovering the arrow symbol of the input you want to target. If the input you are targeting is already connected to another node, this existing connection will be deleted. While you hover over a node or an input, a preview and a description of the new connections is shown. Sometimes, creating such a new connection will result in several connections, or a new connection is not possible, because other nodes are displayed in the same window. If you want to avoid that, make sure that the target node is displayed in a window of its own.

To change a connection, click on the little arrow that denotes an input, hold the mouse button and start dragging. When you drop the connection, the result will be the same as for a new connection, with the exception that the connection that already existed between an output and the input where you started the operation will be deleted.

To delete a connection, right-click on the input at the end of the connection you want to remove and select "Remove Connection".

2.2.5 Workflow Explorer Context Menu

The nodes displayed in the workflow explorer all have a context menu. Even though it might differ a little depending on the column and the type of the node, it generally looks somewhat as shown in figure 2.4

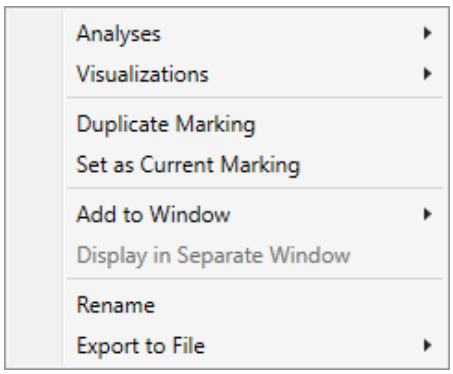

Figure 2.4 — Context menu of nodes in the workflow explorer

Analyses: Add a new analysis to the first output of the node, if the node is not a visualization. If you want to add an analysis to another output than the first, activate the context-menu directly on the little arrow icon of the output. If the node is a visualization, add a new analysis to the same output of the previous node, that this node is currently attached to, and add the new analysis to the window that displays the current node.

Visualizations: Add a new Visualization to the first output of the node, if the node is not a visualization. If you want to add a visualization to another output than the first, activate the context-menu directly on the little arrow icon of the output. If the node is a visualization, add a new visualization to the same output of the previous node, that this node is currently attached to, and add the new visualization to the window that displays the current node.

Duplicate Marking: Create a copy of the marking created by this node, and store the copy in the "Markings" column. This option is only available for nodes that output a marking.

Set as Current Marking: Set the current marking to be the same as the marking output by this node. This option is only available for nodes that output a marking and are not the current marking node.

Add to Window: Add settings and visualization of this node to another window. This option is only available for analyses and visualizations, and only windows of other nodes displaying the same data can be selected. By using this option it is possible to couple different visualizations, e.g. a line graph and a parallel scan path.

Display in Separate Window: Display the settings and visualization of this node in a separate window. This option is only available if the node is displayed in a window containing more than one node.

Rename: Rename the node.

Input: A submenu for the input of the node. Note that in cases where the node has several inputs, several such submenus will exist, one for each input, and with the same name as the respective input.

Export Selected Items to File(s): Export all the data that is currently selected in the Input Selection to one or several files. This allows the precise selection of the data you want to export. If you want to export all data, it might be advisable to use the export command on the output of the predecessor node.

Copy Selected Items to New Data Node: Copy all the data that is currently selected in the Input Selection to a new data node.

Output: A submenu for the output of the node. In cases where the node has several outputs, several such submenus will exist, one for each output.

Export to File(s): Export all data on this node's output to one or several files.

Copy to New Data Node: Copy all the data on this node's output to a new data node.

The Input and Output submenus are also available by activating the context menu directly on the little arrow icons for inputs and outputs.

2.3 Dashboard Window

A dashboard window contains information about analyses and visualizations. A dashboard window normally displays one visualization or analysis, but it is possible to coalesce several visualizations or analyses in a single dashboard window (See Workflow Explorer Context Menu). This is especially useful for visualizations, as it allows linking their visual representation, e.g. their cursors. Figure 2.5 shows a breakdown of the different areas of a dashboard window.

The leftmost part displays the input selection. The middle of the dashboard window is reserved for visualization windows. To the right the settings are displayed. The settings consist of a distinct tab for each visualization or analysis

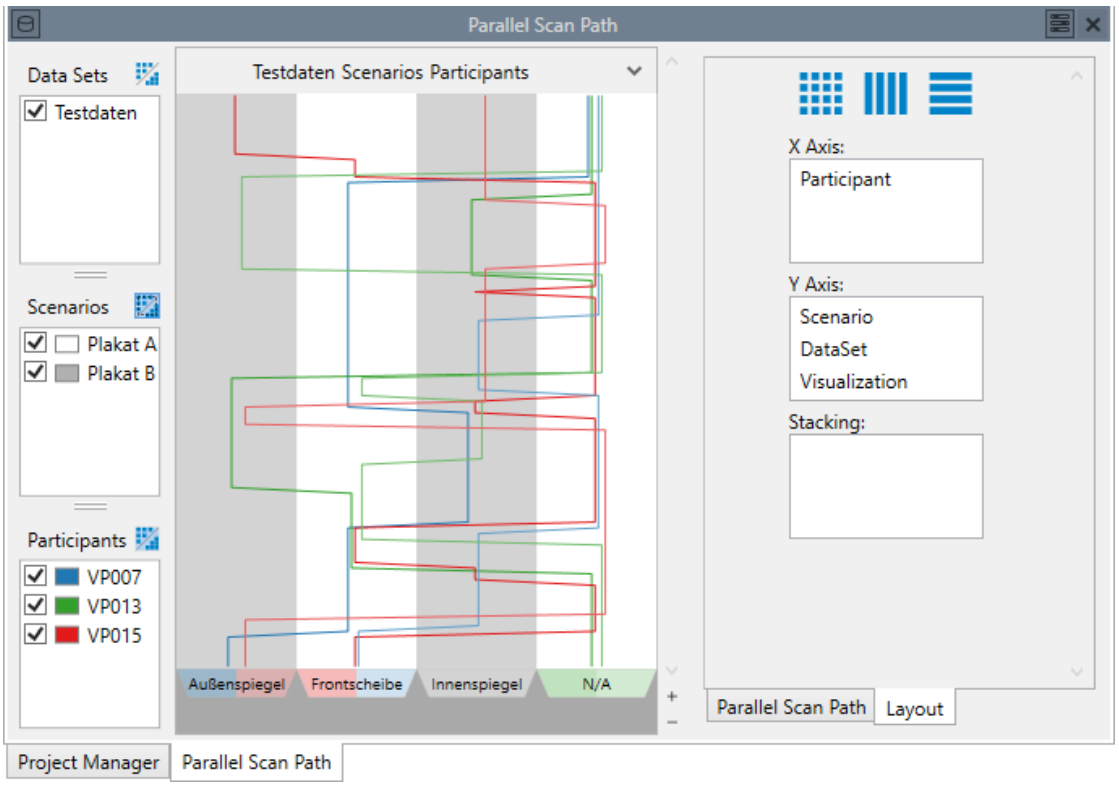

Figure 2.5 — A Dashboard Window

that is displayed in the dashboard window, plus a "Common" tab that is common to all visualizations. The input selection and the settings can be shown and hidden via buttons in the header of the dashboard window, in order to free visual space for the visualization windows. The close button in the header of the the dashboard window will close the dashboard window. The window can then be re-opened by clicking on the appropriate node in the workflow explorer. Thus, this close button is distinct from the delete button on the node in the workflow explorer, which deletes the entire node.

2.3.1 Input Selection

Figure 2.6 shows the data input selection of a dashboard window. Here you can select which data sets, scenarios and participants should be visualized, or be used as an input for an analysis. Additionally, the three buttons allow to select aggregation for data sets, scenarios and participants.

Aggregation treats the elements of the aggregated category as a single element. If you aggregate the scenarios of a heatmap visualization, you will get a heatmap

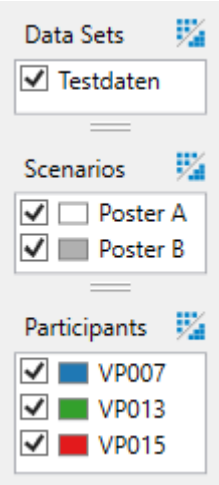

Figure 2.6 — Data Input Selection

for each participant, showing the gazes over all scenarios for that participant. If you aggregate the participants, you will get a heatmap for each scenario, showing the gazes all participants during that scenario. If you aggregate both scenarios and participants, you will get a single heatmap displaying the entire data set.

Aggregation is not possible for all visualizations and analyses. The documentation of each visualization and analysis will give a detailed description of the concrete effect of aggregation.

The context menu of the Input Selection (and the menu hidden behind the arrow next to the aggregation button) has some options for selecting and deselecting elements quicker than by selecting each data set/scenario/participant manually:

Check All: Check all data sets/scenarios/participants, regardless of which are currently selected.

Check All Selected: Check all data sets/scenarios/participants that are currently selected, in addition to those that are already selected.

Uncheck All Selected: Uncheck all data sets/scenario/participants that are currently selected.

Check Only Selected: Check all data sets/scenarios/participants that are currently selected, uncheck all those that are not selected.

Invert Checked/Unchecked: Check all data sets/scenarios/participants that are currently unchecked, uncheck all those that are currently checked.

2.3.2 Visualization Windows

The visualization windows are the central part of each dashboard window. Here, the visualizations, or the previews of analyses, are displayed. If no aggregation is active, data sets x scenarios x data sets visualization window are shown. The top of each visualization window shows which data set/scenario/participant is visualized in the window.

2.3.3 Common

The common tab contains some common settings that are applied to all Visualizations. Additionally, it allows influencing how the visualization windows are laid out in the central area. Note that any change on the layout settings here will immediately override any manual layout created by dragging and dropping visualization windows.

Each common setting is only active if at least one visualization shown in the window supports it. Refer to common settings of time-based visualizations for a detailed description of the common settings that apply to time-based visualizations.

1. Layout The Layout part of the common settings tabs allows influencing how the visualization windows are laid out.

   Layout Wizard The first of the buttons on top allows enabling the Layout Wizard, which tries to create a fitting layout for the selected visualizations. In most cases, leaving the Layout Wizard enabled is the best solution for handling multiple visualizations in the same window.

   Mixed Layout Primacy The mixed layout primacy influences how a layout wizard operates on windows that contain both time-based visualizations and other visualizations like a scan path. Figure 2.7 shows a layout where the stimulus visualization has primacy over the line graph. In figure 2.8 the line graph has primacy over the stimulus visualization.

   In specific circumstances it might be desired to more directly control the layout. Therefore it is possible to manually create a fitting layout. This layout can be set via five preselection buttons and an advance layout area. Both are only active if the Layout Wizard is disabled.

   The five remaining buttons at the top provide five often used preselections for the layout:

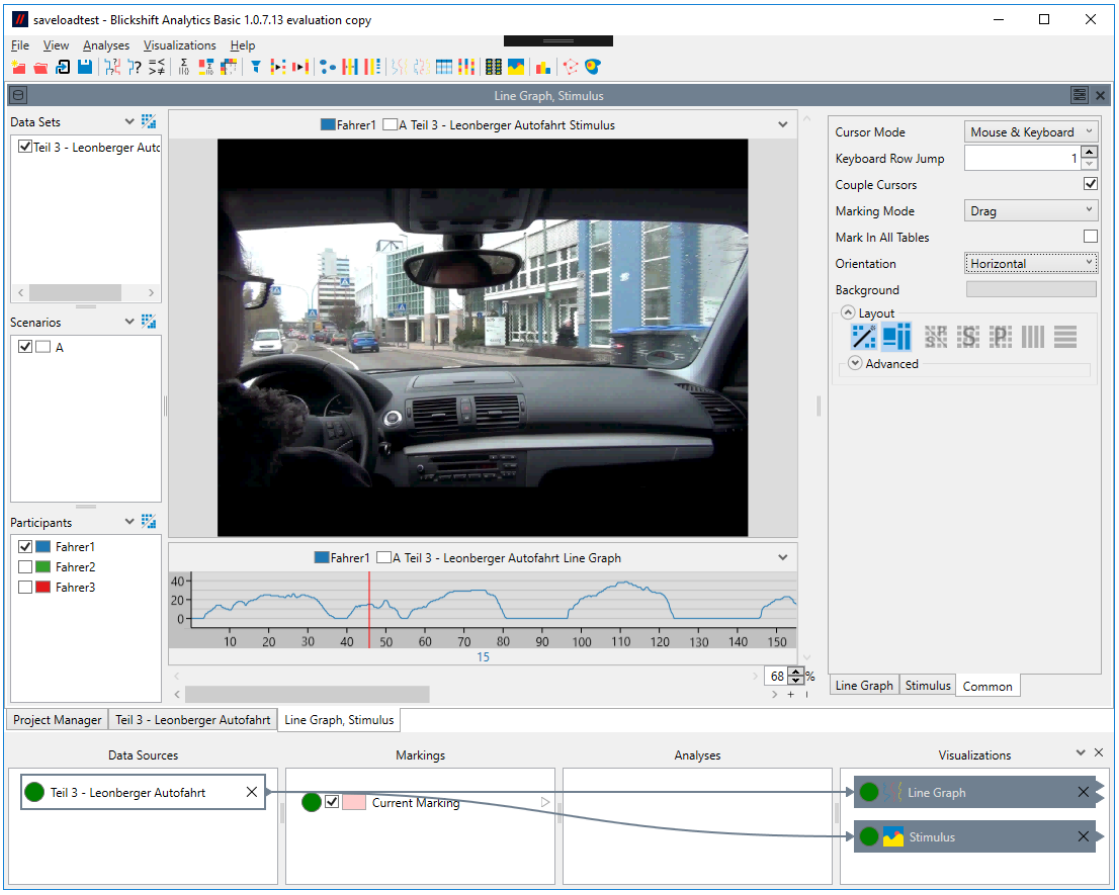

Figure 2.7 — Mixed Layout primacy enabled: Primacy is given to the stimulus, the line graph works as an addition (horizontal mode).

Scenario-Participant Grid The scenario-participant grid layout will show participants next to each other in the horizontal direction, and everything else in the vertical direction.

Scenario Grid The scenario grid layout shows scenarios in both horizontal and vertical direction and is meant for stimulus-based visualizations like heat maps in cases with many scenarios but few participants.

Participant Grid The participant grid layout shows participants in both horizontal and vertical direction and is meant for stimulus-based visualizations like heat maps in cases with few scenarios but many participants.

Horizontal The horizontal layout will show all visualization windows next to each other in the horizontal direction.

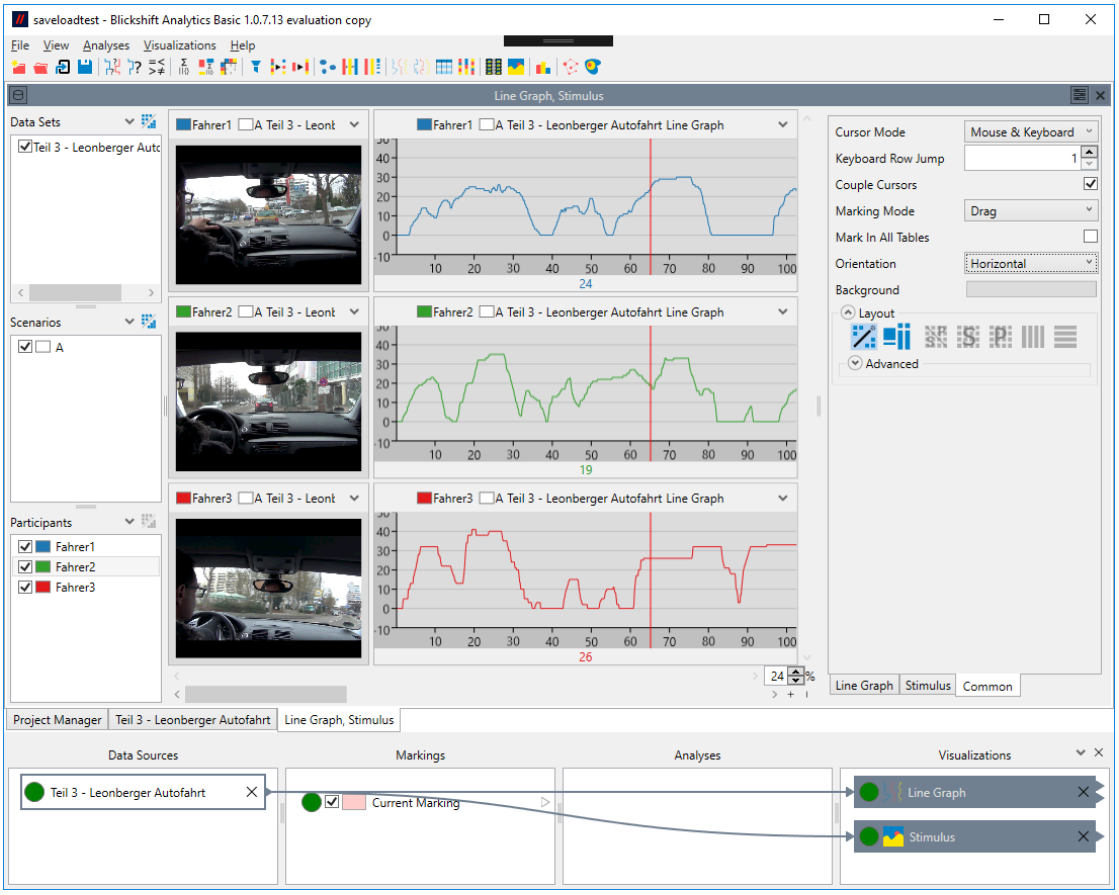

Figure 2.8 — Mixed layout primacy disabled: Primacy is given to the line graphs, with stimuli displayed as an addition.

Vertical The vertical layout will show all visualization windows next to each other in the vertical direction.

Note that the horizontal layout best fits vertically oriented visualizations (see "orientation of time-based visualizations") and vice versa.

2. Advanced Layout The Advanced Layout settings allow fine tuning of the layout and need to be explained in further detail. All five preselections can also be created via the advanced settings. In fact when clicking one of the buttons, the effect can be seen immediately in the advanced settings.

    a) Manual Layout Control There are four dimension, in which visualization windows can exist: Data Set, Scenario, Participant (as in the Input Selection), and Visualization, i.e. different visualizations displayed in the same dashboard window. This means that, if no

aggregation is activated, a visualization can exist for each DataSet / Scenario / Participant / Visualization combination. These dimension are called "visualization dimensions". They can be laid out in three dimensions: X Axis, Y Axis, and Stacked (on top of each other). These dimensions are called layout dimensions. The Advanced Layout settings makes it possible to directly influence how the visualization dimensions are mapped onto the the layout dimensions and thus how the visualizations are arranged on the screen.

For each visualization dimension except "Visualization" it is possible to assign up to three layout dimensions. In most cases, one or two layout dimensions will suffice. For example setting "Participants" to "X" only will result in all participants being shown next to each other along the X direction (Figure 2.9). Setting it to "X" and "Y" will create a two-dimensional grid of participants (Figure 2.10). By setting the "Factor" setting, it is possible to influence the ratio of the grid. The "Invert" checkbox will invert the layout direction, i.e. the last participant will be displayed first.

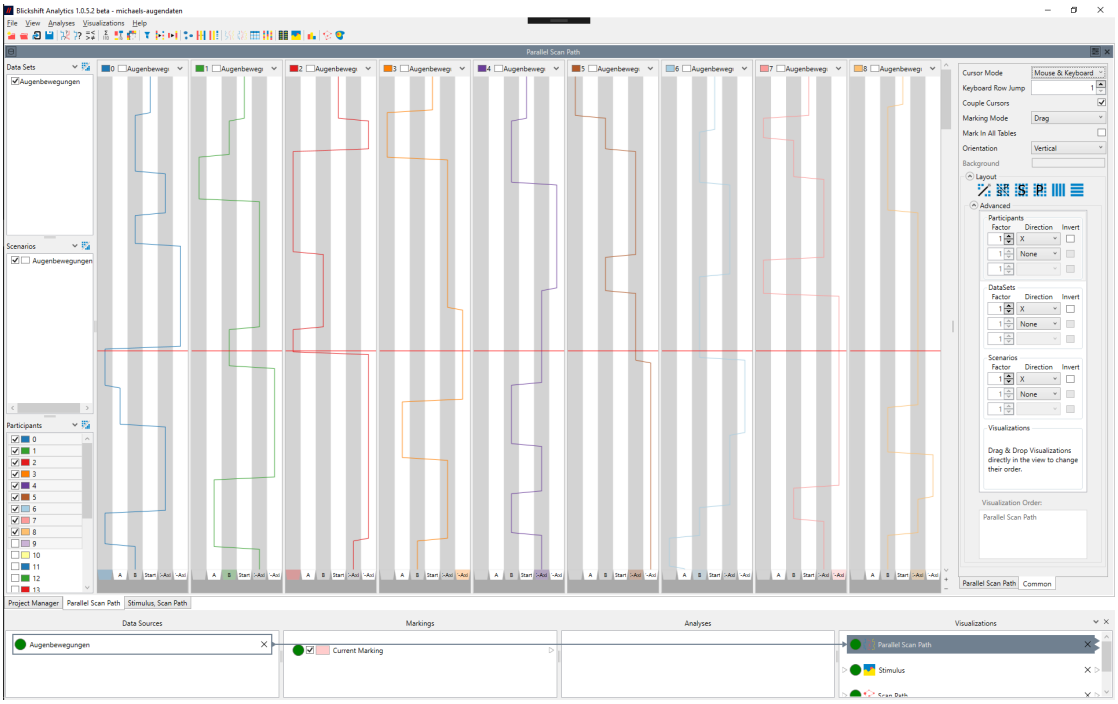

Figure 2.9 — A layout with participants laid out in X direction.

The "Visualization" visualization dimension does not have the same settings as the other visualization dimensions. The reason is that the layout of visualizations can be determined directly in the visu-

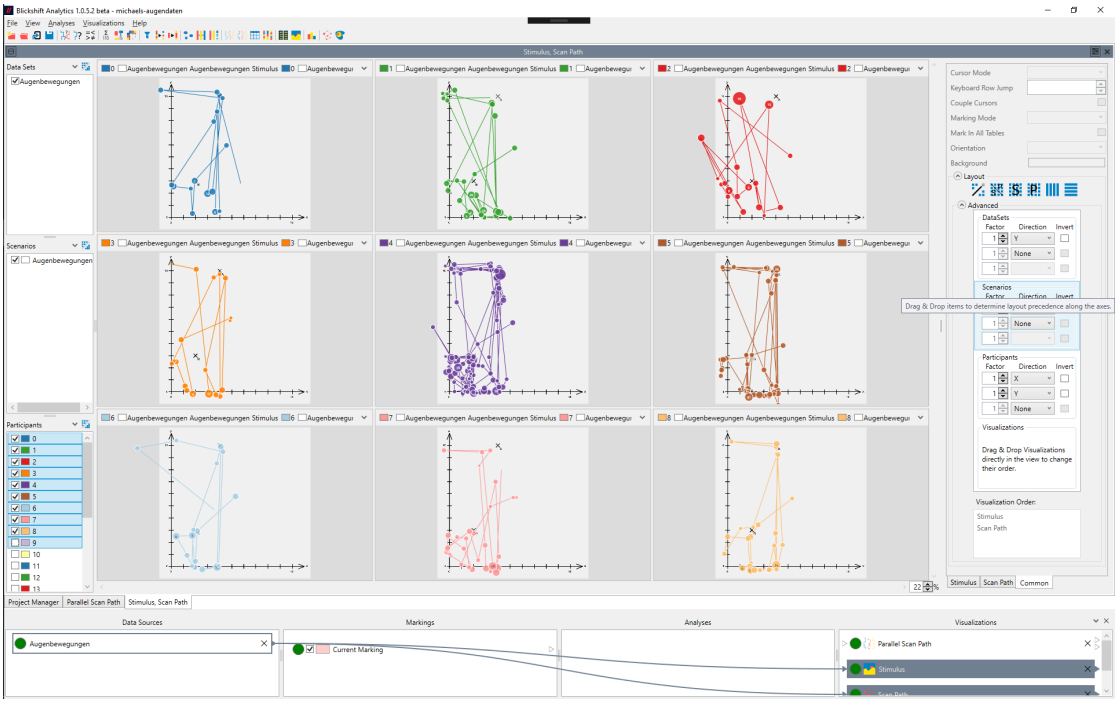

Figure 2.10 — A layout with participant laid out in X and Y direction.

alization area via drag and drop of the header of the visualization windows. Note that dragging and dropping a visualization window will always affect all windows that display the same visualization, but never affects the other visualization dimensions. If you drag & drop a window while the Layout Wizard is activated, the Layout Wizard becomes deactivated automatically. If visualizations are displayed on top of each other and you want to select one for drag & drop, click on the arrow in the header of the visualization window and select the visualization from menu that opens to initiate the drag.

Additionally, the order of the visualization window items can be changed via Drag&Drop. This order determines their precedence, e.g. whether each visualization of one participant is displayed next to each other (figure 2.11) or all participants are shown with the same visualization next to each other (figure 2.12).

b) Visualization Order In the lower part of the Advanced Layout settings, the order of displayed visualizations can be changed. This is only possible as long as the Layout Wizard is active, and determines the order of visualizations within the layout that is determined by the Layout Wizard. Figures 2.13 and 2.14 show the same visualizations laid out by the layout wizard, but with differing visualization orders.

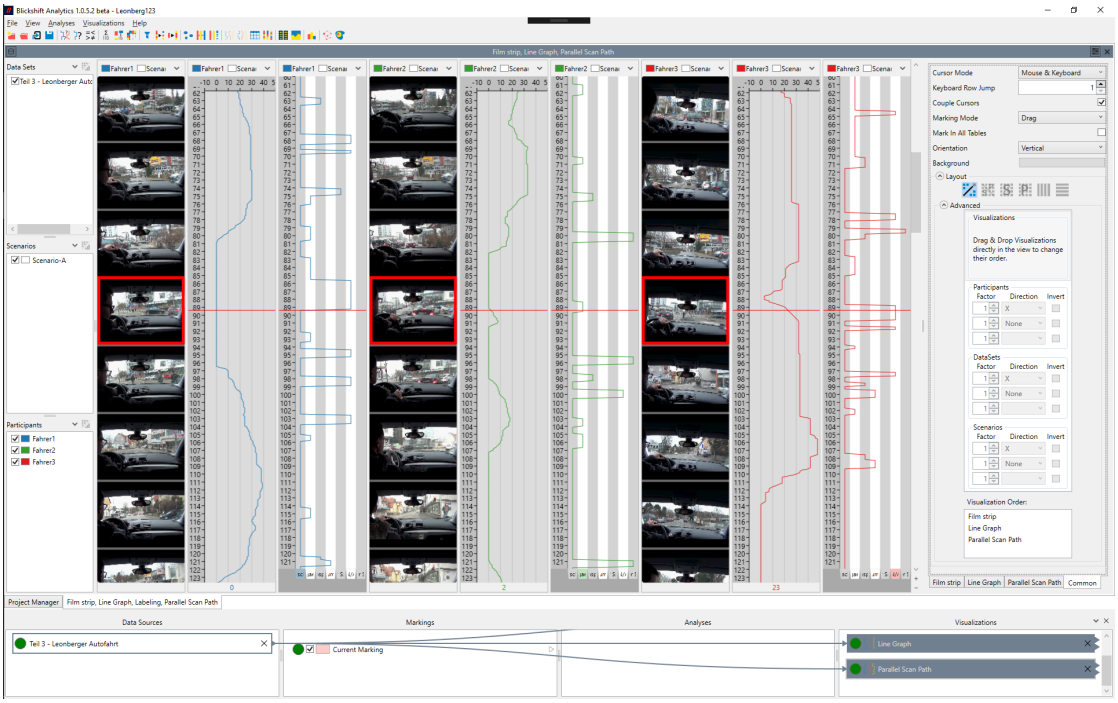

Figure 2.11 — A layout with visualization precedence: Because visualizations have a higher precedence than participants in the advanced layout section, different visualizations for the same participant are shown next to each other.

2.4 Data Import

Importing Data into Blickshift Analytics is normal the first step that needs to be done before an analysis can start. Blickshift Analytics currently imports .csv files (and related formats like .tsv), i.e. text files in which data columns are separated by a common separator symbol. To start importing, Select "File - Import Data" from the main menu, press the "Import Data" button on the main tool bar, or select "Import Data" from the context menu of the "Data Sources" column in the workflow explorer.

The import process consists of five steps, each with its own window, which are connected by "Next" and "Back" buttons.

2.4.1 Step 1: Select Input Files

On this page you can select all files that are to be imported, and you can determine how their content is mapped onto participants. It contains the following options:

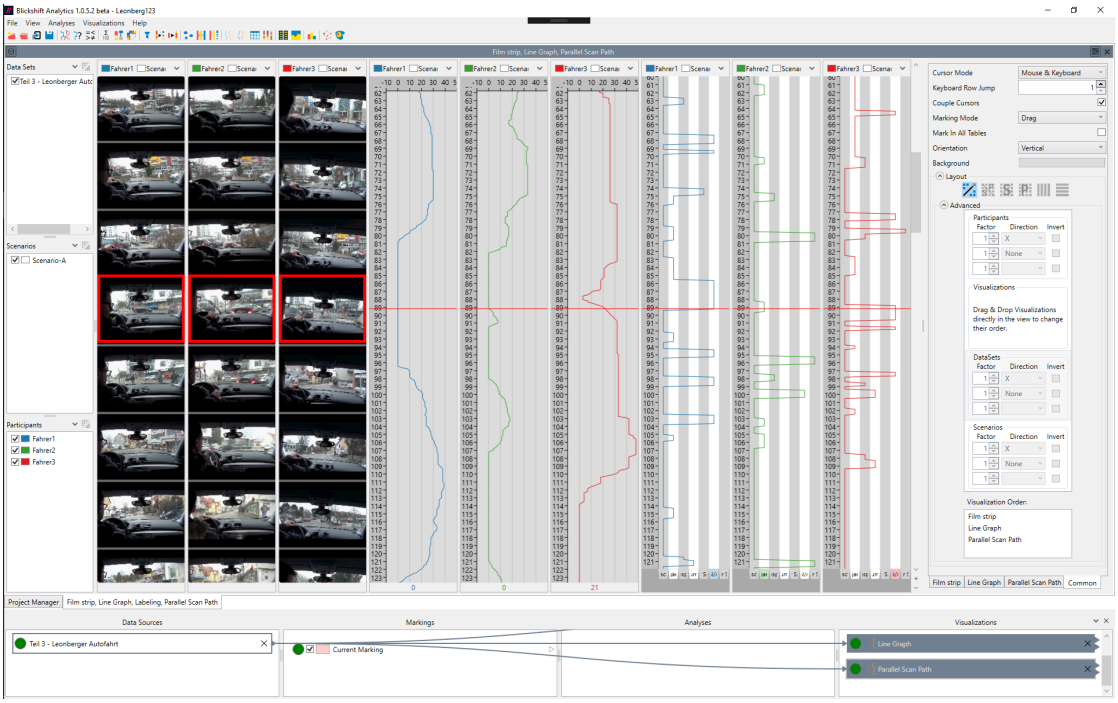

Figure 2.12 — A layout with participant precedence: Because participants have a higher precedence than visualizations in the advanced lyout section, all participants are shown with the same visualization next to each other.

Directory: Determines the base directory from which files should be imported. All data files that you want to import should be in this directory or one of its subdirectories. If you use the "Browse" button, you can select one or several files. The directory in which the files reside will be set as the base directory. If you select several files, Blickshift Analytics will try to automatically set the Filename Filter Include option accordingly.

Include Subdirectories: If selected, also import all files residing in subdirectories of the selected directory, if they fulfill the Include and Exclude filters.

Filename Filter Include: A filename filter for all the files you want to include. In the simplest case (e.g. when you select several files with the "Browse" button) this is a space-separated list of filenames. Note that the filenames need to be enclosed in quotation marks if they contain spaces. The filter can contain the wildcard character * to signal that any none, one or several arbitrary character(s) can be in its place. Let's assume you have a number of files called "Particpant1.csv", "Participant2.csv", . . . , "Participant35.csv".

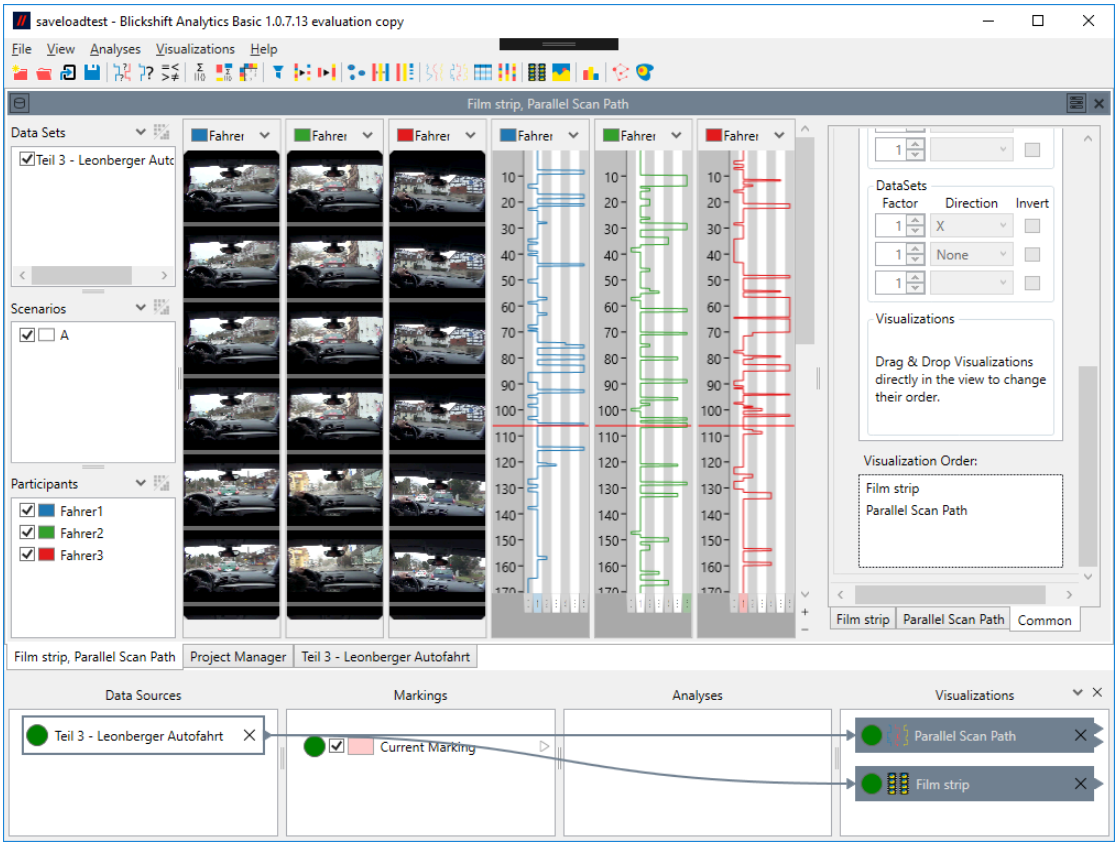

Figure 2.13 — A layout created by the layout wizard, where the film strip is above the parallel scan path in the visualization order.

If you set the include filter to "Participant*.csv", all of those files will be selected.

Filename Filter Exclude: A filename filter for all the files you want to exclude from your selection. This filter is applied after the include filter, i.e. you can "deselect" some files from all the files selected by the include filter. In the example, if you set the exclude filter to "Participant*9.csv", all the files ending with a 9 will not be included.

Scenario Name: Determines, how the data maps onto scenarios. The scenario name selection consists of a combo box and a text box. Blickshift Analytics will try to guess the correct settings from the files you have selected, but it is possible that you will need to set this manually. The combo box offers the following options:

Fixed The data consists of only one scenario. You can set the name of the scenario in the text box.

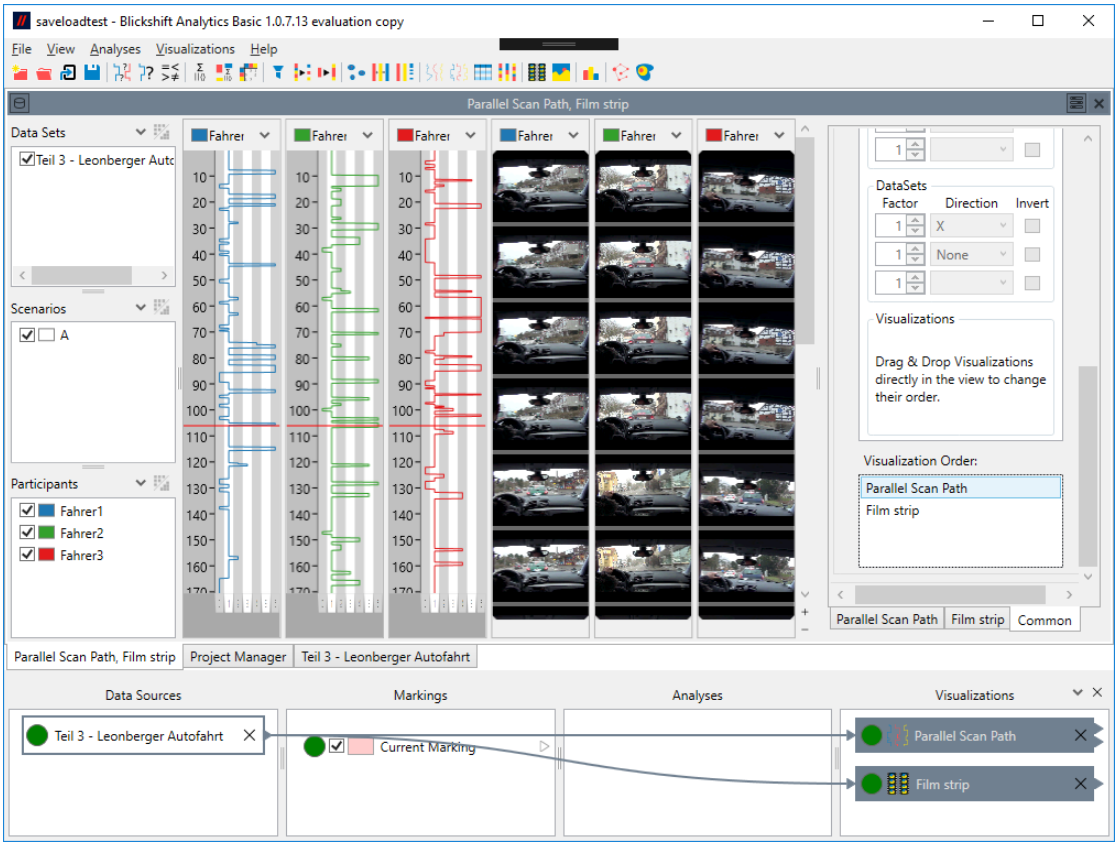

Figure 2.14 — A layout created by the layout wizard, where the parallel scan path is above the film strip in the visualization order.

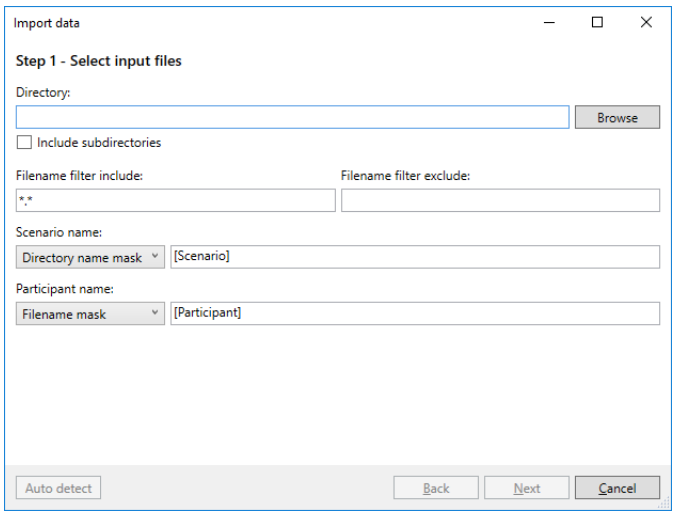

Figure 2.15 — Step 1: Select Input Files

Directory Name Mask The files reside in subdirectories, and each subdirectory (or a part of a subdirectory name) signifies one scenario. In the text box you can set how the directory names map onto scenario names. Setting it to just "[Scenario]" will name each scenario like the directory name, but you can use other characters and the wildcard * in order to restrict the scenario name to just parts of the directory name. So, if your directories are called "ABC_ScenarioX" (where X is a number), you can use "*_[Scenario]" to have the scenarios called just "ScenarioX".

Filename Mask The name of the scenarios can be found in the filename. Use the placeholder [Scenario] and the wildcard * in the text box to map the filenames to scenarios. E.g. if your filename has the pattern ScenarioX_ParticipantY.csv (where X is a number), use "[Scenario]_*" to have your scenarios called "ScenarioX".

File Header The name of the scenario can be found in the header (the first lines of) the files. In Step 2 you can determine how the scenario name is read from the file header.

Data Column One file can contain more than one scenario, but the current scenario is available in one of the data columns in the file. Note that the respective column needs to be set in Step 3.

Participant Name: Determines how the data maps onto participants. The participant name selection consists of a combo box and a text box. Blickshift Analytics will try to guess the correct settings from the files you have selected, but it is possible that you will need to set this manually. The combo box offers the following options:

Fixed The data consists of only one participant. You can set the name of the participant in the text box.

Directory Name Mask The files reside in subdirectories, and each subdirectory (or a part of a subdirectory name) signifies one participant. In the text box you can set how the directory names map onto participant names. Setting it to just "[Participant]" will name each participant like the directory name, but you can use other characters and the wildcard * in order to restrict the participant name to just parts of the directory name. So, if your directories are called "ABC_ParticipantY" (where Y is a number), you can use "*_[Participant]" to have the participants called just "ParticipantY".

Filename Mask The name of the participants can be found in the filename. Use the placeholder [Participant] and the wildcard * in the text box

to map the filenames to participants. E.g. if your filename has the pattern ScenarioX_ParticipantY.csv (where Y is a number), use "_**[Participant].**" to have your participant called "ParticipantY".

File Header The name of the participant can be found in the header (the first lines of) the files. In Step 2 you can determine how the participant name is read from the file header.

Data Column One file can contain more than one participant, but the current participant is available in one of the data columns in the file. Note that the respective column needs to be set in Step 3.

File List: A preview list of all the files that will be imported, and how they map onto scenarios and participants. This list is automatically updated each time one of the input elements above loses focus.

2.4.2 Step 2: Set Data Format

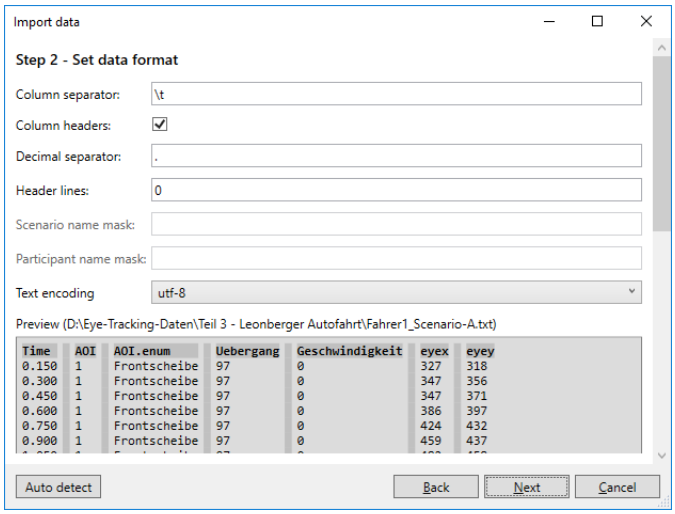

Figure 2.16 — Step 2: Set Data Format

In this step, the data format inside the files is determined. Normally, the settings of these fields are detected automatically, and a bar at the bottom shows the progress of the auto detection. If for some reason the auto detection is not running (e.g. because you have canceled it before), you can always re-start it by clicking the "Auto Detect" button on the bottom left corner. Auto detection can take a considerable amount of time, because all files are being parsed completely. If you are sufficiently sure that all files are in the same format (or you know what needs to be set in the options below), you can skip the auto detection after an initial phase by clicking the "Stop" or the "Next" button.

This page of the import dialog has the following options:

Column Separator The character that separates columns. Use \t for tab-separated columns.

Column Headers Check this, if the first line of columns does not contain data, but headers for the columns. These headers will be used as the default names for the columns in Step 3.

Decimal Separator The character used as decimal separator for non-integer numbers.

Header Lines Some files contain lines at their beginning that do not adhere to the column format of the rest of the file. Set the number of lines that should be ignored by the importer here.

Scenario Name Mask This mask is used to find the scenario name in the file header. E.g. if your file header contains the scenario name in the format "Scenario: XYZ," where "XYZ" is the scenario name, write "Scenario: [Scenario]," to read the scenario name from the file headers. This setting is only available if you have set Scenario Name to "File Header" in Step 1, and it only makes sense if Header Lines is greater than zero.

Participant Name Mask This mask is used to find the participant name in the file header. E.g. if your file header contains the participant name in the format "Participant: ABC," where "ABC" is the participant name, write "Participant: [Participant]," to read the participant name from the file headers. This setting is only available if you have set Participant Name to "File Header" in Step 1, and it only makes sense if Header Lines is greater than zero.

Text Encoding The text encoding format of the files that are to be imported.

2.4.3 Step 3: Configure Columns

In this step, it is determined how the columns are imported into Blickshift Analytics. Normally, the settings of these fields are detected automatically, and a bar at the bottom shows the progress of the auto detection. If for some reason the auto detection is not running (e.g. because you have canceled it before), you can always re-start it by clicking the "Auto Detect" button on the bottom left corner. Auto detection can take a considerable amount of time, because all files are being parsed completely. If you are sufficiently sure that all files are in the same format (or you know what needs to be set in the options below), you can

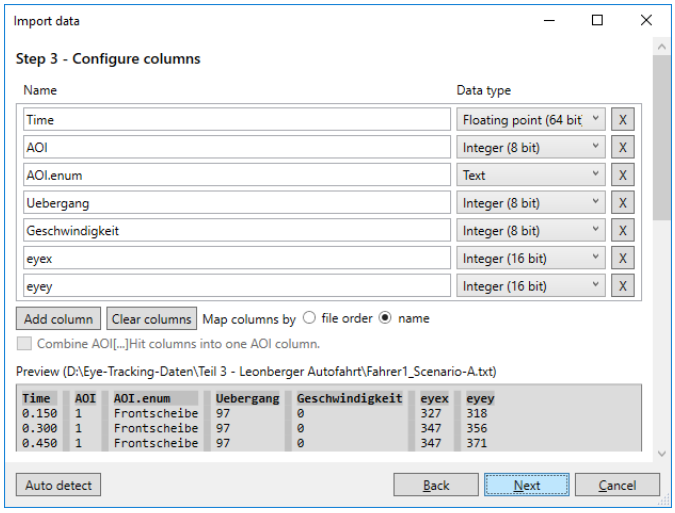

Figure 2.17 — Step 3: Configure Columns

skip the auto detection after an initial phase by clicking the "Stop" or the "Next" button.

This step provides a list of all columns that have been detected in the data, with their name and their data type. In most cases you do not need to change anything here, if auto detection has run. If "Scenario Name" or "Participant Name" has been set to "Data Column" in Step 1, the columns that contain scenarios or participants can be selected here.

2.4.4 Step 4: Stimuli

This step allows importing media files (normally stimuli) and linking them to scenario/participant combinations automatically. This step is optional, but it is helpful, if you are handling a lot of files and don't want to link the stimulus files to the scenario/participant combinations (see Project Manager: Stimuli).

The automatic import of media files assumes that their names (or paths) contain the names of the scenarios and/or stimuli that have been determined in earlier steps. If you want to import stimuli, check the "Import Stimulus Files" checkbox. The location of the stimulus files is assumed to be the concatenation of the Base Directory and the Stimulus Path Mask, in which [Scenario] is replaced by the scenario name and [Stimulus] is replaced by the stimulus name.

Let's assume you have a base folder for your experiment, called c:\experiment. Let's further assume the stimulus files are located inside subfolders named "Stimuli_for_ScenarioX" (where "ScenarioX" are the names of the scenarios determined in the previous steps), and the stimulus files in those folders

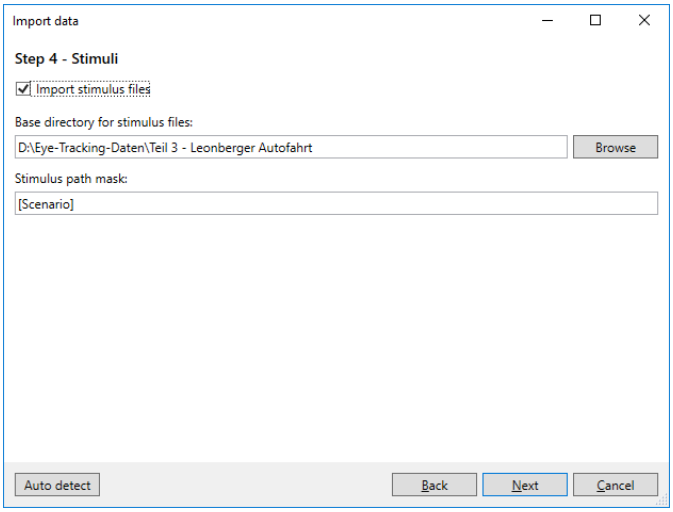

Figure 2.18 — Step 4: Stimuli

are called "Stimulus_ParticipantY.png" (where "ParticipantY" are the names of the participants determined in the previous steps). Then you would set Base Directory to "c:\experiment" and the Stimulus Path Mask to "*_*_[Scenario]*_[Participant].png".

2.4.5 Step 5: Importing Data

During this step the data is imported into Blickshift Analytics. You cannot set any options here. If there are errors during the import process, they are logged into "Messages" field. These errors can occur if you have set data formats or column types that do not conform with what is found in the files being parsed, or you have aborted the auto detection process too early and therefore wrong values persisted for data formats or column types.

2.5 Data Export

Exporting Data is possible from any node within Blickshift Analytics. You can select "Export to File(s)" either from the output submenu of the node's context menu, or directly from the output's own context menu. The submenu might be called differently, if the node has several outputs that are named specifically. In that case refer to the node's documentation in order to see what data is available on which output. This will export all the data currently available on the node's output.

You can also select "Export Selected Items to File(s)" from either the input

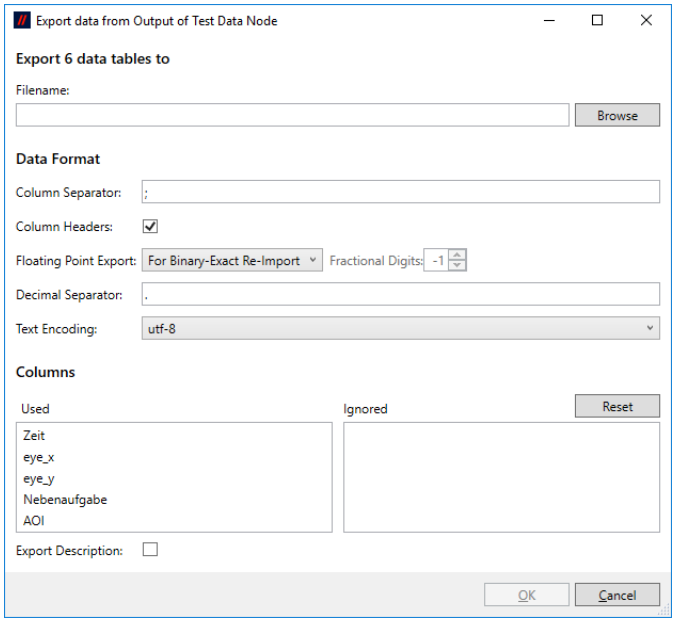

Figure 2.19 — The Options for Exporting Data

submenu of the node's context menu or directly from the input's own context menu (the name of the submenu might be different here as well, for the same reason). This allows you to export only the data that is currently selected in the node's input selection. If you want to export all the data, simply export from the predecessor node's output.

Filename: A name for the export, including the destination directory. Note that this filename will be appended by set, scenario and participant name for each exported table. Click on "Browse" to select the destination filename in a dialog.

Column Separator: The separator between the data columns. Use for tab-separated columns.

Column Headers: If selected, the names of the columns are written into the first line of the file.

Floating Point Export: Exporting floating point values from the binary representation in the computer's memory to a decimal representation in a text file is always problematic, as many binary floating point numbers do not have a terminating decimal expansion and vice versa (see Floating Point Conversion on Wikipedia for more information). Therefore, Blickshift Analytics offers two modes of floating point export, "For Human

Readability" and "For Binary-Exact Re-Import". These modes are meant for two different use cases. If you want to use the exported data for a presentation to humans, e.g. on slides, select "For Human Readability", which will round the exported numbers for human readability. If you want to continue computing on the exported files, e.g. by importing them into other programs, select "For Binary-Exact Re-Import". This will write the decimal representation in such a way that it can be parsed into the same number in binary format as existed withing Blickshift Analytics prior to export. Note that whether the re-imported number is actually the same binary might depend on the importing program's parsing.

Fractional Digits: Determines how many fractional digits are exported for floating point numbers. Use -1 to automatically determine the best representation. This option is only available if "Floating Point Export" is set to "For Human Readability".

Decimal Separator: The decimal separator for floating point numbers and DateTime and TimeSpan formats.

Text Encoding: The text encoding of the exported file.

Columns: Here you can select which columns you want to export and in which order they should appear in the created files. Use drag and drop to sort the columns or put them into the "ignored" list. Columns on this list are not exported. Press the "Reset" button to order the columns as they are in the data table.

Export Description: This option allows to export optional column descriptions into a separate file, called "[filename].columninfo.txt". Currently, these optional column descriptions can only be created via the Target Description option of the Labeling Node.

CHAPTER 3

Analyses

3.1 Sequence Analysis

The sequence analysis is able to find sequences that are common to several or all data sets / scenarios / participants. The sequences can be sequences of AOIs or any similar data, i.e. data that consists of a fixed number of different values.

The sequence analysis does not consider the length of different values, e.g. if the data consists of values AAABBBBCC, the analyzed sequence is ABC.

When analyzing sequences, there are normally two variables one is interested in: Finding long sequences, and finding sequences that appear often. These two objectives are contrary to each other. The sequence analysis node concentrates on finding long sequences. Regarding the number of appearences of a sequence, there is a setting for requiring at least k occurences. But even then the primary objective is finding long sequences, not finding sequences that appear as often as possible. So technically, the sequence analysis finds the n longest common subsequences that appear at least k times in at least x% of the input sequences.

3.1.1 Output

The sequence analysis has two outputs. The first output provides the found sequences themselves. These are also displayed in the preview. The second

output outputs a marking on the base data, marking all the occurences of the found sequences, similar to the operation of the sequence search. As with any node that outputs markings, the display of these markings can be (de)activated in the workflow explorer.

3.1.2 Aggregation

The sequence analysis is not able to aggregate, as aggregation does not have a meaningful definition in its context.

3.1.3 Settings

Column: The data column in which the sequence analysis is performed. Only enumerable columns are available for selection.

Common Subsequence Count: The number of common subsequences that should be found. Note, that the result can be a larger number of sequences than given here, as the sequence analysis will always output all results of equal length. E.g. if you set 2 as the common subsequence count, and there is one common subsequence of length 6, but 3 of length 5, the analysis will output all 4 sequences.

Min Occurence Count Per Sequence: How often a subsequence must at least occur per input sequence in order to be a possible result. Assuming that you have 1 data set with 1 scenario and 10 participants, setting this value to 3 means that all found subsequences occur at least 3 times in each participant's data.

In Min % of Sequences: Determines in how many sequences a subsequences needs to be found in order to be a possible result. The standard value of 100% means that a subsequence needs to be found in all data set/scenario/participant combinations. Assuming that you have 1 data set with 1 scenario and 10 participants, setting this value to 80% means that a subsequence needs to be found in the data of at least 8 participants in order to be considered for longest common subsequence.

Note, that the number of found results might change when you change this parameter. In the example, with the standard value of 100%, the algorithm might find 8 values of length 5. They will all be displayed, despite the Common Subsequence Count being 3. If now the value is reduced to 80%, the algorithm might find 3 sequences of length 10, that only occur for 8 participants. Then, only those 3 sequences will be displayed, thus effectively reducing the number of found results.

Input Sequence Kind: If this option is set to "Section of a Marking", and the input is a split marking, a subsequence must be found not only in each data set/scenario/participant combination, it must also be in each section of the data. If it is set to "Data Table", a subsequence is only required to exist in each data set/scenario/participant combination.

Consider the following example: Figure 3.1 shows a sequence analysis that analyzes the region marked in the parallel scan path on the left. As it does not consider the split input, two lengthy sequences are found that occur in the data of all participants. Figure 3.2 shows what happens when it is required that a sequence occurs in every section of the input: Only one short sequence can be found in all marked section. The section marked in the lower half of the blue participant gives an immediate indication of why no longer sequence could be found.

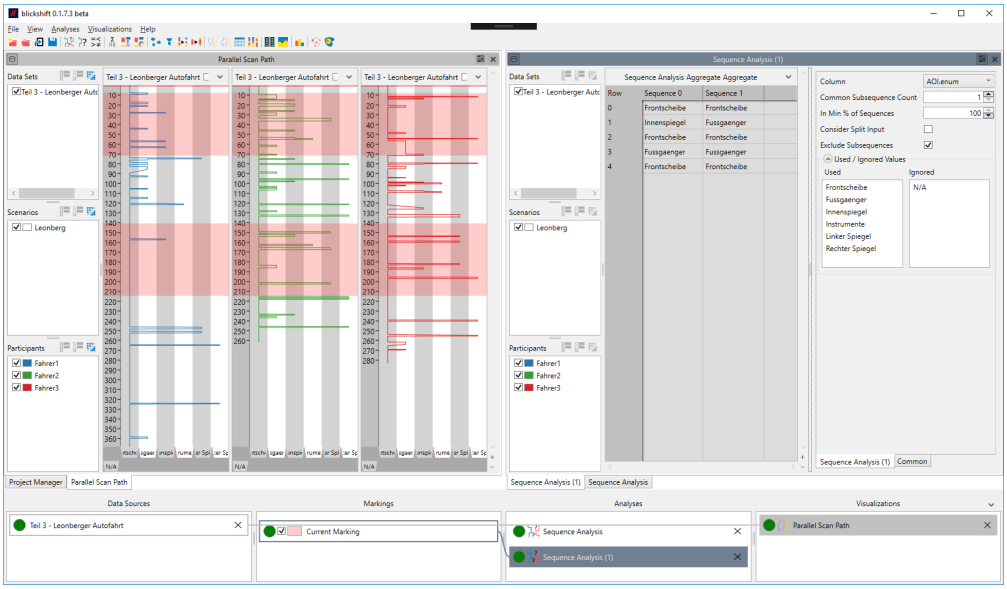

Figure 3.1 — A sequence analysis analysing the marked area, without considering split input.

Split Results: Determines whether the marking result on the second output will be a single data set or a different data set for each found sequence. Activating this option can be useful when each found sequence should be analyzed or visualized separately in the following node. Additionally, activating this option makes it possible to determine overlapping found sequences, if the result is shown in a visualization of the base data.

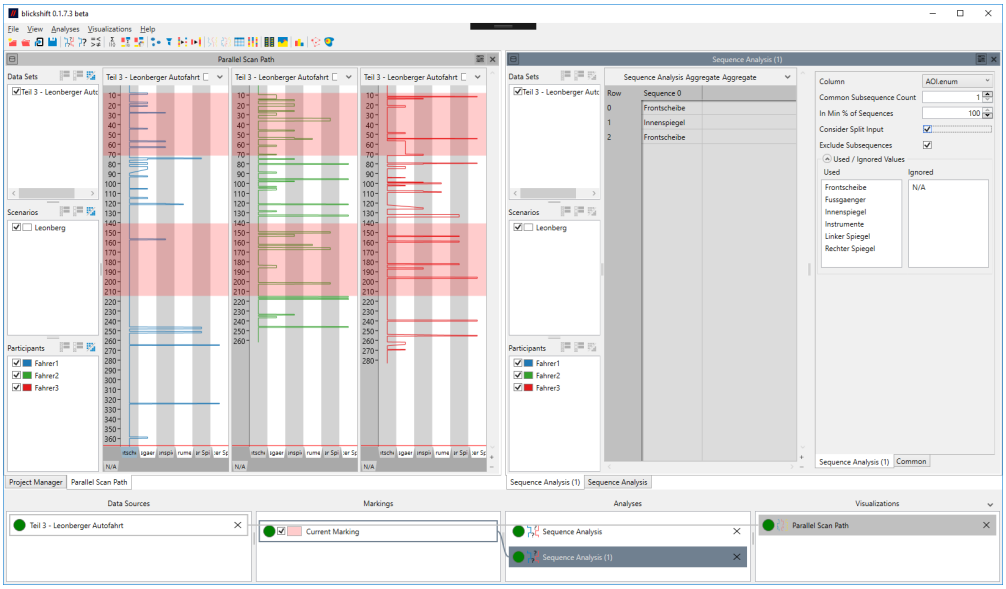

Figure 3.2 — A sequence analysis analysing the marked area, considering split input.

Ignored Values: In this option you can set some values to be ignored completely by the algorithm. E.g. if you ignore X, ABXCD will be considered to be the sequence ABCD.

3.2 Sequence Search

The sequence search is able to find the location of a given sequence in the data. The sequences in the data (and the search sequence) are considered independent of the length of their occurence, e.g. if a sequence AAABBCCCC appears in the data, a search for AABCC will find it.

The sequence search does not give a preview of its output. Instead, when you add it to the window of a time-based visualization (or create it directly therein by creating the sequence search from the context menu of the time-based visualization in the workflow explorer), the marking output is previewed in the time-based visualization itself.

3.2.1 Output

The output of the node is a marking that marks the found sequences in the data. The marking can be either visualized or further analyzed itself or shown in a

visualization that displays the data on which the search was performed (see section Markings).

3.2.2 Aggregation

The sequence search node is not able to aggregate, as searching for sequences always happens on the unchanged input data.

3.2.3 Settings

Column: The column in which the sequence is searched. You can only select columns that contain a limited number of different values.

Available: All the different values that occur in the selected column. Drag and Drop values from here to the "Search Sequence" and "Ignored" lists.

Search Sequence: This is the sequence that should be searched. Drag and Drop the available items from the list to the left into this list. The sequence has a specific order and it is possible to change the order in the list itself by drag & drop.

Ignored: This are the items that are ignored during the search. If you ignore item X and search for the sequence ABCX, even a sequence AXBXC will be found.

Split Results: Determines whether the result will be a single data set or a different data set for each found sequence. Activating this option can be useful when each found sequence should be analyzed or visualized separately in the following node. Additionally, activating this option makes it possible to determine overlapping found sequences, if the result is shown in a visualization of the base data.

Search Type: Offers a selection of the different search types. "Simple Search" will find exact matches of the selected sequence, "Levenshtein Search" offers several options for a fuzzy search, finding sequences that are similar, but not necessarily equal to the required sequence. The following parameters are only available if Levenshtein Search is selected.

Levenshtein Properties: Parameters that affect the Levenshtein search:

Max Dissimilarity: Sequences whose Levenshtein distance to the selected sequence is less than or equal to this value are found by the search. The Levenshtein distance is influenced by the following three parameters:

Insert Cost: Affects an insert operation in the data. If insert cost is X, a sequence ABD in the data has a Levenshtein distance of X to a search sequence ABCD.

Delete Cost: Affects a delete operation in the data. If delete cost is X, a sequence ABYCD in the data has a Levenshtein distance of X to a search sequence ABCD.

Replace Cost: Affects a replace operation in the data. If replace cost is X, a sequence ABYD in the data has a Levenshtein distance of X to a search sequence ABCD.

Max Length Difference: Only sequences that have difference in length of less than or equal to this value are found. This parameter is independent of the other Levenshtein parameters.

Subsequence Treatment: Determines how found sequences that are subsequences of other found sequences are treated. "Use Largest Sequence" will only report the largest sequence, i.e. the sequence that is a supersequence of the other found sequences. "Use Smallest Sequence" will only report the smallest subsequence of such a set. "Use all Sequences" will report all found sequences. Note that this last option is only relevant, if "Split Results" is active.

3.3 Value Search

The value search finds data according to specified values contained therein. In the most basic case, it finds values based on numeric comparisons (e.g. "where column X contains values larger than ..."), but it can also work with string-based comparisons.

The value search does not give a preview of its output. Instead, when you add it to the window of a time-based visualization (or create it directly therein by creating the value search from the context menu of the time-based visualization in the workflow explorer), the marking output is previewed in the time-based visualization itself.

3.3.1 Output

The output of the node is a marking that marks all areas in the data where the specified search criteria were met. The marking can be either visualized or further analyzed itself or shown in a visualization that displays the data on which the search was performed (see section Markings).

3.3.2 Aggregation

The sequence search node is not able to aggregate, as searching for sequences always happens on the unchanged input data.

3.3.3 Settings

The value search is controlled by a list of predicates, each containing a single comparison statement. You can add comparison statements by clicking the "Add Comparison" Button.

Each comparison contains four parts:

- The first part determines how the comparison relates to the previous comparison (unless it is the very first comparison). It can be either a conjunction ("and") or a disjunction ("or"). If both "and" and "or" operators are used, "and" operators always take precedence, i.e. the value search will first evaluate all "and"ed expressions and then "or" their results.
- The second part determines the Data Column on which the comparison is performed. Each comparison can be performed on a separate data column.
- The third part is the comparison type. If the selected Data Column is numerical, the comparison can be any of the standard numerical comparators: equal, unequal, less than, less than or equal, greater than, greater than or equal. Note that due to the way floating point operations work, in many cases the equality and inequality operators should only be used for integer columns.
  If the selected Data Column is of type string, the comparison can be either one of "is", "starts with", "ends with", "is not" or "contains".
- The fourth part is the value to which the column should be compared. If you have selected a numerical column, but enter a non-numeric value here, the node will signal an error.

Each comparison can be removed by clicking on its "Delete" Button.

3.4 Simple Statistics

The simple statistics node computes basic statistics for data columns.

3.4.1 Output

The output of the simple statistics node are new data tables. Each row of the data tables contain the statistics for one selected input column. The columns of the tables contain the following statistics:

Sum The sum of all values in the selected column.

Count The number of values found in the selected column.

Average The average over all values in the selected column.

Variance The variance over all values in the selected column.

Standard Deviation The standard deviation over all values in the selected column.

Sequence Count The number of continuous sequences in the selected column. This value is normally only of interest, if the input data is a split marking.

3.4.2 Aggregation

The simple statistics node is able to aggregate. Depending on the selected aggregation, the output values will reflect the sum, average, etc. of not just one participant, but all selected participants and / or scenarios / data sets.

3.4.3 Settings

Columns: The columns of the input data for which to compute the statistics. Only columns that contain numeric values exclusively can be selected. For each selected column, the resulting data will contain one row.

3.5 Gaze Duration Statistics

The gaze duration statistics node computes some statistics regarding gazes on AOIs. Thus, it presupposes the existence of AOI data, or other enumerable columns.

3.5.1 Output

The output of the gaze duration statistics node are tables containing one row for each distinct value contained in the selected AOI column. The columns of the tables contain the following statistics:

Total Gaze Duration How long was looked at the specific AOI in total.

Normalized Gaze Duration A relative value of how long the specific AOI was looked at, normalized to 1.

Average Gaze Duration How long was looked at the specific AOI in average.

Maximum Gaze Duration The longest time someone looked at the specific AOI continuously.

Minimum Gaze Duration The shortest time someone looked at the specific AOI continuously.

Gaze Count How often was the specific AOI looked at.

Time To First Fixation The time between the start of the data and the first time the participant looked at a specific AOI. Note that if you execute the gaze duration statistics on a marking, the time to first fixation is computed relative to the start of the marking.

3.5.2 Aggregation

The gaze duration statistics node allows aggregation. Depending on the selected aggregation, the output values will reflect the gaze duration statistics of not just one participant, but all selected participants and/or scenarios / data sets. In particular, the effects of aggregation are:

Total Gaze Duration The sum of all aggregated total gaze durations.

Normalized Gaze Duration The sum of all aggregated total gaze durations normalized via the sum of all aggregated gaze counts.

Average Gaze Duration How long was looked at the specific AOI in average.

Maximum Gaze Duration The longest time of all all aggregated maximum gaze durations.

Minimum Gaze Duration The shortest time of all aggregated minimum gaze durations.

Gaze Count The sum of all aggregated gaze counts.

Time To First Fixation The shortest time of all aggregated time to first fixations.

3.5.3 Settings

Time Column: The time column, used for the computation of durations. Only columns containing monotonically increasing numbers can be selected. Note that the resulting statistics are displayed in the same time unit as used in this column.

AOI Column: The column for which the statistics are computed, i.e. normally the AOI column. Only enumerable columns can be selected.

Used / Ignored Values: Determines ignored values, and additionally the sequence of the used values in the output. Use drag & drop to change.

Note, that ignoring values does not simply remove them from the output as if the rows containing these values did not exist. Instead, if an ignored value exists between equal values, the computation assumes that the ignored value should be the same value as the value between which it exists. E.g. if a sequence AAXXAAA exists, and X is ignored, the computed statistics are equal to the sequence AAAAAAA. If an ignored value exists between two values that are different, the algorithm acts as if the ignored value does not exist, as it can not be decided to which of the two values the ignored value should belong. E.g. if a sequence AAXXBBB exists, and X is ignored, the computed statistics are equal to the sequence AABBB, except for the normalized gaze duration, as this uses the total length as reference, and thus sums to less than 1.0, if a value is ignored.

This behavior is intentional and can be used to correct small detection errors. It corresponds the ignore function in the Parallel Scan Path, and the result reflects what can be seen in this visualization, if the same value is ignored. If used excessively, ignoring values can lead to results that might create incorrect impressions. If in doubt, ignoring values should not be used.

Output as TimeSpan: Decides whether computed times are output in a TimeSpan format (hh:mm:ss[.ffff]) or as a double. If Output as TimeSpan is enabled, the TimeToFirstFixation column is a string that looks like a TimeSpan. The reason is that a distinction between looking immediately at an AOI (time to first fixation = 0), and never looking at an AOI (time to first fixation = "[null]") is required. If you want to do further computations on the TimeToFirstFixation results, we recommend disabling Output as TimeSpan.

Output Unit: The time unit that is used for the output, if Output as TimeSpan is false. Note that this can only result in the correct unit, if the unit of the

Time Column is set in the source Data Node. If the Output Unit is set to "None", the result is output in the same unit as the Time Column.

3.6 Transition Matrix

The transition matrix node computes a transition matrix. It presupposes the existence of AOI data, or similar enumerable columns.

3.6.1 Output

The transition matrix node outputs tables that represent a transition matrix. Each row represents one AOI, and shows how many transitions from this AOI to another AOI (displayed per column) have been made.

3.6.2 Aggregation

The transition matrix node allows aggregation. Depending on the selected aggregation, the transition matrix represents the transitions not for just one participant but for all selected participants and/or scenarios / data sets.

3.6.3 Settings

Column: The column used to compute the transition matrix, normally the AOI column.

Used / Ignored Values: Determines ignored values, and additionally the sequence of the used values in the output. Use drag & drop to change.

Note, that ignoring values does not simply remove them from the output as if the rows containing these values did not exist. Instead, if an ignored value exists between equal values, the computation assumes that the ignored value should be the same value as the value between which it exists. E.g. if a sequence AAXXAAA exists, and X is ignored, the computed transition matrix will not display any transition as the assumed sequence is AAAAAAA. If an ignored value exists between two values that are different, the transition matrix acts as if the ignored value does not exist, i.e. it will display a transition from one value to the other. E.g. if a sequence AAXXBBB exists, and X is ignored, the transition matrix will display a transition from A to B.

This behavior is intentional and can be used to correct small detection errors. It corresponds the ignore function in the Parallel Scan Path, and

the result reflects what can be seen in this visualization, if the same value is ignored. If used excessively, ignoring values can lead to results that might create incorrect impressions. If in doubt, ignoring values should not be used.

3.7 Filter Node

The filter node is a very basic node that outputs the selected data sets, scenarios and participants of its input. As a result any data set, scenario or participant not selected in this node is not selectable in all following nodes in the graph. It can therefore be helpful to use a filter node, if you have a lot of data sets, scenarios or participants, but want to examine only a subset of them simultaneously.

3.7.1 Output

The filter node outputs all data sets, scenarios and participants selected in its input selection.

3.7.2 Aggregation

The filter node does not allow aggregation as it only filters the data but never changes it.

3.7.3 Settings

The filter node does not have any settings. Select the desired data sets, scenarios and participants in the input selection.

3.8 Marking Adaptation

The marking adaptation node creates a marking based on an already existing marking, e.g. by adding a certain amount of time at the start or end of each sequence in the marking. Consequently, the input for a marking adaptation node must always be a marking.

A common use case is to mark a certain time before a specific signal, e.g. if you want to investigate the behaviour of the participants before a certain event. In such a case, use a sequence search or value search to find the signal, and then adapt the output of the search node with a marking adaptation.

3.8.1 Output

The output of a marking adaptation is always a marking. The marking adaptation does not change the input marking, it creates a new marking based on the input marking.

3.8.2 Aggregation

The marking adaptation node does not allow aggregation, as the output marking always has to refer to the same base data as the input marking.

3.8.3 Settings

Extension Unit: Extend by a specific time or by number of data columns.

Time Column: The time column used for time-dependent adaptations. Note that the following options use the same time unit as this column, when "Extension Unit" is "Time". When this column contains a time in a TimeStamp or DateTime format, the following options use seconds as their time unit.

Mark Before Start: Mark this much time / this many data rows before each sequence in the marking.

Mark After Start: Mark this much time / this many data rows after the start of each sequence in the marking.

Keep Middle (Source): If selected, keep the source sequences, else discard them. If this option is selected, "Mark After Start" and "Mark Before End" are deactivated, because the data rows that would be marked by those options are already included in the source rows.

Mark Before End: Mark this much time / this many data rows before the end of each sequence in the marking.

Mark After End: Mark this much time / this many data rows after each sequence in the marking.

3.9 Group Node

The group node can be used to group data according to a specific enumerable column. As an example, the data might contain a column "task", containing the values "task1", "task2" and "task3", and each participant might have performed

that specific task at a different point in time, or even returned to a task after performing another task in between.

Now, if you want to compare how the different participants performed during a specific task, this would be difficult to achieve without the group node. With the group node, it becomes possible to just look at the data from a specific task for each participants, regardless of when the task was performed.

3.9.1 Output

The group node outputs the same data as on its input, but grouped according to the selected column. The output has a distinct data set for each value in the selected column. If more than one column is selected, there will be a distinct data set for each existing combination of the values of the selected columns. So in the example above, the output would contain a data set for "task1", "task2" and "task3", each containing only the data where the task column of the input data had the respective value.

3.9.2 Aggregation

The group node is not able to aggregate, because it only performs a reorganization of the input data, and thus a computed aggregation can not exist.

3.9.3 Settings

Data Columns: The input data is grouped according to the values contained in these columns.

3.10 Fixation Computation

The fixation computation node can be used to compute fixation data from raw eye movement data. In many cases, fixation data already exists in the data, and in that case, there is normally no need to use this node. However, if your data does not contain fixation data, and you want to use a visualization that requires fixation data (e.g. a scan path), then using this node is required.

This node implements the dispersion threshold identification algorithm described by Salvucci and Goldberg[1].

[1] Dario D. Salvucci and Joseph H. Goldberg, *Identifying Fixations and Saccades in Eye-Tracking*, http://dx.doi.org/10.1145/355017.355028

3.10.1 Outputs

The fixation computation node computes fixation data from the input node and adds the computed data as extra columns to input data. Thus the output consists of the same columns as the input, plus these additional three columns:

FixationNumber The fixations are consecutively numbered in this column, starting with 1. If no fixation was detected at a specific moment in time, this column will be 0 at that time.

FixationX The x coordinate of the fixation, as computed from the input coordinates.

FixationY The y coordinate of the fixation, as computed from the input coordinates.

3.10.2 Aggregation

The fixation computation node is not able to aggregate, because it only adds columns to the input data and thus a computed aggregation cannot exist.

3.10.3 Settings

Eye X Column: The X coordinate column.

Eye Y Column: The Y coordinate column.

Fixation Index Column Name: The name for the fixation index column.

Fixation X Coordinate Column Name: The name for the fixation x coordinate column.

Fixation Y Coordinate Column Name: The name for the fixation y coordinate column.

Ignore (0, 0): Ignore (0, 0) coordinates. If selected, a (0, 0) coordinate will not break a fixation. This can be useful, if the data maps many undetected eye positions to (0, 0), but should be used carefully, as it may lead to incorrect fixation detection.

Maximum Dispersion: The maximum dispersion.

Minimum Rows: The minimal fixation length, in data rows.

3.11 Column Computation

The Column Computation node makes it possible to produce new columns depending on the values in the existing columns. It is the simplest freely programable node in Blickshift Analytics, because a value can only depend on other values in the same row. This makes it very easy to create a new column that contains e.g. the sum of two columns, or the values of different columns depending on the content of another column. However, more complicated computations that also take previous or following rows into account are not possible.

Each new column is produced by a user-defined C#-function that has the input columns as parameters and must return the value of the new column.

The computation of the new column is started as soon as the focus leaves the code input area. If the provided code contains syntax errors, they will be listed at the bottom of the settings pane. Double clicking on an error message will automatically set the cursor to the position of the error.

The code automatically includes the System namespace of Microsoft's .net API. This means that you can e.g. access the square root function in the Math namespace by using "Math.Sqrt(x)". The most commonly used functions for the column computation node besides simple mathematical operators are those found in the Math namespace (for mathematical operations), and in the String class for manipulating strings.

3.11.1 Output

The node outputs all incoming columns plus a number of additional columns containing the results of the computations set by the user.

3.11.2 Aggregation

The column computation node is not able to aggregate, as all input tables are output with additional columns.

3.11.3 Settings

Columns: The columns that are used in the creation of the new columns.

Input Column Aliases: Alias names for each selected column. The alias names are used as function parameters and are sometimes required in order to ensure valid identifiers. They can also be used to shorten overly long column names for more convenient use in the function.

Output Columns Properties Each output column is defined by three properties: Its type, its name and the function code. In the UI these properties are presented in the form of a C# function. This function has parameters with the names given in the Input Column Aliases and must return a value of the specified type. This function is applied to each row of the data, the values in that rows are passed to the appropriate parameters of the function and the return value written into the newly created column.

3.12 Labeling

The Labeling Node makes it possible to add one or more additional columns to the data and manually fill those with new data.

Labeling differs from most other nodes in that it does not compute data but uses data set by the user, i.e. you. Every labeling action you trigger adds to or changes the already existing data without completely recomputing the output. The settings control of the labeling node does not display settings that are automatically applied to the node upon recomputation, instead it offers controls for the labeling operations.

3.12.1 Output

Labeling has one output that outputs all the data available at its input, plus as many columns as the you created in this node.

3.12.2 Aggregation

The labeling node is not able to aggregate, as all data at its input is also forwarded to its output.

3.12.3 Settings

Columns: The topmost area of the settings displays the columns created by the labeling node and the currently selected target column. You can add new columns, and change their name and data type. Once a labeling operation has been performed on a column, its name and data type become fixed and cannot be changed any more. Exactly one column is always selected as a current target. All labeling operations will be performed on that column.

Target Description: This is an optional description for a column. If you write something into this text field, the description of the current target column

will be added (or replaced) the next time you start a labeling operation. These column descriptions can be exported via the data export.

Target Marking: The target marking determines, which area of a column will be labeled by the next labeling operation. Most of the time the target marking will be the current marking. Note that in many cases Couple Cursors should be set to false on the Common Settings tab. In addition to the current marking, any marking that is set on either the input or the output data of the labeling node will work, making it possible to e.g. label all rows found by a value search. A labeling operation will set all rows of the target column selected by the target marking to the label value.

Quick Labels: The quick labels are a number of labels you can set and then use via a button or via keyboard shortcuts in Keyboard Mode. Quick Labels are particularly helpful, if a column is to be labeled with a limited number of lengthy labels: They can be used as labels with a single keystroke instead of entering them repeatedly.

The labeling node has three modes of operation:

Standard Mode Labeling is only possible with the buttons next to the quick labels. This is the default mode, but only because the keyboard grabbing of the other modes can interfere with the expected behavior of other parts of Blickshift Analytics.

Quick Label Mode Allows labeling with the keyboard shortcuts displayed next to each quick label. The number keys can be used to change the target column or (Ctrl+Number) the target marking. Additionally, pressing Return will pop up a small window that allows entering labels not present in the list of quick labels. Pressing Ctrl+Return will switch to Immediate Labeling Mode.

Immediate Labeling Mode Allows directly entering data into the marked areas. A labeling operation is finished by simply selecting another area. Pressing Esc will return to the previously active mode.

CHAPTER 4

Visualizations

4.1 Time-Based Visualizations

Time-based visualizations are visualizations that display data along a time axis. The time axis can be displayed either vertically or horizontally.

The available time-based visualizations are:

- Line Graph
- Parallel Scan Path
- Drill Down
- Bar Graph
- Film Strip

4.1.1 Interactions

All of these visualizations have several interactions in common. Some of these interactions can be configured on the Common Settings Tab of the Dashboard Window.

Scrolling Use the mouse wheel or the scroll bar to scroll the view.

Zooming Use Ctrl + mouse wheel or the +–Buttons in the bottom right corner to zoom.

Marking In the default mode, click and drag to set a certain area as the current marking. Ctrl-click and drag to add a certain area to the current marking. Alt-click and drag to remove a certain area from the current marking. How marking works exactly can be heavily influenced by the specific common settings, i.e. "cursor mode", "couple cursors", "marking mode" and "mark in all tables".

Note that, if cursors are coupled, the marking will be changed in all displayed visualizations. If the cursors are not coupled, the marking change will only occur in the visualization in which the click and drag operation is performed.

Shifting If visualizations are not aggregated, it is possible to shift a data table in comparison to the other tables. Use Shift + click and drag to shift a displayed table. Note that shifting does not only influence the visualization but actually changes the base data, meaning that all analyses and visualizations that are connected to the source data node are affected by such a change. This can be useful, if certain events in the data need to be aligned.

4.1.2 Common Settings

There are some common settings that apply only to time-based visualizations.

Cursor Mode Determines, how the cursor in the visualizations can be moved. By standard, it can be moved by both mouse and keyboard. It can also be set to react to either mouse or keyboard input exclusively. This is especially helpful if precise operations with the keyboard need to be done and moving the mouse interferes with the cursor. Please note that in keyboard mode, the visualization needs to have keyboard focus, which can be achieved by clicking on the desired visualization with the mouse. In Keyboard mode, the cursor can be moved with the cursor keys. Holding down the Shift key will create a marking. Holding Ctrl+Shift will add to an existing marking, holding Alt will remove from an existing marking.

Keyboard Row Jump The cursor is advanced this many data rows for each keyboard press (cursor keys).

Couple Cursors Couple the cursors in all visualizations in the dashboard window. If active, the cursors in all windows will be at the same position in the data all the time. Deactivating this options makes it possible to mark different areas in different visualization windows.

Marking Mode Determines, how markings are created when using the mouse. If the marking mode is "Drag", markings are created by dragging the mouse while holding down a mouse button. If the marking mode is "Click", the area between two consecutive clicks in a visualization gets marked.

Mark In All Tables If this option is activated, a current marking created in a visualization applies to all tables, not only those that are currently visible. Let's assume you have five participants, and two are currently selected in the input selection of a Line Graph. If this option is disabled, marking an area in the linegraph will create a marking on those two participants. If this option is enabled, the same data rows of all five participants will be marked. This option is only available if Couple Cursors is true.

Orientation Configure the orientation of the visualizations. "vertical" will display the time along the vertical axis, "horizontal" will display the time along the horizontal axis.

Background Configures the background color of the visualizations. This setting applies only to visualizations that do not draw their own backgrounds, such as line graphs and drill down. The background is not drawn by the visualization itself, and therefore it becomes possible to "manually aggregate" visualizations by dragging and dropping two visualization window onto each other. This is not possible for visualizations that have to draw the background themselves, because the topmost visualization will draw over the lower visualizations.

4.1.3 Marking Context Menu

Time-based visualizations (with the exception of the film strip) display markings, if they are enabled in the workflow explorer. These can be the current marking, other stored markings or markings ouput by analysis nodes. The context menu of a visualization contains a submenu for each marking under the mouse, containing the following items:

New Analysis Create a new analysis that has the marking node as an input, i.e. that analyzes only the data covered by the marking.

New Visualization Create a new visualization that has the marking node as an input, i.e. that visualizes only the data covered by the marking.

Set as Current Marking Change the current marking to cover exactly the same data as this marking. This option is only available if the marking is not already the current marking.

Duplicate Create a new marking node in the markings column of the workflow explorer that outputs a marking covering exactly the same data as this marking.

Hide Marking Hide the display of this marking from all visualization windows. This option has the same effect as unchecking the checkbox of the node that originates this marking in the workflow explorer.

Delete Source Node Delete the node that originates this marking. This has the same effect as deleting the node in the workflow explorer.

Export to File Export the data covered by this marking to one or several files.

4.2 Line Graph

The line graph is a simple visualization that plots numeric values along a time axis.

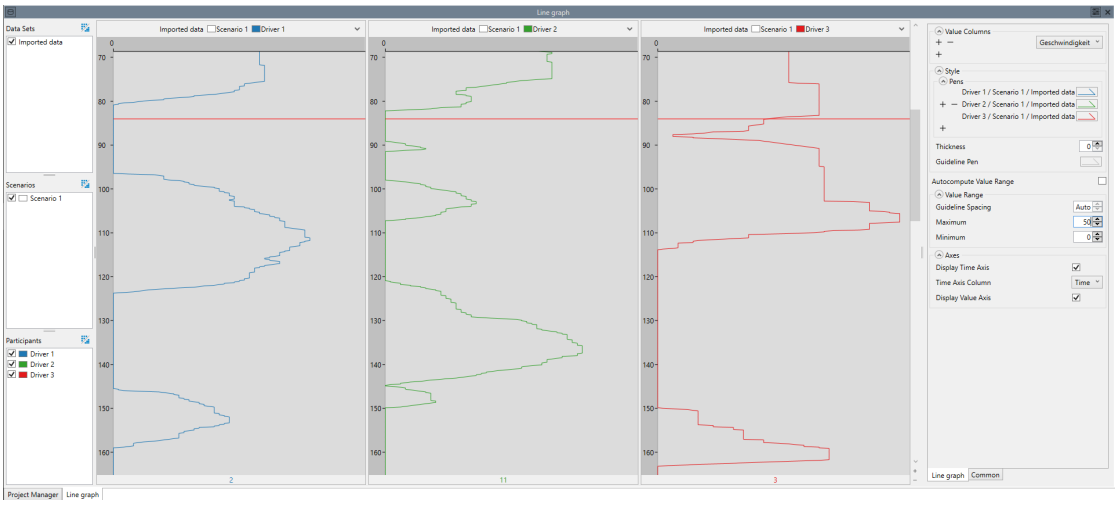

Figure 4.1 — A Line Graph Visualization

4.2.1 Aggregation

Aggregating a line graph displays the graph of all participants and/or scenarios / data sets in the same window.

4.2.2 Settings

Value Columns: The columns displayed as lines in the graph.

Style: Parameters that determine the look of the visualization:

Pens: The style used to draw the lines.

Thickness: The thickness of the lines that are drawn. If the value is 0, the thickness of each line can be set individually in the pen dialog of each line. Otherwise this value overrides the individual values and can be used to quickly change all lines.

Guideline Pen: The style used to draw the guidelines, i.e. the lines drawn at specific intervals.

Autocompute Value Range: Automatically compute minimum, maximum and linespacing. This will ensure that the entire value range of the graph is visible.

Value Range: Determines the range of values that is displayed. Disable Autocompute Value Range to enable these settings. This can be useful if you want to take a more detailed look at a specific value range, and the minimum / maximum values of the entire graph need not necessarily be visible. The value range consists of the following parameters:

Minimum: The smallest value that is displayed. Values smaller than this minimum are cropped.

Maximum: The largest value that is displayed. Values larger than this maximum are cropped.

Guideline Spacing: The distance between two guidelines. This value is restricted by the Maximum and Minimum settings, because not too many guidelines can be drawn at once.

Axes: The parameters that influence the display of axes:

Display Time Axis: Determines, whether the time axis is displayed.

Time Axis Column: The column used for the labels of the time axis.

Display Value Axis: Determines, whether the value axis is displayed.

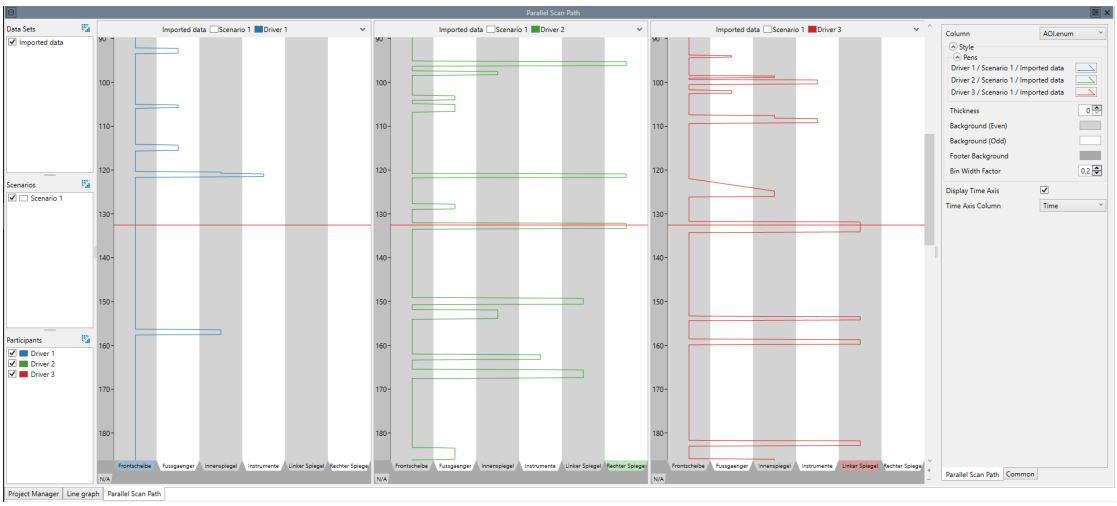

Figure 4.2 — A Parallel Scan Path Visualization

4.3 Parallel Scan Path

The parallel scan path (PSP) displays the progression of AOIs (or similar values) through time. Figure 4.2 shows a parallel scan path.

The parallel scan path visualization allows some user interaction: The AOIs can be sorted by dragging and dropping the tabs displaying their names. Additionally, a right-click on a tab opens a context menu that allows ignoring the AOI (similarly, an ignored AOI can be re-enabled). Ignoring an AOI will result in slanted connections, if the ignored AOI was between two different AOIs. However, if an ignored AOI existed between a single AOI, the result is a straight line, as if this single AOI was never interruped (by the ignored AOI). This behavior is intentional, and can be used when the Eye-Tracking data had intermittent failures (e.g. by the participant blinking) that were recorded as a distinct AOI. However, if used with real AOIs, ignoring can lead to misleading visualizations and thus should be used cautiously.

4.3.1 Aggregation

Aggregating a Parallel Scan Path displays all PSPs of all participants and/or scenarios / data sets in the same window. Note that the display of aggregated PSPs can be tuned with the parameter Bin Width Factor.

4.3.2 Settings

Style: Parameters that determine the look of the visualization:

Pens: The styles used to draw the parallel scan paths.

Thickness: The thickness of the lines that are drawn. If the value is 0, the thickness of each line can be set individually in the pen dialog of each line. Otherwise this value overrides the individual values and can be used to quickly change all lines.

Background (Even): One of the two colors used for the background.

Background (Odd): The other of the two colors used for the background.

Footer Background: The color for the background of the footer.

Bin Width Factor: The bin width used when aggregating several parallel scan paths. If this value is 0, the aggregated scan paths are drawn on top of each other. If it is 1, the entire width of the column is used to draw scan paths.

Display Time Axis: Determines, whether the time axis is displayed.

Time Axis Column: The column used for the labels of the time axis.

Used / Ignored Values: Determines the values to be ignored in the visualization as well as the sequence of the used values. Use drag & drop to change.

If ignored values exist between two other distinct values, in place of the ignored value a slanted line is displayed in the visualization. If an ignored line exists between values that are equal, this value is used and a straight line is shown, as if the ignored value didn't exist at all. This is intentional, as it allows filtering out unwanted noise, but should be used sparsely, as it can result in parallel scan paths that have nothing in common with the base data, if used excessively.

4.4 Drill Down

The drill down node displays data in a spreadsheet-like way. It does not use any abstraction, but instead directly displays the source data row by row. As the display is only readable when a row has enough space to display a line of text, it is often necessary to zoom in, before anything becomes visible.

4.4.1 Aggregation

The drill down node cannot aggregate as it is a direct display of the input source data.

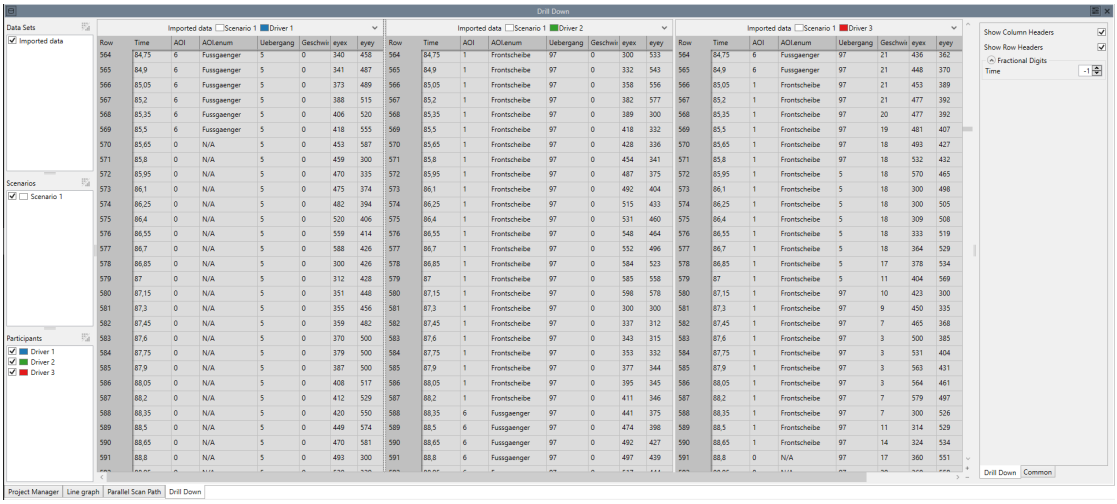

Figure 4.3 — A Drill Down Visualization

4.4.2 Settings

Show column headers: Display the headers of the columns.

Show row headers: Display the headers of the rows.

4.5 Bar Graph

The bar graph is a visualization that displays enumerable columns as colored bars, where each color represents one value.

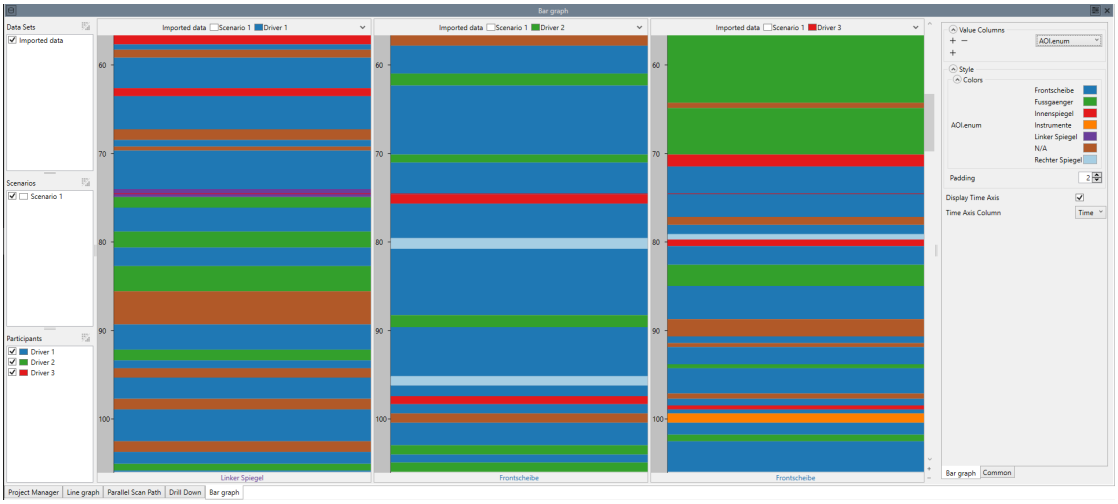

Figure 4.4 — A Bar Graph Visualization

4.5.1 Aggregation

The bar graph is not able to aggregate, as each column requires the entire space of the window.

4.5.2 Settings

Value Columns: The data columns for which should be displayed by the bar graph.

Style: Parameters that determine the look of the visualization:

Colors: The colors representing the values in the selected columns.

Padding: The space between the columns if more than one data column was selected.

Display Time Axis: Determines, whether the time axis is displayed.

Time Axis Column: The column used for the labels of the time axis.

4.6 Film Strip

The film strip is a visualization of a video file. By default, it displays the video that is associated with the data (See Add to Window" for information on how to couple visualizations.

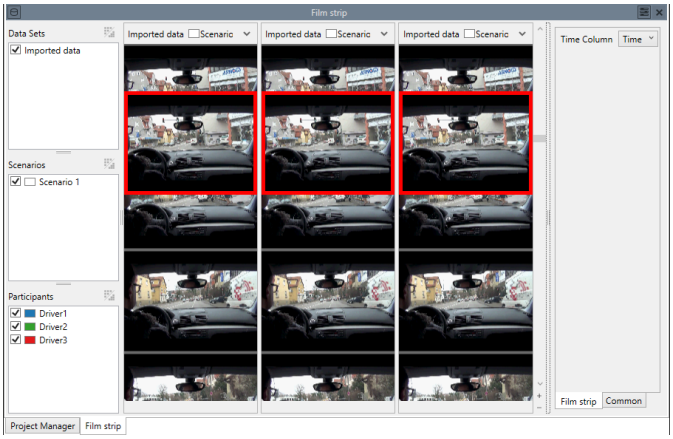

Figure 4.5 — A Film Strip Visualization

4.6.1 Aggregation

The film strip is not able to aggregate, as each film strip requires the entire space of its window.

4.6.2 Settings

Time Column: In order to display the film strip correctly, a video time in the data must be present, which is used for synching the video to the displayed data. Select this column here. If the time unit for the Video time is not set, it is assumed to be given in seconds. If you have problems displaying your video, set the time unit for the video time column in the properties section of the data node where the data originates.

Video Time Factor: A factor that can be used to adapt the video time before it is used. Normally, there should be no need to set this value to anything other than 1, if the time unit is set correctly for the video time. It can be used to adjust the video time, if you only have a time column that does not properly sync to the video time.

Video Time Offset: This value can be used to offset the start of the video regarding the start of the data. This value is assumed to be in the same time unit as the Time Column. If the unit of the Time Column is unknown, or if the Time Column is a TimeStamp or DateTime value, the Offset is assumed to be given in seconds.

Media Selection: Determines, how the video, that is being displayed, is selected. There are two different options:

Project: The video is selected according to the stimulus association you can set in the Project Manager.

Manual: You can select the video via the Manual Media Association.

Note that regardless of the setting, all media need to be have been added in the Project Manager, even if they are used via the "Column" or "Manual" option and thus no association is set in the Project Manager.

Manual Media Association: Associates media with scenario/participant combinations, if Media Selection is "Manual". This setting can be used, if you have several media files per scenario/participant combination and want to display another one than is associated in the Project Manager.

4.7 Stimulus Visualization

The stimulus visualization displays a simple still image. It is mostly used as a backdrop for scan path and heat map visualizations. The stimulus visualization can display either simple images, or a single frame of a video, if the selected data has a video as an associated stimulus. The stimulus visualization requires that a video stimulus is associated to the data (See Project Manager).

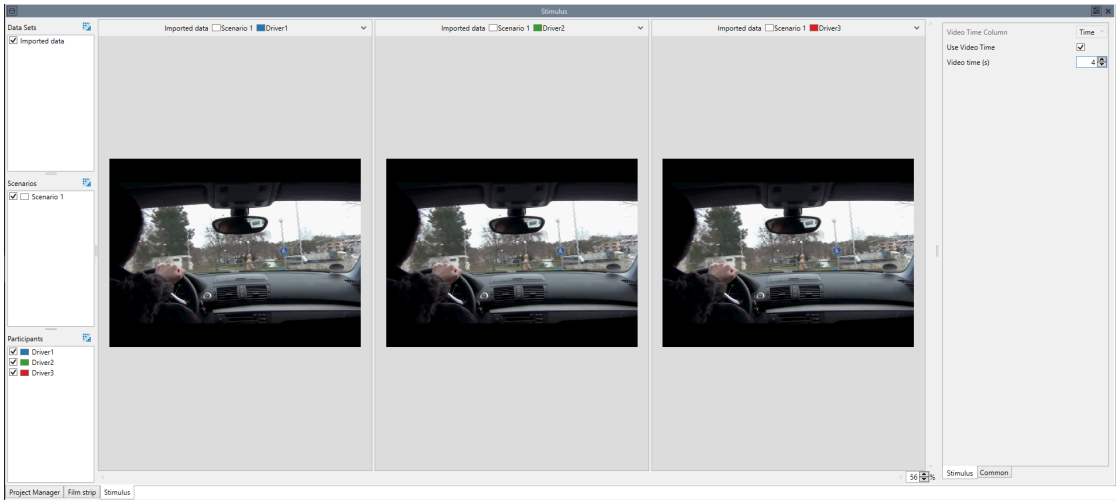

Figure 4.6 — A Stimulus Visualization

4.7.1 Aggregation

The stimulus visualization is able to aggregate, but only if the selected data sets / scenarios / participants are associated with the same stimulus. In that case the stimulus is drawn only once.

4.7.2 Settings

Video Time Source: Determines, how the frame of the video that is being displayed is determined. There are three different options:

Manual: Allows you to manually set a time via the Video Time setting.

Start: The first frame of the displayed data area is used. Note, that this is not necessarily the first frame of the video. If you are using a marking as an input, the value in the Video Time Column at the start of that marking is used.

Cursor: The value in the Video Time Column is used, but at the current cursor position. Note that this setting is only relevant, if the stimulus visualization is displayed in the same window as a time-based visualization. If no such visualization exists in the same window, the "Cursors" setting has the same effect as the "Start" setting.

Video Time Column: Determines the origin of the time of the video frame that is displayed if the displayed stimulus is a video and the Video Time Source is either "Start" or "Cursor".

Video Time Factor: A factor that can be used to adapt the video time before it is used. Normally, there should be no need to set this value to anything other than 1, if the time unit is set correctly for the video time. It can be used to adjust the video time, if you only have a time column that does not properly sync to the video time.

Video Time Offset: This value can be used to offset the start of the video regarding the start of the data. This value is assumed to be in the same time unit as the Time Column. If the unit of the Time Column is unknown, or if the Time Column is a TimeStamp or DateTime value, the Offset is assumed to be given in seconds.

Video time (s): If "Use Video Time" is true, use this specific time in the video for the displayed frame.

Media Selection: Determines, how the media, that are being displayed, are selected. There are three different options:

Project: The media are selected according to the stimulus association you can set in the Project Manager.

Column: The names of the media files is read from the selected Stimulus Column. This is useful, if you have an experiment where several stimuli are associated with the same participants and cannot meaningfully be split into different scenarios. If possible, it is normally preferable to use different stimuli as different scenarios. Note, that in order to select a stimulus from a column, you need to either have a visualization with a cursor in the same window as the stimulus visualization, or you need to have grouped the data according to the stimulus column with a Group Node.

Manual: You can select the media via the Manual Media Association.

Note that regardless of the setting, all media need to be have been added in the Project Manager, even if they are used via the "Column" or "Manual" option and thus no association is set in the Project Manager.

Stimulus Column: Selects the column from which a stimulus name is read if "Column" is selected as the Stimulus Selection. Note that for this to work, the stimuli need to have been loaded in the Project Manager, although an association with a Scenario or Participant is not necessary.

Manual Media Association: Associates media with scenario/participant combinations, if Stimulus Selection is "Manual". This setting can be used, if you have several media files per scenario/participant combination and want to display another one than is associated in the Project Manager.

4.8 Diagram

The diagram visualization displays a bar chart visualization. It accepts one or more data tables and maps column values to rectangles with a length proportional to the value. This gives a quick overview over the relation of a limited set of values.

Each bar in the diagram corresponds to one of its categories. If, for example, you set up an analysis that computes the average of values in a data column named "x" and the average of values in data column named "y", a diagram visualizing these results would have two categories, "x" and "y". The diagram would contain two bars, directly relating the average values of both columns.

A diagram may have multiple data series. A data series relates to a group of bars, so with multiple data series, there will be one bar group per series, each group containing one bar per category. If, for example, you set up an analysis to compute both the average and the standard deviation of two data columns "x" and "y", a diagram visualizing these results could have the two categories "x" and "y" and the two data series "Average" and "Standard deviation". The diagram would contain two groups of bars ("Average" and "Standard deviation"), each showing two bars ("x" and "y"). This enables you to compare different quantities within a single diagram.

The diagram node maps the rows of its input data to categories and the columns to data series.

When hovering the mouse cursor over a bar in the diagram, the tooltip will identify the bar (the corresponding data column and data set/scenario/participant information as appropriate) and its value.

4.8.1 Aggregation

The diagram visualization supports aggregation for data sets, scenarios, and participants. Aggregation combines the individual diagrams in the correspon-

ding dimension by adding each combination of the aggregated dimension and the selected data columns as a diagram category. When simultaneously aggregating sets, scenarios, and participants, the result will be a single diagram with (number of data sets) times (number of scenarios) times (number of participants) times (number of selected data columns) categories.

4.8.2 Settings

Data Columns: The data columns of the input data to be used as diagram series. The selected data columns should contain numerical data or no bars will be shown.

First Column Is Label: If enabled, the first selected column will not be used as a diagram series. Instead, the value in this column will be used to identify the corresponding bar in the tooltip and the diagram legend. In this case, any column type is acceptable for the first column. This option is only available if more than one column is selected.

Bar Colors: The colors to be used for the diagram bars. Categories (i.e. bars in bar groups) will be colored using the colors specified here. If there are more bars than colors, bars will cycle through the colors.

Use Project Colors : If this option is enabled and a bar is specific to a scenario and/or participant, the brightness or color associated with the scenario/participant in the project view will be used instead of a color from the "Bar Colors" setting.

Value Axis: Sets the range and interval spacing for the value axis of the diagram. By default, each diagram will scale its value axis based on the value range encountered in its data. In order to force multiple diagrams to the same value range, supply explicit values for the "Minimum" and "Maximum" settings.

Minimum: The minimum value of the value axis or "Auto" to auto-detect an appropriate value

Maximum: The maximum value of the value axis or "Auto" to auto-detect an appropriate value

Interval: The interval of horizontal segmentation lines along the value axis or "Auto" to auto-detect an appropriate value

Options: Various options for the diagram

Use Color Gradients: If disabled, the bars will be rendered as simple, flat-shaded rectangles.

Show Title: Shows a title above the diagram. The title includes the name of the diagram node and the names of the data set, scenario, and participants (unless the corresponding aggregation option is enabled, in which case a single diagram includes the data of all data sets/scenarios/participants).

Show Legend: Shows a legend identifying the categories below the diagram. This is a three-state value. If it is indetermined, the legend will be displayed if there is more than one category in the diagram.

4.9 Scan Path

The scan path node is used to display fixations and saccades. Normally, the scan path is combined with a stimulus visualization, so that it becomes possible to see where on the stimulus the fixations of the participants are located. The stimulus visualization is always created automatically, and coupled with the scan path. If you want a scan path without a stimulus visualization, simply delete the stimulus visualization node that gets created automatically.

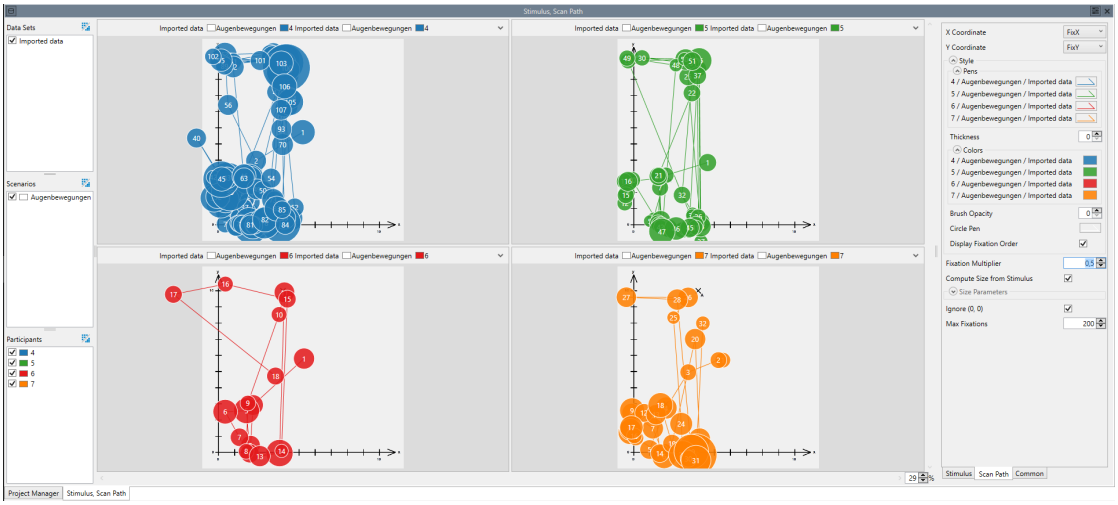

Figure 4.7 — A Scan Path Visualization (not aggregated)

4.9.1 Aggregation

The scan path visualization is able to aggregate by drawing the scan paths of different participants / scenarios onto the same stimulus. Note that aggregation

for the scan path is only enabled, when the underlying stimulus visualization is able to aggregate. Figure 4.8 shows a scan path with the same data as shown in figure 4.7, but with aggregated participants.

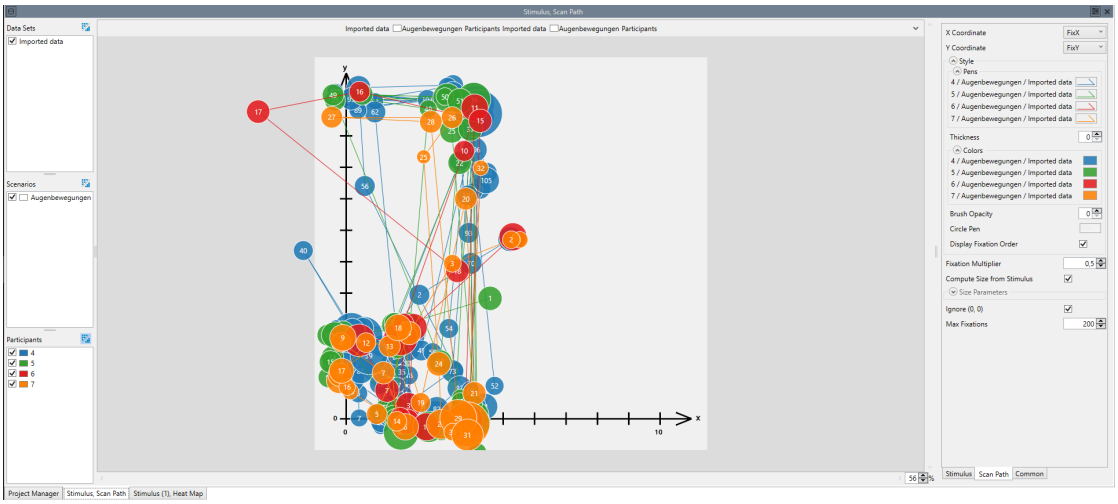

Figure 4.8 — A Scan Path Visualization (aggregated)

4.9.2 Settings

X Coordinate: The column containing the x coordinate of the fixations.

Y Coordinate: The column containing the y coordinate of the fixations.

Radius Computation: Determines, how the size of the circles for each fixation is computed. The default ("Fixation Length") results in the traditional style of using the fixation length for the circle size. The other options allow selecting an independent column. The mean, minimum or maximum value of this column during a fixation is used to determine the circle size. Note that the used value is not used as the radius, but as the area of the drawn circle.

Fixation Length Source: Determines, how the length of a fixation is computed. This parameter has three options. "Time Column" will compute the fixation length from the time column, and should work in most cases. "Duration Column" can be chosen, if your data has a column that contains the duration of each fixation, but no time column. "Row Count" will use the number of data rows as a substitute for the duration of a fixation and can be used, if your data has a fixed sampling rate, but neither a time nor a duration column.

Time Column: The time column. This parameter is only used, if Fixation Length Source is "Time Column".

Duration Column: The fixation duration column. This parameter is only used if Fixation Length Source is "Duration Column".

Radius Source Column: The column that influences the area of the circles, if Radius Computation is "Selectable Column Mean", "Selectable Column Min" or "Selectable Column Max".

Fixation Captions: Determines the caption that is displayed inside of each fixation. The default ("Fixation Order") results in the traditional style of assigning a number to each fixation in ascending order. "None" will not display any caption. The other options allow selecting an independent column. The mean, minimum or maximum value of this column during a fixation is displayed as a caption of that fixation.

Fixation Captions Column: The column that is used for computing the Fixation Captions, if that is set to "Selectable Column Mean", "Selectable Column Min" or "Selectable Column Max".

Couple to Cursor: This option has no effect as long as the scan path is displayed as the only visualization in its window. When it is displayed in the same window as a visualization with a cursor, this option, if enabled, couples the display of the scan path to that cursor. In that case only a certain number of fixations before and/or after the current cursor position is displayed and the visualization interactively adapts to the current cursor position.

Before Cursor: Display this many fixation before the current cursor position, if Couple to Cursor is active.

After Cursor: Display this many fixation after the current cursor position, if Couple to Cursor is active.

Style: Parameters that determine the look of the visualization:

Pens: The styles that are used to draw the saccades.

Thickness: The thickness of the lines representing the saccades. If the value is 0, the thickness of each line can be set individually in the pen dialog of each line. Otherwise this value overrides the individual values and can be used to quickly change all lines.

Colors: The colors that are used to fill the fixations.

Brush Opacity: The opacity of the circles representing the fixations. If the value is 0, the opacity can be set individually in the dialog of each color.

Circle Pen: The Pen that is used to draw the circles around the fixations.

Fixations Ignore Zoom: If enabled, fixations do not scale accordingly, when zoomin in or out, but instead keep their fixed size.

Display Fixation Order: If enabled, display ascending numbers indicating the order of fixations.

Reference Radius: This value can be used to adjust the size of the circles representing fixations. It represents the size of a fixation of 300ms (or 20 data rows) and needs to be adjusted if your time or duration column is not in milliseconds (or depending on your sample rate).

Size Parameters: Parameters that determine the size of the scan path:

Clamp to Stimulus: If enabled, the size of the scan path depends on the size of the stimulus and fixations outside of the visible area are not shown. If disabled, the scan path is as large as necessary to display all fixations.

Margin: A margin on which fixations are displayed outside of the stimulus area, if Clamp to Stimulus is enabled. The margin is given as a percentage of the stimulus size.

Offset (X): The Visualization gets moved in X direction by this offset.

Offset (Y): The Visualization gets moved in Y direction by this offset.

Scale (X): Scale the entire scan path by this factor in x direction.

Scale (Y): Scale the entire scan path by this factor in y direction.

Ignore (0, 0): If true, ignores all (0, 0) coordinates when computing the scanpath. Enable this, if the data defaults to (0, 0) on undetected eye direction.

Max Fixations: Display only the first n fixations. If more fixations are in the data, later fixations are truncated.

4.10 Heat Map

The heat map visualizes the distribution of fixations, where the intensity and density of fixations are coded into colors.

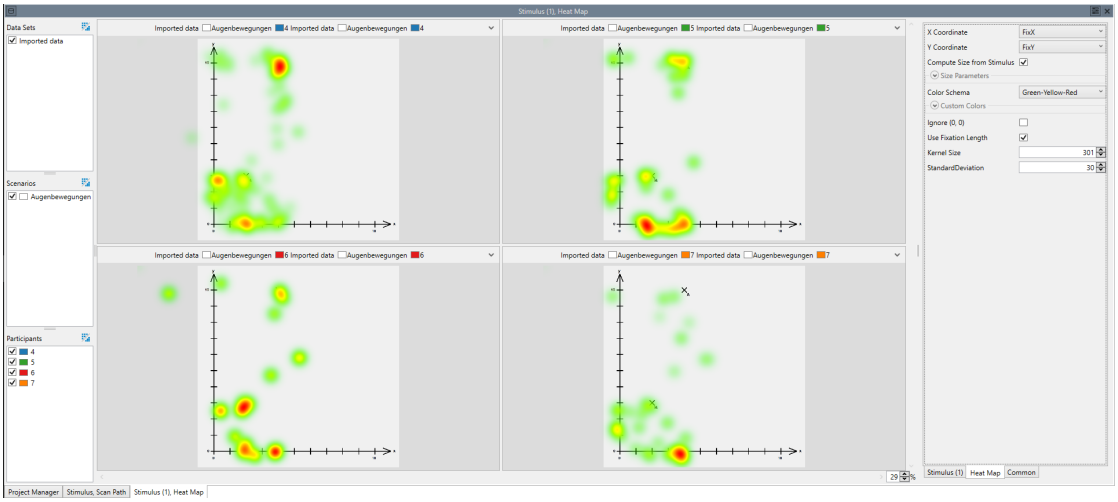

Figure 4.9 — A Heat Map Visualization (not aggregated)

4.10.1 Aggregation

The heat map visualization is able to aggregate. When aggregating, the displayed heat map shows a heat map over all selected participants and/or scenarios / data sets. Note that aggregation for the scan path is only enabled, when the underlying stimulus visualization is able to aggregate. Figure 4.10 shows a heat map of the same data as shown in figure 4.9, but with aggregated participants.

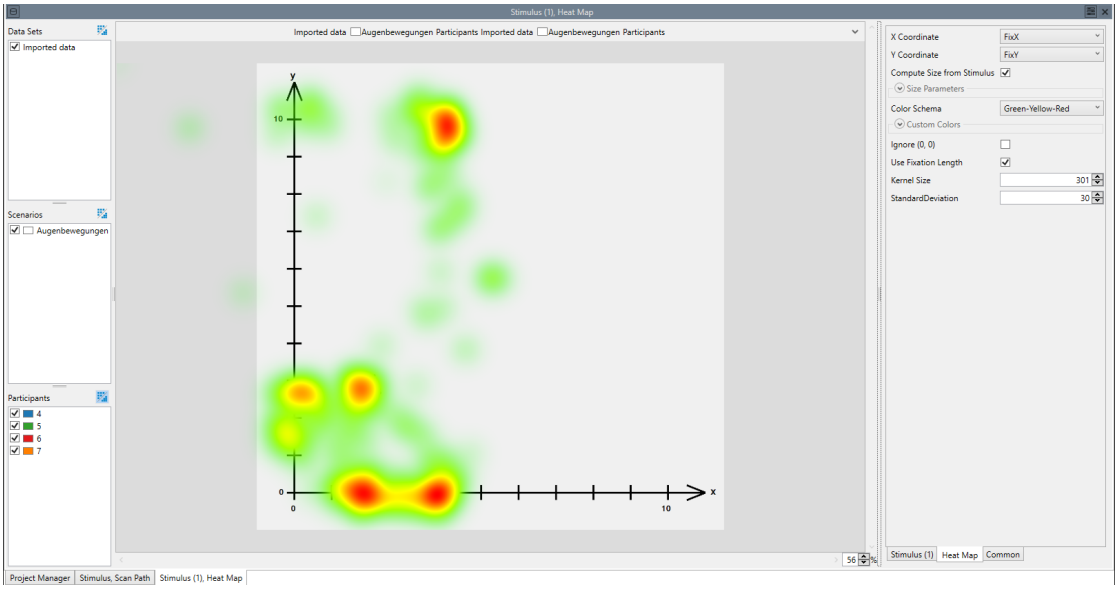

Figure 4.10 — A Heat Map Visualization (aggregated)

4.10.2 Settings

X Coordinate: The column containing the x coordinate of the fixations from which the heat map is computed.

Y Coordinate: The column containing the y coordinate of the fixations from which the heat map is computed.

Size Parameters: Parameters that determine the size of the heat map:

Clamp to Stimulus: If enabled, the size of the heat map depends on the size of the stimulus and coordinates outside of the visible area are ignored. If disabled, the heat maps is as large as necessary to display all coordinates in the data.

Margin: A margin on which the heat map can be displayed outside of the stimulus, if Clamp to Stimulus is enabled. The margin is given as a percentage of the stimulus size.

Scale (X): Multiply the values found in the "X Coordinate" column by this value.

Scale (Y): Multiply the values found in the "Y Coordinate" column by this value.

Offset (X): Offset the result of the scaled X coordinate by this value.

Offset (Y): Offset the result of the scaled Y coordinate by this value.

As an example, assume the coordinates of your fixations are between -1 and 1, indicating a normalized coordinate system centered on the stimulus, and your stimulus has a size of 800 x 600. To convert between those coordinate systems, the formula Coordinate * Scale + Offset is applied. So, in the example, set "Scale (X)" to 400 and "Scale (Y)" to 300. The resulting coordinates will thus be between -400 and 400 (-300 and 300). Set Offset (X) to 400 and Offset (Y) to 300, to move the coordinates into a system that fits the heatmap.

Compute Scale : This factor can be used to downscale the heatmap for the computation, if the computation of the heatmap is too slow. A smaller heatmap (scaled by the given factor from the original size) is used, and the result is upscaled to the original size for displaying. Accordingly, the smaller the Compute Scale, the faster the heatmap is computed, but the less accurate it is.

Color Schema: The Color Schema that is used for this Heatmap.

Custom Colors: The Colors for the "Custom" Color Schema. The drawn heatmap interpolates between all colors selected here.

Ignore (0, 0): If true, ignores all (0, 0) coordinates when computing the heatmap. Use this, if the data defaults to (0, 0) on undetected eye direction.

Use Fixation Length: When basing the heatmap upon fixations, use the length of a fixation as a weight.

Kernel Size: The size of the Gaussian kernel used to create the heatmap.

StandardDeviation: The standard deviation of the Gaussian kernel used to create the heatmap.