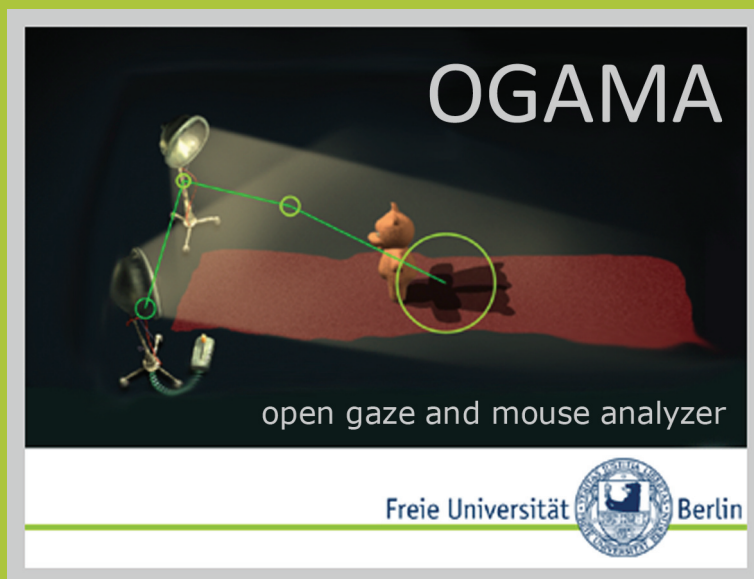


## OGAMA Description (for Version 2.5)

A software to record, analyze and visualize gaze and mouse movements in screen based environments



<http://didaktik.physik.fu-berlin.de/ogama>

Adrian Voßkühler

Freie Universität Berlin  
Fachbereich Physik  
Didaktik der Physik  
Arnimallee 14  
14195 Berlin  
Germany

[adrian.vosskuehler@fu-berlin.de](mailto:adrian.vosskuehler@fu-berlin.de)

<http://didaktik.physik.fu-berlin.de>

## Introduction<sup>1</sup>

In recent years, eye tracking research benefited from the fact that hardware and applications development started to complement one another so that currently the hardware is stable and tailored to a wide range of applications. But still, the software for analyzing the data 'is usually proprietary, nonextensible (...) and nonmodifiable in the sense that the user cannot add additional analyses or modify the algorithms' (Gitelman 2002, S. 605).

Inspired by this, we decided to start an open source project for developing a gaze-analyzer. We think that the 'open source' idea is the best way to combine the power of different researchers in one tool that provides them with the opportunity to fulfil their own needs. Such a project should combine common known 'state of the art' gaze-analyzing tools such as attention maps, areas of interest definition and real-time replay along with an easy-to-use graphical user module. Because of the huge amounts of data that are recorded during eye-tracking experiments it should rely on visual representations of the results so that the human image processing can identify the crucial aspects to concentrate on in further processing and the statistical analysis. From our point of view graphical representations and visualisations of gaze-data are a key to eye movement research, especially for the facilitation of data mining processes (Tan et al. 2005). Therefore such a tool should be designed in a powerful programming language that handles large databases, supplies flexible graphical modules and provides the researchers with a basis for customisation, e.g. to define new saccade-related variables.

At the same time, coming from our own applied research in the field of physic didactics it would be nice to have a tool that can handle mouse-tracking data as well. From usability research during web browsing it is known that there is a strong relationship between gaze position and mouse position (Chen et al. 2001). In physic didactics we investigate students interacting with direct manipulative media (Shneiderman & Plaisant 2004) in screen based environments. Does the close relationship between gaze and mouse movements hold true for interactive screen experiments (ISE) used in learning situations (Kirstein & Nordmeier 2007) or do gaze and mouse patterns differ depending on previous knowledge? To improve the conclusions drawn from tracking

---

<sup>1</sup> This paper is a revised version of the following publication including the changes up to version 2.5:

Voßkühler, A., Nordmeier, V., Kuchinke, L., & Jacobs, A.M. (2008). OGAMA - OpenGazeAndMouseAnalyzer: Open source software designed to analyze eye and mouse movements in slideshow study designs. *Behavior Research Methods*, 40(4), 1150-1162.

ISE we required a tool that merges eye-tracking data with mouse movement data.

Here we describe a software application that is intended for recording and analyzing gaze and mouse data in parallel. We called this tool 'OGAMA' - the OpenGazeAndMouseAnalyzer. OGAMA is suitable for experimental setups with screen based slide show stimuli.

The main goals in designing OGAMA were the following:

- [1] recording and analyzing of gaze and mouse movement data together
- [2] open source publishing to provide extensibility to the research community
- [3] implementation of state-of-the-art eye tracking analysis functions like replay, areas of interest (AOI) definition, attention maps in an up-to-date user module
- [4] visual-guided data mining through the use of different visual representations
- [5] database-driven software layout to handle highly sampled eye- and mouse tracking data

The article is split in two main sections. In the Application Design section we describe in detail basic principles of the underlying application structure, the database and the fixation detection algorithm. The other section consists of a brief description of the currently implemented application modules to give an overview of the processing steps to follow during the analysis from the data import to the computation of the statistical output variables.

## 1. Application Design

To have full freedom in developing algorithms and their fast implementation, and on the other side to have fast and powerful tools for the graphical output and the user-module we decided to use the Microsoft® Visual Studio® development suite with the language C#™ .NET. This is a low level programming language that enables real time support and fast data access due to its performance.

### 1.1. Storage Files

OGAMA creates a folder 'OgamaExperiments' in the 'MyDocuments' folder of the current user. Each experiment there resides in its own folder with at least three subdirectories. The .oga file, which has an XML layout, stores the key values for the description of each experiment. These are the database files, the path to stimuli, sampling rates, eye tracker screen size, the fixation calculation parameters and a complete description of the slideshow of the experiment. This file can be used to start the

OGAMA session by double clicking on it. The 'Database' subfolder contains the two MSSQL database files. They are pure MSSQL standard, so you can access the database without OGAMA with any MSSQL enabled language or application (e.g. Microsoft<sup>®</sup> Access, SQLExpress). All stimuli resources that are used in the slideshow are copied to the 'SlideResources' subfolder. Recording videos of dynamic slideshow content and user camera videos are saved into the 'Thumbs' subfolder.

## 1.2. Slideshow concept

Each experiment contains a slideshow with the displayed slides during the recording, but the smallest analysis unit is a trial. It is the default display and analysis unit, so e.g. statistical output is retrieved trial by trial. By default each trial consist of only one single slide, but one can concatenate slides within one trial, to separate subsections of a trial. Consider a story of 5 pictures that should be replayed continuously one after another during analysis, but have moving areas of interest. In that case one would create 5 slides each with the associated picture and concatenate them to one trial. During analysis in the different modules one can select the trial and then switch on a timeline between the different slides. The statistical output will be aggregated over the 5 slides contained in the trial.

## 1.3. Database Layout

OGAMA stores the gaze and mouse tracking data in SQL Express local data files that are created for each experiment. The underlying database is shown in Figure 1. The whole database is subdivided into nine tables. The subjects table contains the list of subjects with their associated descriptive attributes like category, age, sex, handedness and comments. These predefined parameters can be additionally extended with custom variables that are stored into the subject parameters table containing subject name, parameter key and value.

The trials table contains the meta-information of all trials. Information of each trial of the experiment for each subject is stored with trial-associated variables like id, name, sequence, start time, trial duration and category along with a column indicating the validity of the recorded data. For each subject a raw data table is created which contains the imported gaze- and mouse tracking data and the corresponding timing information. For each time stamp this can include up to two columns for pupil diameter data (PupilDiaX, PupilDiaY), two columns for gaze position data (GazePosX, GazePosY), two columns for mouse position data (MousePosX, MousePosY) and a column associated with an occurred trial event. All the trial events that occurred during recording are

stored into the trial events table containing the events time, type, task and parameter. OGAMA records mouse, key, audio, video, usercam, marker and user response events; so that the trial time line is separated and interesting points in time can be marked during recording.

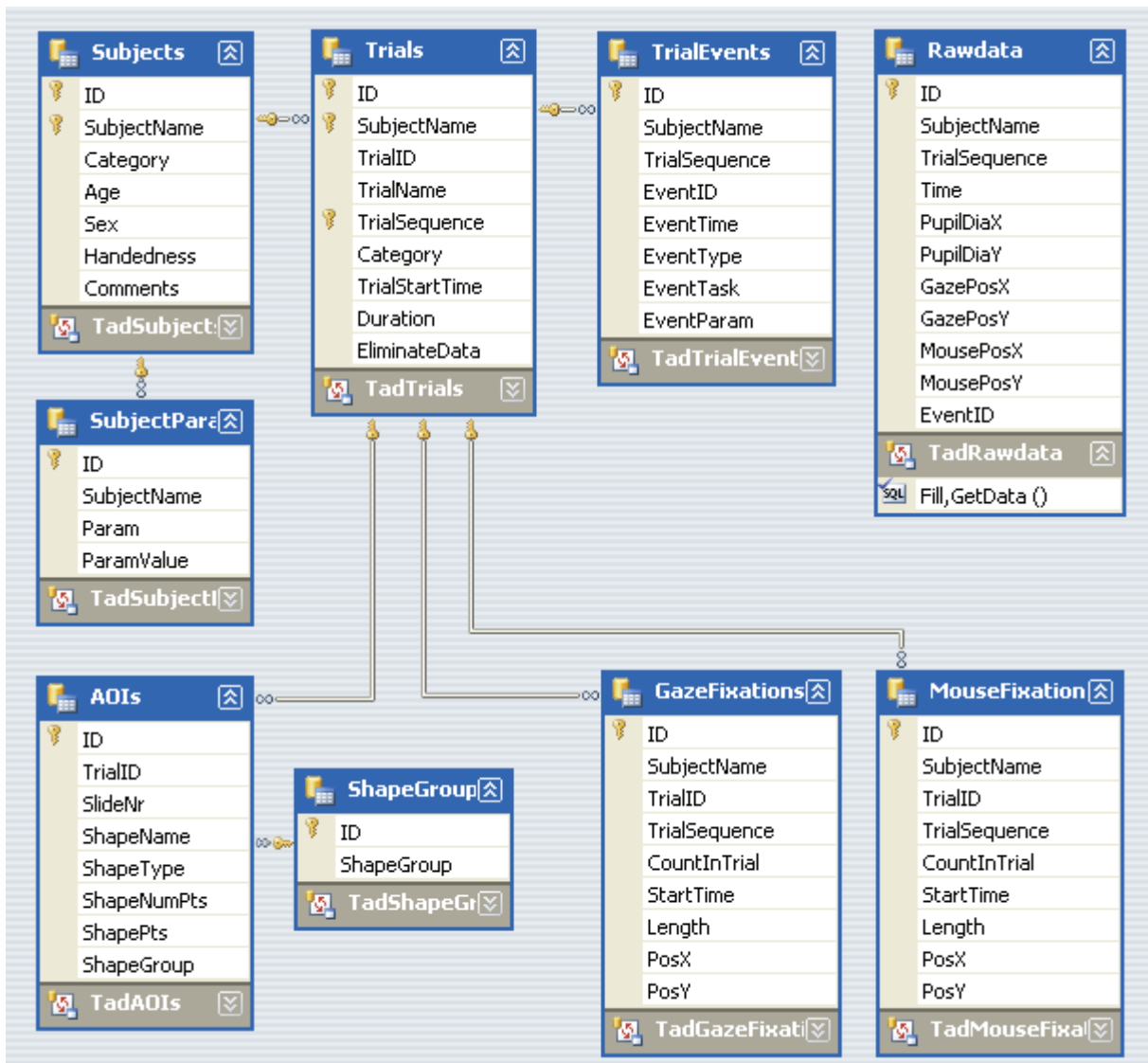


Figure 1: Database design of OGAMAs SQL database template file.

If areas of interest (AOI) are defined during the analysis they are stored in the AOI table. AOI are defined to additionally analyse sub-regions in a stimulus image. For example, AOI information can be used to calculate fixation durations on a special feature/object of a slide. OGAMA provides the possibility of creating rectangular, ellipsoid and polygonal shaped AOI (see Section 2.3). Each shape is store together with a name (ShapeName), the owning trial and slide (TrialID and SlideNr), its type (ShapeType), its location and size (ShapePts) and a column for AOI categorization (ShapeGroup). The used shape groups of the experiment are stored into the shape group table which contains by default a 'target' and a 'searchrect' entry. This allows groups of areas of interest to be visually separated from each other and eases the analysis process.

Finally, two fixation tables list the gaze and mouse fixation information for each subject and trial with its associated variables as there are the number of the fixation in a trial (CountInTrial), the starting time (StartTime), fixation duration (Length) and the position (PosX, PosY).

All nine tables reside in the experiments' locally stored SQL database file. Data integration is ensured over foreign key constraints between these tables as shown in Figure 1.

#### 1.4. Fixation detection

The fixation calculation is done using the fixation detection algorithm published from LC technologies. It is a dispersion-type algorithm with a moving window (Salvucci & Goldberg 2000). We ported the source code to C#™ and added a time estimation support but the working principle has not been changed. This is described in the source code documentation: “The function detects fixations by looking for sequences of gaze- point measurements that remain relatively constant. If a new gaze point lies within a circular region around the running average of an on-going fixation, the fixation is extended to include the new gaze point. (The radius of the acceptance circle is the above user specified maximum distance.) To accommodate noisy measurements, a gaze point that exceeds the deviation threshold is included in an on-going fixation if the subsequent gaze point returns to a position within the threshold. If a gaze point is not found, during a blink for example, a fixation is extended if

- a) The next legitimate gaze point measurement falls within the acceptance circle, and
- b) There are less than the above minimum numbers of samples of successive missed gaze points. Otherwise, the previous fixation is considered to end at the last good gaze point measurement.” (LC Technologies 2006)

Prior to the fixation calculation the fixation detection algorithm applies a two step filter. In the first step empty values are ignored in the calculation. This is necessary due to the different sampling rates of the gaze and mouse data. The second step is to omit samples with both x- and y coordinates equal to zero which often marks the eye tracker output during a blink, and to omit samples that lie out of the screen.

In the experimental settings one has to specify two parameters for this algorithm. First, the maximum distance (in pixels) that a point may vary from the average fixation point that is still considered to be a part of the fixation. Second, the minimum number of samples that defines a fixation. Depending on the experimental sampling rate and the research domain

these highly interdependent parameters have to be chosen carefully (see Karsh & Breitenbach 1983, for a discussion).

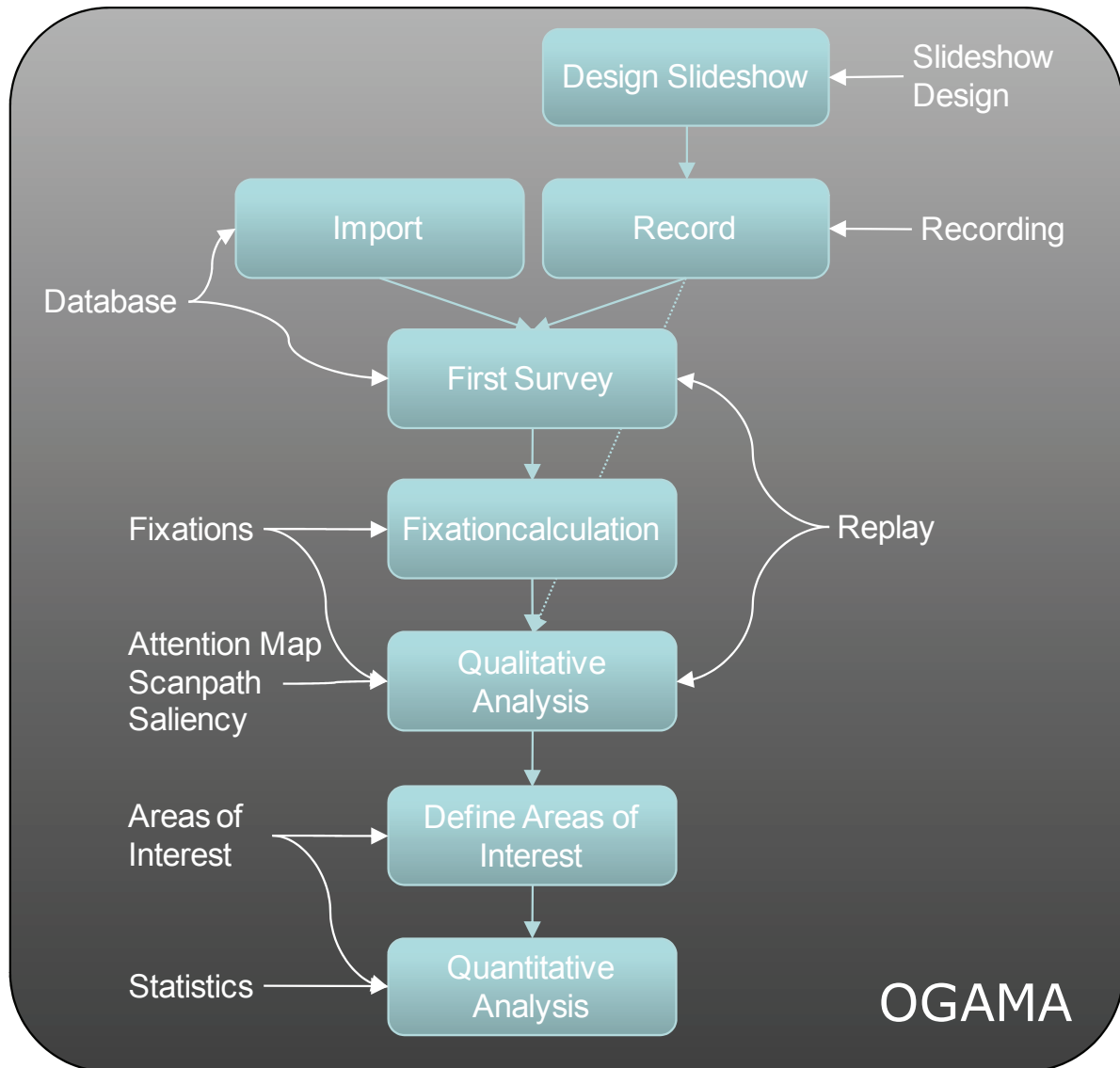
## 2. Modules

This section contains a brief description of OGAMAs capabilities. When using OGAMA we suggest that the user follows the subsequently described processing steps. OGAMAs workflow is depicted in Figure 2. A new project is started by either designing a slideshow presentation (see section 2.1) and recording new data (see section 2.1.2), or by importing new sampling data with OGAMAs import assistant (see section 2.1.3). Yet, recording or importing succeeded, each step of the analyzing process is supported by one of eight different modules. Each module is specialized on its intended purpose. The modules can be grouped into five sets for visual data analysis which are Replay (see 2.1.4), Fixations (2.2.2), Scanpath (0), Saliency (0) and Attention Maps (2.2.5), two modules for statistical data analysis which are Areas of Interest (2.3) and Statistics (0) and the Database module (2.1.4) which is used for controlling the underlying data.

The modules reside in a main window which also hosts the menu, a tool strip, a status strip and a context panel. The menu provides the basic functionality for the application like opening, saving, closing experiments, copy and paste methods for table and module content, activation and creation of the different modules, modification of experiment settings, application options and database connection and a help section. Tool strip enables control over the activation of the modules and status strip contains current workflow or progress information.

The context panel contains lists of subjects and the slideshow with the trials to easily select a trial and/or a subject for analysis in the current active module. It also shows the current selected subjects user camera video if available along with a context help for the active module.

Each of the following sections describes the module in detail and depicts an exemplary screen shot of the modules created with the OGAMA demo project (available from the OGAMA website).



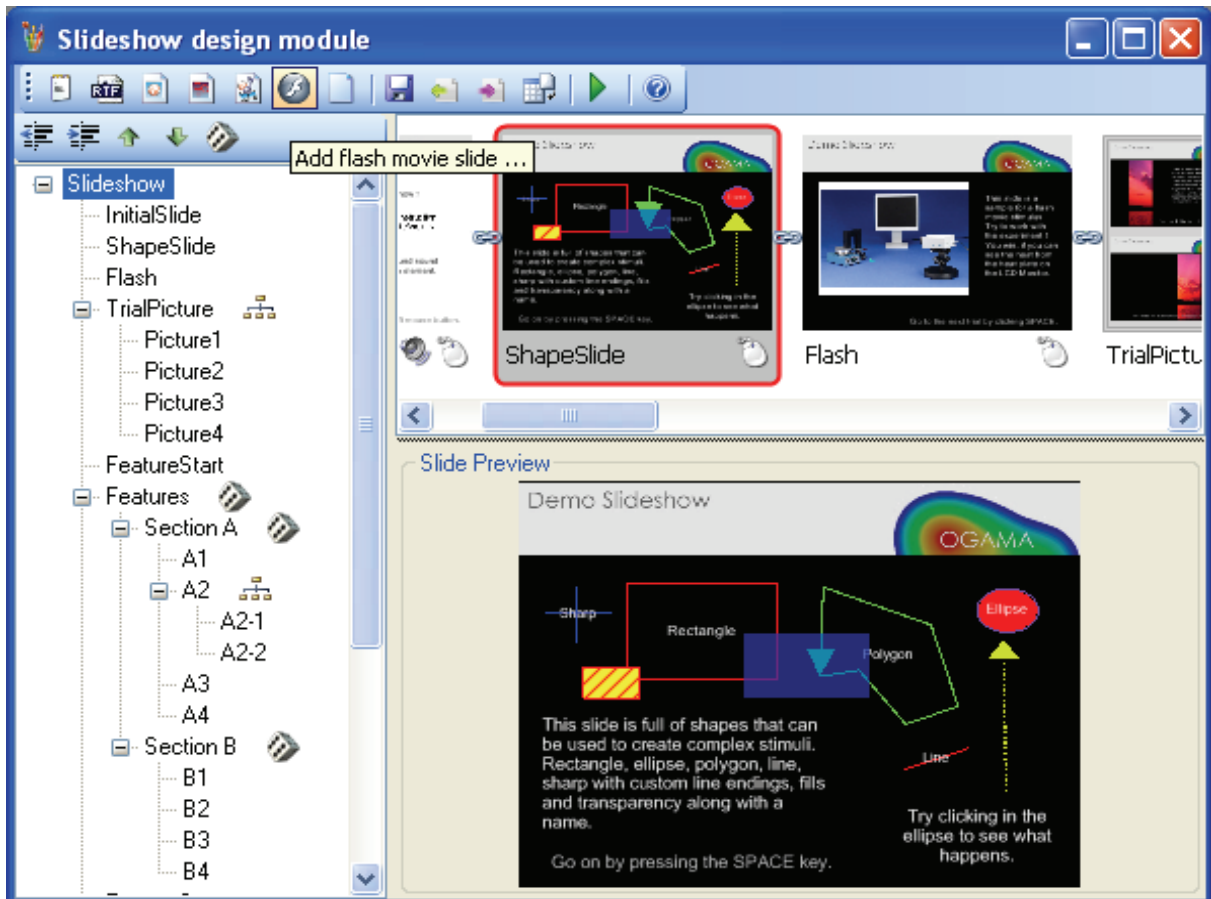
**Figure 2: OGAMA's workflow with the supporting modules**

## 2.1. Design and Record

### 2.1.1. Slideshow Design

The Slideshow Design module is designed to enable the creation of slides to be presented during an experiment. Six different stimulus types are supported, which are textual stimuli, image stimuli, shape stimuli, blank stimuli, auditory stimuli and flash movies. The newly created slide sequence is depicted in a tree view and a timeline. Selecting a slide displays its contents in the slide preview window (see Figure 3 for a screenshot).





**Figure 3: Screenshot of the Slideshow Design module with tree view, timeline and preview (Windows XP style)**

Create a slide by clicking on of the stimulus type buttons or select the folder import mode. This is to facilitate the import of a lot of stimuli files e.g. pictures contained in a folder. This dialog provides the options to specify general slide properties for all imported slides and then creates a slide with this general options for each of the stimulus files in the folder. They are inserted into the slideshow and can be afterwards modified via custom dialogs.

When slides has been created they can be arranged and grouped in a hierarchical order that is visualized in the tree view at the left side of the module. This is helpful to achieve shuffling of specific subslices or just to group slides to be clearly laid out (see Figure 3 for a screenshot). The connecting icon between the different slides in the timeline indicate shuffled or joined slides. The context menu of the tree view provides more sophisticated options on arranging and shuffling, so it is e.g. possible during presentation to display only 3 randomly selected slides out of 20 in the section for the specific subject.

To modify a slide double-click on it in the timeline to open a slide properties dialog consisting of the following areas (see Figure 4 for a screenshot)

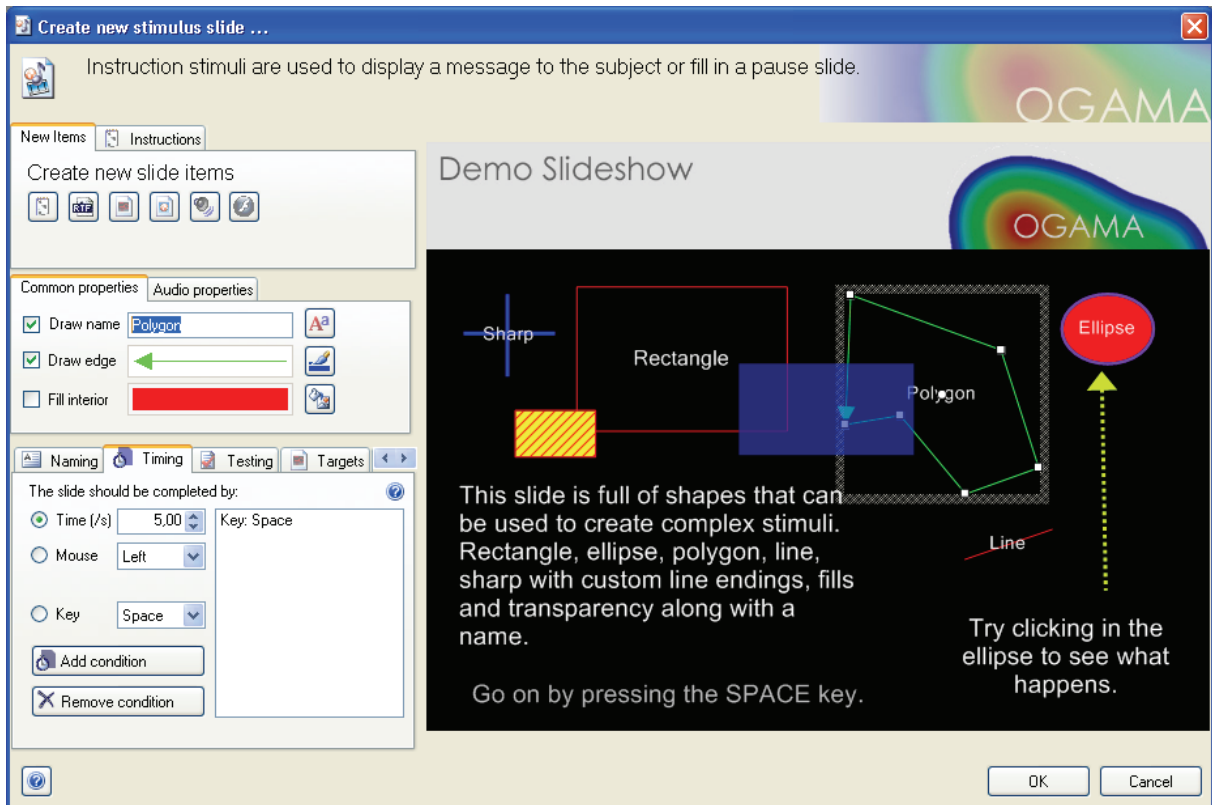


Figure 4: Screenshot of the slide properties dialog. (Windows XP style)

### 2.1.1.1. Stimulus creation and modification

This section covers the properties needed to add a specific stimulus to the slide. For example adding instructions to a slide is done by pressing the “Add instruction” button which opens a dialog for writing the instruction and choosing its style properties like font, size and color. The position of the instruction on the slide is then defined with the mouse by dragging the bounding rectangle on the slide. Otherwise clicking on an existing element switches this area to some quick change fields, double-clicking on the element shows the original creation dialog.

*Instructional stimuli* can be defined in two ways using default plain text with one font, alignment and color, rendering nicely, or in a rich text enabled version with all rtf available design options, e.g. custom style for each letter, which renders a bit crispy. The layout and word wrap of the text can be modified after insertion by dragging the grab handles of the stimulus bounding rectangle.

*Image stimuli* can be imported from bmp, jpg, gif, png, ico and wmf file formats. Image layout options are ‘Center’, ‘None’ where the image position and size is defined via dragging the bounding rectangle, ‘Stretch’ where the image is stretched across the slide area, ‘Tile’, or ‘Zoom’ where

the image is proportionally enlarged to fit within the slide area. Although images by default fill the area they are bound to one can optionally define a fill style for this element to enable e.g. overlay effects.

*Shape stimuli* can be rectangles, ellipses, polylines, lines and sharps. Each element can be drawn just like all other elements with a name an edge and a fill and supplemented with a music file. All shapes are created by dragging the bounding rectangle except the polyline which is to define point by point. When reaching the first point the mouse cursor changes to indicate the optional closing of the polyline. Note that open polylines with fill seem to be closed, because the fill closes the bounds to generate the fill area.

*Sound stimuli* consist of a sound file to be played on click or on appearance, but are by default hidden from the user, the click zone can be specified by modifying the bounds.

*Flash movie* stimuli are namely .swf Files, that are started to play on initial slide display during presentation. Because they reside in an underlying ActiveX control they can be controlled also during design mode. To enable all changes on the flash player to be available during analysis a screen movie is captured during recording. For large or lengthy movies be sure to have an effective video compressor installed that can be used in the recording module. Screen capturing is a CPU demanding process, so in this case a dual core processor is advised. Note that only one flash movie is allowed per slide.

The second part of the stimulus creation and modification area contains two tabs for common properties of all elements like naming, edge, fill and audio properties. Once an element is selected one can switch the visibility of a descriptive name, the bounding edge or the brush used to fill the interior of this element along with options to play a sound on appearance or clicking into its bounds. The font, pen and fill styles can be separately defined via dialogs similar to the dialogs known from common graphical applications. Note the very extensive options like texture and hatch brushes, transparent fills and sizable dash and dotted lines with line caps.

### 2.1.1.2. Slide options

The left bottom of the slide properties dialog contains a bunch of tabs to modify different slide wide properties described below.

*Naming:* The Name and category of the slide should be entered here whereby the name should be unique but doesn't have to, OGAMA internally allocates unique slide ids. The category is an optional grouping possibility at slide level, that can be used in the statistics output.

*Timing:* This tab is used to define the conditions under which the slide should be exchanged. That can be after a specific amount of Time (Time stop condition), after a specific or arbitrary key response (Key stop condition) or after a specific or arbitrary mouse click (Mouse stop condition). Note that multiple stop conditions can be defined for one slide, so it is e.g. possible to exchange the slide either after a users left mouse click or a time limit of 30 seconds. Two keys have special bindings, the ESC key always ends the presentation and the F12 key sets a time marker, so they both should not be used as a valid user response, they are reserved for the experimenter.

*Testing:* Within this tab targets or responses can be defined (e.g., a target shape to click to or a valid key or mouse response) to specify a correct response associated with a slide. This enables OGAMA to track correct or wrong answers that are stored in the trial events database table as a response event. Valid mouse clicks can be restricted to specific target areas as they were defined in the Targets tab.

*Targets:* This tabs allows the optionally definition of rectangular, ellipsoid or polygonal shaped target areas that can be used in the testing section to identify specific valid areas. The most common use for the targets are multiple choice questions, where the experimenter wants to track the users answers. Define rectangular targets around each answer, allow them to be a valid response for slide exchange in the Timing tab but specify only correct answers as a testing target in the testing tab.

*Links:* Sometime jumping between slides in a slideshow is needed to enable quick access to the final slide or different trails through the slideshow. Similar to the timing section one specifies the valid input for a slide exchange and adds the target slide to be jumped to. Note that this may lead to different slide sequences for each recorded subject.

*Background:* An image or color can be assigned to the background of each slide along with the possibility to add background music on appearance or click on the slide. While this is a kind of gimmick, maybe it is useful for someone.

*Mouse:* During presentation sometimes the mouse cursor should be hidden or by default located at a specific starting position. This can be achieved in this tab by defining position and visibility along with the option not to change the cursors initial position, but let it stay at the position where it left the foregoing slide.

*Trigger:* When eyetracking should be synchronized with other devices like EEG or NIRS<sup>2</sup> this tab is suitable to set trigger output device (e.g. LPT, COM) with port address, an 8-bit trigger signal value and the

---

<sup>2</sup> Near infrared spectroscopy, an optical tomography system, see e.g. [www.nirx.net](http://www.nirx.net)

signalling time. When triggering is enabled OGAMA sends the trigger signal at the initial display time of the slide. Note: be sure to test the settings with your connected device to ensure proper communication.

### 2.1.1.3. Preview window

The creation process is WYSIWIG, i.e. each modification is instantly shown in the preview window at the right half of the dialog. To change the draw order of the displayed elements analogue to e.g. Powerpoint use +, -, PgDown, PgUp keys to bring forward or send back. The ALT key helps during selection of covered elements. Right click for a context-menu.

After finishing slide creation, saving stores the slides in xml format in the experiments .oga file. A slideshow can also be exported separately to other OGAMA experiments via export, import buttons in a special xml slideshow (.ogs) format. Image files and flash movies are copied to the 'SlideResources' folder of the experiments main folder to provide independency of the absolute referenced file system (e.g. when moving the experiment to another computer). To preview the complete slideshow for testing purposes under presentation conditions just click onto the 'Preview slideshow' button in the menu tool strip. In this case the slideshow is validated but no recording is created.

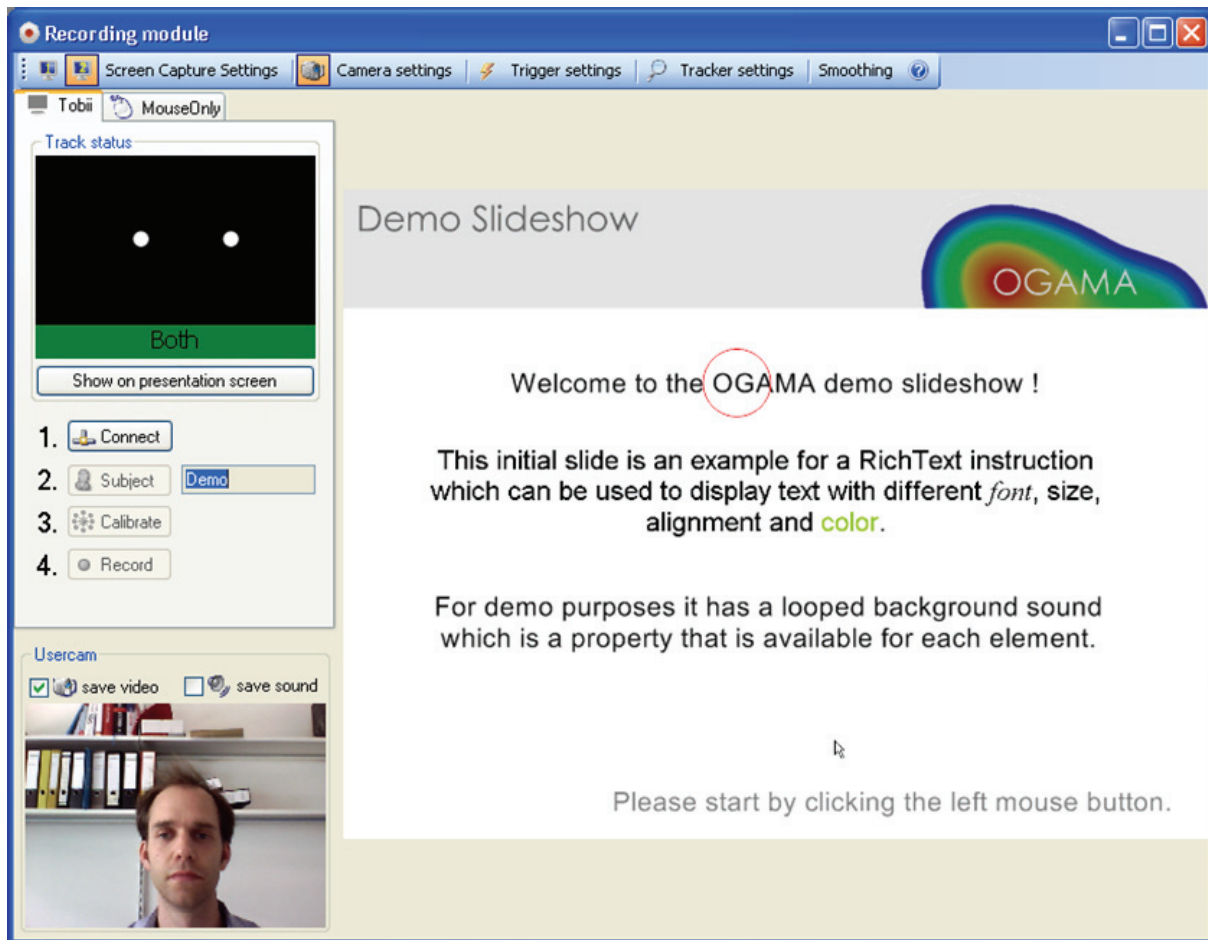
### 2.1.2. Recording

The Recording module is designed to capture subjects' gaze and/or mouse movements during the presentation of the experiments slideshow. Three different tracking systems are currently implemented: OGAMA is able to control and record from Tobii gaze tracking systems<sup>3</sup>, from the Alea Technologies IG-30 system<sup>4</sup> and includes a software mouse tracking device (see Figure 5 for a screenshot of the module).

---

<sup>3</sup> Tobii Technology AB, [www.tobii.com](http://www.tobii.com)

<sup>4</sup> alea technologies, [www.alea-technologies.de](http://www.alea-technologies.de)



**Figure 5: Screenshot of the recording module during capturing, including user video (Windows XP style)**

In general four steps are required to start a recording:

1. Connect: Start the communication between the eye tracker and OGAMA
2. Subject: Opens a dialog to enter subject information (e.g. identification code, age...)
3. Calibration: Start the (selected) eye tracker calibration process
4. Record: Start data recording and show slideshow on the presentation screen

The settings associated with the selected tracking module are edited via a customized dialog (e.g. specifying the trackers' IP Address, the number of calibration points, their speed, size and color). At the end of the experimental procedure the tracked data is stored into OGAMAs database and is immediately available for analysis or replay.

During load of the module the system scans for connected tracking devices and displays the control panels only for available tracker. When a webcam is connected to the computer a user video can be captured. In dual monitor setup environments the presentation monitor can be specified and trigger settings are available for synchronization with other recording devices. Optionally a general trigger for all slides can be defined, so if one just wants to signal beginning of each slide, there is no

need to set the trigger on each slide, but simply activate the general trigger.

In dual monitor setups which are the recommended environment for recording, the record module shows a live view of the current recording including an overlaid gaze and mouse cursor along with a track status for the current tracker, indicating the track quality so that the experimenter has complete control over the recording. Hitting F12 key will record a custom marker to be found in the replay indicating interesting situations e.g. in Usability-Testing scenarios, hitting ESC cancels presentation and recording.

If the slideshow contains flash movies it is recommended to review the screen capture settings available in the menu tool strip to have control over the video compressor and the screen capture device<sup>5</sup>. When recording is finished OGAMA asks for writing the data into the database and immediately calculates gaze and mouse fixations according to the current settings.

### 2.1.3. Data Import

To enable recordings with other tracking devices to be analyzed and visualized within OGAMA one can use the import assistant. It is designed to flexibly support data import from different data sources of various eye tracking systems (see Figure 6 for screenshots of the procedure).

---

<sup>5</sup> OGAMA uses by default the free VH Screen Capture driver from Hmelyoff Labs, [www.hmelyoff.com](http://www.hmelyoff.com)

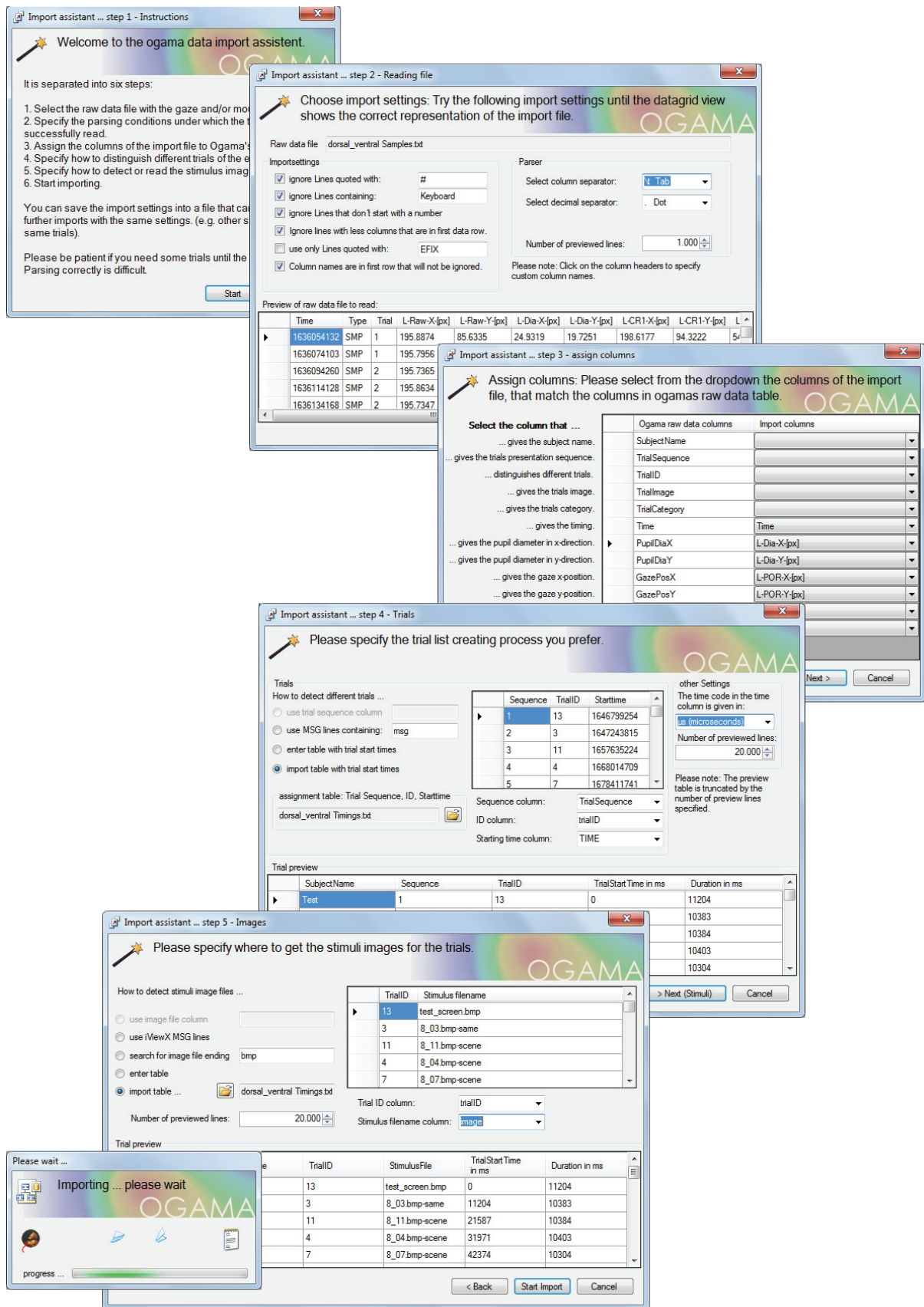


Figure 6: Steps during import of an SMI log file using custom starting times, trialIDs and images (Windows 7 style)



The data import is performed in six steps:

*1. Select the log file with the gaze and/or mouse samples.*

OGAMA can read ASCII files with delimiter separated columns. The log file should contain the whole raw data samples from the recording.

*2. Specify the parsing conditions.*

To successfully read the log file, one has to specify the column delimiter (e.g. semicolon or tab), decimal separator and the string that identifies quotes which should be ignored by the import assistant. It is also possible to skip lines from the log file that don't start with a number or contain a specific key (e.g. to omit all lines containing keyboard events) or to parse only lines starting with another specific key (e.g. EFIX to parse only fixation rows in EyeLink data files). In addition, the user can select whether the first readable row contains the column names. After each modification a preview of the log file with the detected columns is shown in a table, where the column names can be modified by clicking on the header.

*3. Assign columns*

To guarantee a matching between the import file columns from different sources and OGAMAs internal database columns, the next step is to assign the relevant columns of the log file to those of OGAMA. For example, assume a 'Tobii 1750' EFD log file with the columns "Timestamp", "Found", "GazepointX" and "GazepointY". In a dialog the correct assignment is done by selecting the log file columns from the drop down buttons. If no subject name column is specified it is assumed that the whole file data is from the same subject and a dialog asks for a unique subject name.

*4. Specify how to distinguish different trials of the experiment.*

OGAMAs import assistant provides three different trial specification modes:

A) In each raw data row a column with the current trial number is present (a column with an incremental trigger, e.g., iViewX System, SMI)

B) The trial separating rows contain a unique string like "MSG" (iViewX System, SMI) or "TRIALID" (EyeLink System, SR research)

C) A table with starting time and trial id columns. This can be imported from an additional trial definition file (Tobii 1750 EVD file) or entered manually.

In this step, the user is also prompted to specify the time code unit.

*5. Specify how to detect or read the stimulus images for each trial.*

In the last step one has to extract the stimulus image file names from the log file to assign them to the imported trial ids. This step works similarly to the trial import (see step 4). Every change is previewed in a data grid view, so the settings can be controlled instantly. Be sure at this step, that your file is parsed correctly by verifying the preview table. Eventually increase the number of previewed lines.

#### *6. Start importing.*

To facilitate the import of similarly structured log files, the user is asked to save the import settings to a file that can be used at the beginning of the next import. During the data import all raw data (including zero values or values that are out of screen) are wrapped into OGAMAs database which enables a customized handling in the fixation detection algorithm and visualization.

#### **2.1.4. Database control**

The Database module is designed to manually view and revise all tables in the experiments database which are the subjects and subject parameters table, the trials and trial events table, the gaze and mouse fixations table, the areas of interest and its shape groups table along with all raw data tables with the sampling data. The user can edit and revise each database entry. To avoid very large table loading times the data is filtered by subject and trial by default, but one can deselect the filter button to get the complete data table (see Figure 7 for a screenshot).

The screenshot shows a Windows-style application window titled "Database Module". It features a menu bar with "Import Assistant" and a toolbar with various icons. Below the toolbar is a tabbed interface with tabs for "Subjects", "Subject parameters", "CustomParams", "Trials", "Trial Events", "Raw data", "Gaze Fixations", and "Mouse Fixations". The main area displays a table with the following columns: ID, SubjectName, TrialSequence, EventID, EventTime, Event Type, EventTask, and EventParam. The table contains 20 rows of data, all for SubjectName "VPN02".

ID	SubjectName	TrialSequence	EventID	EventTime	Event Type	EventTask	EventParam
5636	VPN02	66	5635	29496	Response	SlideChange	Mouse: Left (712,457), at target: Target 1. Co...
5637	VPN02	67	5636	1997	Response	SlideChange	Time: 2000 ms
11376	VPN02	68	338	3491	Mouse	Down	Mouse: Left (410,430)
5638	VPN02	68	5637	3541	Response	SlideChange	Mouse: Left (410,430), Wrong response to th...
11377	VPN02	69	339	4273	Mouse	Down	Mouse: Left (246,251), at target: Target 1
5639	VPN02	69	5638	4382	Response	SlideChange	Mouse: Left (246,251), at target: Target 1. Co...
5640	VPN02	70	5639	2005	Response	SlideChange	Time: 2000 ms
11378	VPN02	71	340	3248	Mouse	Down	Mouse: Left (398,388), at target: Target 1
5641	VPN02	71	5640	3303	Response	SlideChange	Mouse: Left (398,388), at target: Target 1. Co...
11379	VPN02	72	341	1272	Mouse	Down	Mouse: Left (447,528)
11380	VPN02	72	342	3428	Mouse	Down	Mouse: Left (567,685), at target: Target 1
5642	VPN02	72	5641	3524	Response	SlideChange	Mouse: Left (567,685), at target: Target 1. Co...
5643	VPN02	73	5642	1997	Response	SlideChange	Time: 2000 ms
11381	VPN02	74	343	3825	Mouse	Down	Mouse: Left (636,346), at target: Target 1
1738	VPN02	74	1737	3872	Response	SlideChange	Mouse: Left (636,346), at target: Target 1. Co...
11382	VPN02	75	344	1402	Mouse	Down	Mouse: Left (747,506)
11383	VPN02	75	345	1971	Mouse	Down	Mouse: Left (750,330)
11384	VPN02	75	346	3984	Mouse	Down	Mouse: Left (285,542)
11385	VPN02	75	347	4596	Mouse	Down	Mouse: Left (245,559)
11386	VPN02	75	348	5370	Mouse	Down	Mouse: Left (273,443)
11387	VPN02	75	349	6249	Mouse	Down	Mouse: Left (539,236)
11388	VPN02	75	350	6940	Mouse	Down	Mouse: Left (549,391)
11389	VPN02	75	351	7659	Mouse	Down	Mouse: Left (551,300)
11390	VPN02	75	352	8413	Mouse	Down	Mouse: Left (756,185), at target: Target 1
1739	VPN02	75	1738	8517	Response	SlideChange	Mouse: Left (756,185), at target: Target 1. Co...

**Figure 7: Screenshot of OGAMAs database module. (Windows 7 style)**

Additionally the import assistant for importing custom sample data can be started (see 2.1.3) along with export functionality for selected subjects.

## 2.2. Visual data analysis

Once gaze and/or mouse data is available in the database follow the workflow diagram depicted in Figure 2 to start a quick survey of the recorded or imported samples with the replay module.

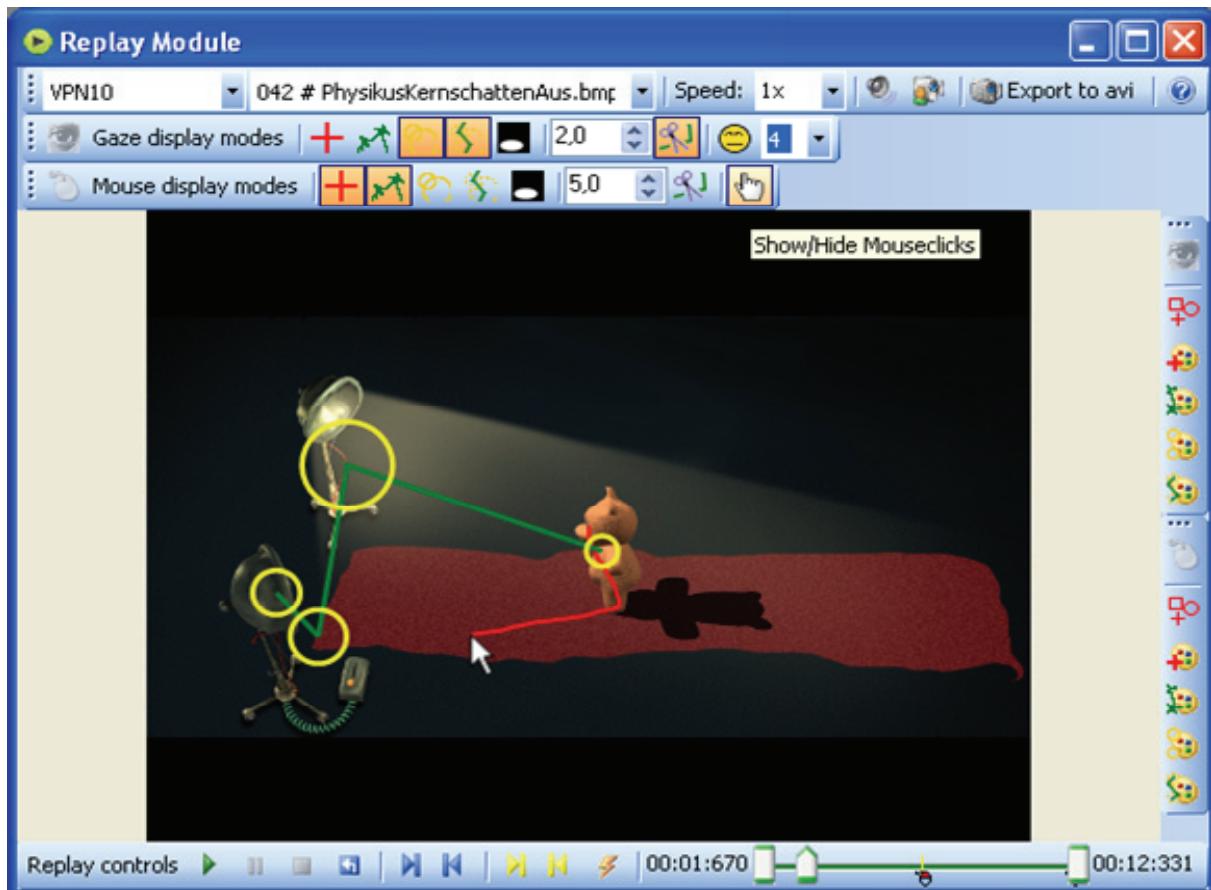
### 2.2.1. Replay

The Replay module is designed to replay the data logged during an experiment (see Figure 8 for a screenshot). Its intended use is to easily visualize and record single subject gaze and mouse tracking segments. To support this, different drawing modes are available for gaze and mouse tracking data (or both). Depending on visualization needs the drawing modes can be combined arbitrarily.

For the replay of the sampling data the user can select the following modes:

1. Cursor - Draw a cursor at the current sample point.
2. Path - Draw a polyline consisting of the sample points up to the current sample point.

3. Fixations - Draw circles that represent fixation position and fixation duration.
4. Connections - Draw straight lines that connect the fixation circles.
5. Spotlight - Shows the current sample point position as a circle of the original stimulus overlaid on the greyed stimulus slide.



**Figure 8: Screenshot of OGAMA's replay module. (Windows XP style)**

To replay a certain tracking segment, a specific subject and trial can be selected from drop down controls or the context panel of the main window. Further controls provide the possibility to change the replay speed (real-time, faster or slower), the drawing modes, shape size and colour of the cursors, and the pen styles of all drawing elements. Available user videos and the corresponding sound can be switched on and off independently.

The timing control is achieved by a time slider and play, pause, stop and reset buttons. Just drag the time caret to move to a custom position in the trial. Markers in the timeline indicating trial events like mouse down, mouse up, key events etc. can be accessed via jump buttons similar to the possibility to jump to the time stamp markers set during recording via the F12 key. If there is the need for subsequent marker creation or modification just drag and drop the existing markers or create new one with the provided button. This is of course only available for the time

stamp markers. If the trial duration is very long and there occurred a lot of events, right-click on the timeline to specify a display filter (e.g. hide mouse and key events).

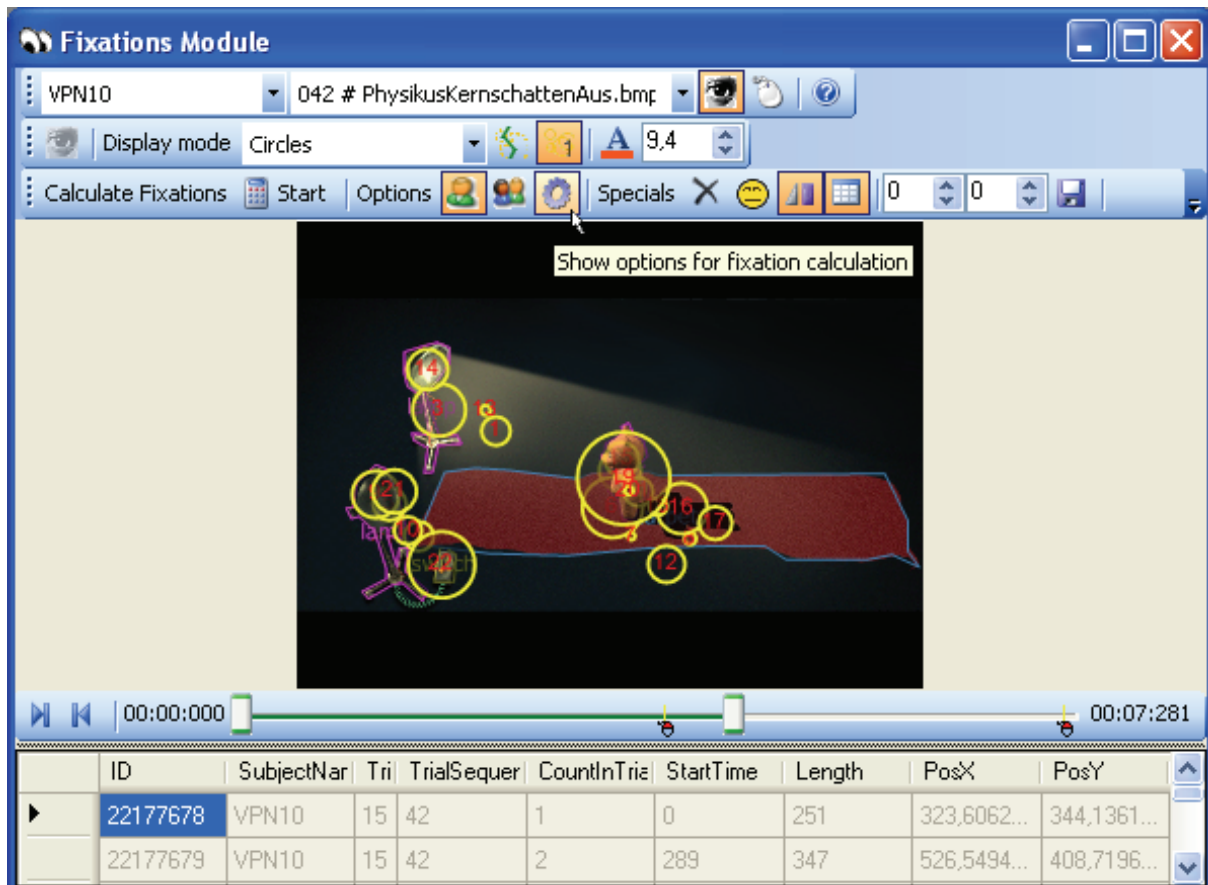
Video recording is done via an 'export to avi' button. The recorded .avi files with custom size and compression are saved on hard disk. Note that the Replay module is a stand-alone tool which calculates fixation data independent of pre-calculated fixations (2.2.2) in real time for the selected trial. This allows a quick check of fixation detection parameters (see 1.4).

### 2.2.2. Fixations

This Fixations module is designed to recalculate, store and display the gaze and mouse fixations made by the subjects (see Figure 9Figure 14 for a screenshot). It is recommended to calculate the fixations for all subjects, right after data import. These are stored in OGAMAs database and afterwards available for the Attention Map module and the Statistics module. Note that changing or adding raw data recommends a new calculation of the fixations.

The Fixation module contains a canvas for visualization and a data table for each gaze and mouse fixation. Five display modes are available, each to be complemented with the option to display the fixation numbers and connections.

1. *Dots* - Draw fixations as a filled round dot.
2. *Circles* - Draw fixations as empty circles with a circle size that is proportional to the fixation duration.
3. *Spotlight* - Draw fixations as a circle of the original image overlaid on the greyed stimulus slide, with a circle size that is proportional to the fixation duration.
4. *Attention Map* – Draw fixations as a coloured heat map. We implemented heat maps as superimposed Gaussian distributions of all fixations in a trial (see 0 for a description of the algorithm).
5. *None* – Only connections and/ or fixation numbers are displayed if selected



**Figure 9: Screenshot of OGAMA's fixations module. (Windows XP style)**

The data table lists the number of the fixation, the calculated starting time, duration and position of each fixation for the current trial. Trials can be selected by subject name and trial number from a drop down control. To facilitate analysis, defined areas of interest (see 2.3.1) can be displayed. It should be noted, that the fixation detection parameters are fixed for the fixation calculation of a whole experiment, but can be edited in the experiments options dialog of the main window. Further controls include deletion of fixations, mark trials as invalid to exclude them from statistical analysis (e.g. if the calibration has a remarkable drift), pen and font style for graphical elements and a button to hide the data table to increase the canvas size.

Similar to the replay module (see 2.2.1) the time range to be displayed can be restricted by moving the left and right border caret thumbs of the timeline.

### 2.2.3. Saliency

The Saliency module provides the experimenter with the possibility to compare recorded human scan paths with the scan path generated by a computational model of vision (see Figure 10 for a screenshot). It uses the source code of the ilab toolkit to calculate bottom up saliency maps of the selected slide (see e.g. Itti & Koch 2001, <http://ilab.usc.edu/toolkit/>). It is intended to provide the user with the results of a computational model of vision for the first few fixations when looking at a slide. The underlying model contains a lot of parameters to precisely specify the conditions for the model which are e.g. different channels of visual processing and other options that can be adjusted via OGAMAs interface to the model. As a result the calculated most salient locations are transformed into a fixation sequence which is displayed with the known available fixation drawing modes (see e.g. 2.2.2). For detailed research the channel maps of the calculation along with the resulting saliency maps are available.

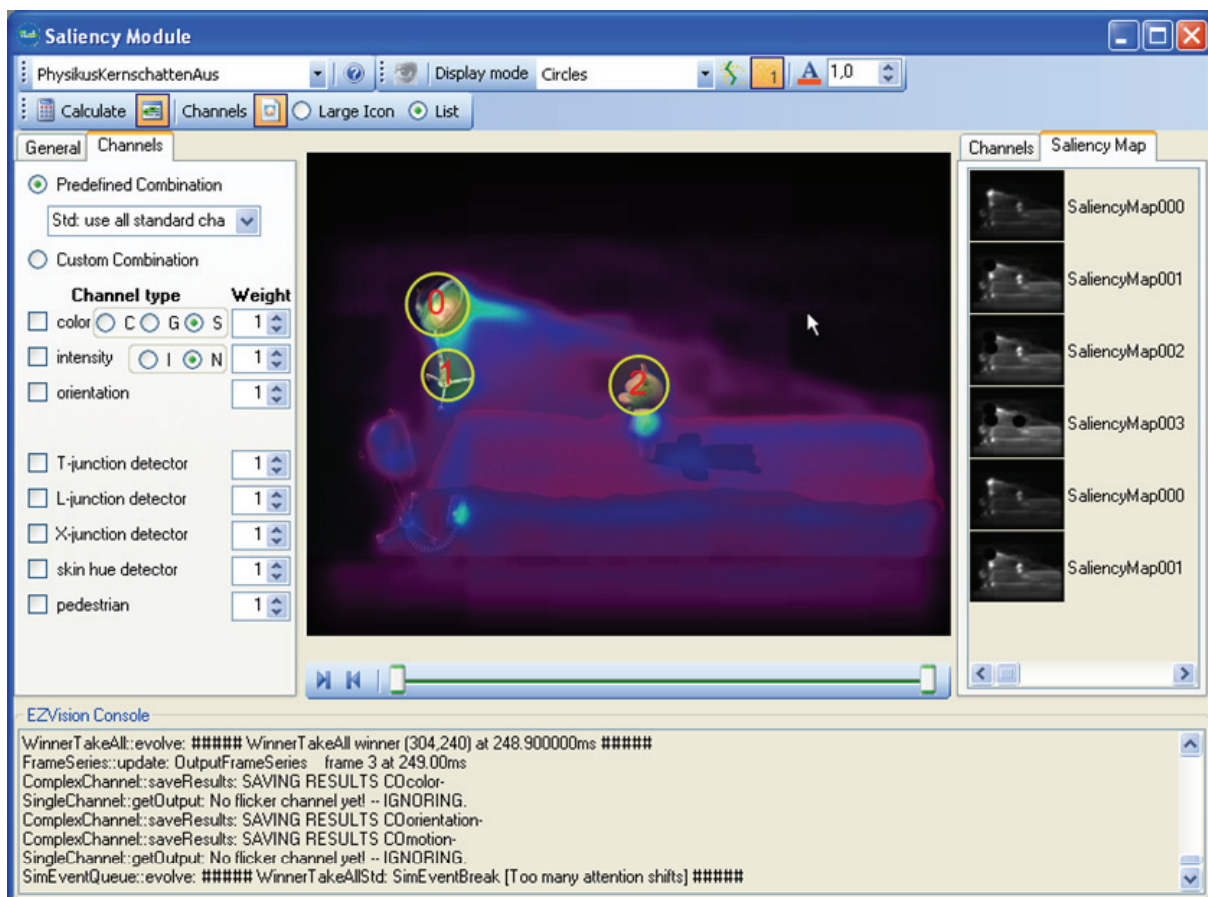


Figure 10: Screenshot of OGAMAs saliency module. (Windows XP style)

## 2.2.4. Scan path

The scan path module is the first module to visualize data of multiple subjects in one place (see Figure 11/14 for a screenshot). It is based on the calculated fixations. On the left side the subjects are grouped by their category with each subject or group assigned a different colour and can be included or excluded from display by checking or unchecking. When selecting a trial the fixations of the selected subjects are displayed in one of the modes known from the fixations module (see 2.2.2) but each coloured in one of the following modes

1. *Subject* – Each subject is assigned a custom colour, by default colored with a gradient.
2. *Group* – Each subject group is assigned a custom colour, all fixations of subjects in the same group are shown in the same colour.

This allows subject- or group-wise comparison of fixation distribution and scan path. Replay functionality can be activated analogue to the replay module (see 2.2.1).

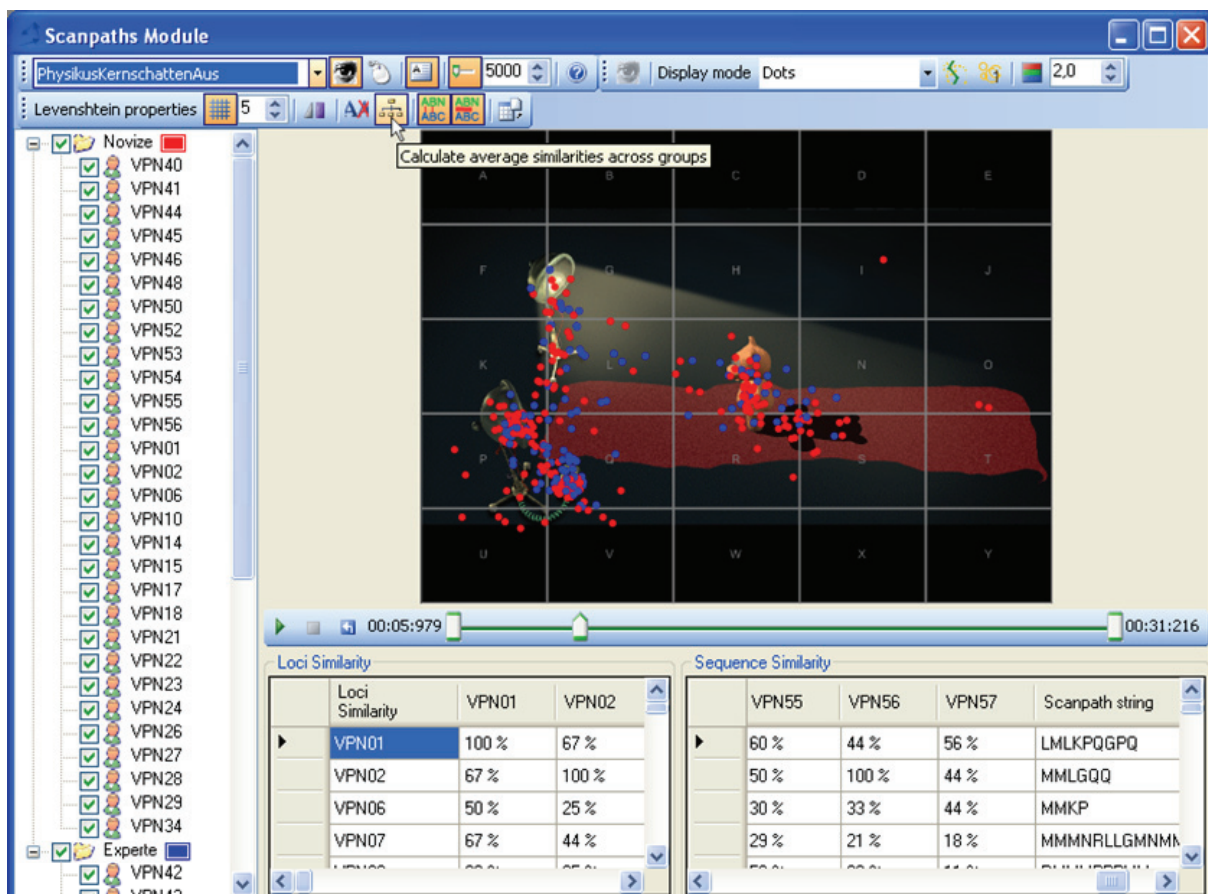


Figure 11: Screenshot of OGAMA's scan path module. (Windows XP style)

OGAMA also provides a numerical analysis of scan paths by calculating Levenshtein distances. Two parameter matrices were calculated, the loci similarity and the sequence similarity in comparison to each other subjects scan path. To achieve this, slide is fielded into a grid with each



field named with a unique letter. Then each scan path is recoded into a string by assigning the letter of the field that contains the current fixation location. This recoding can be done with or without ignoring subsequent fixations in the same field. The loci similarity between two scan paths is now defined as the percentage of letters that the first string matches the second string. Translated into gaze movements this is the percentage of locations both scan paths have passed by, independently of time and sequence. A value of 100% indicates that fixations of both subjects were at the same places.

The sequence similarity is the percent of character sequences that are concordant in both strings. It is based on the Levenshtein distance which counts the number of operations needed to turn one string into the other (see Brandt & Stark 1997; Levenshtein 1966). A value of 100% indicates identical scan paths.<sup>6</sup>

Further controls provide visualization properties, export of scan path comparison matrices and simplification of the user interface.

### 2.2.5. Attention Maps

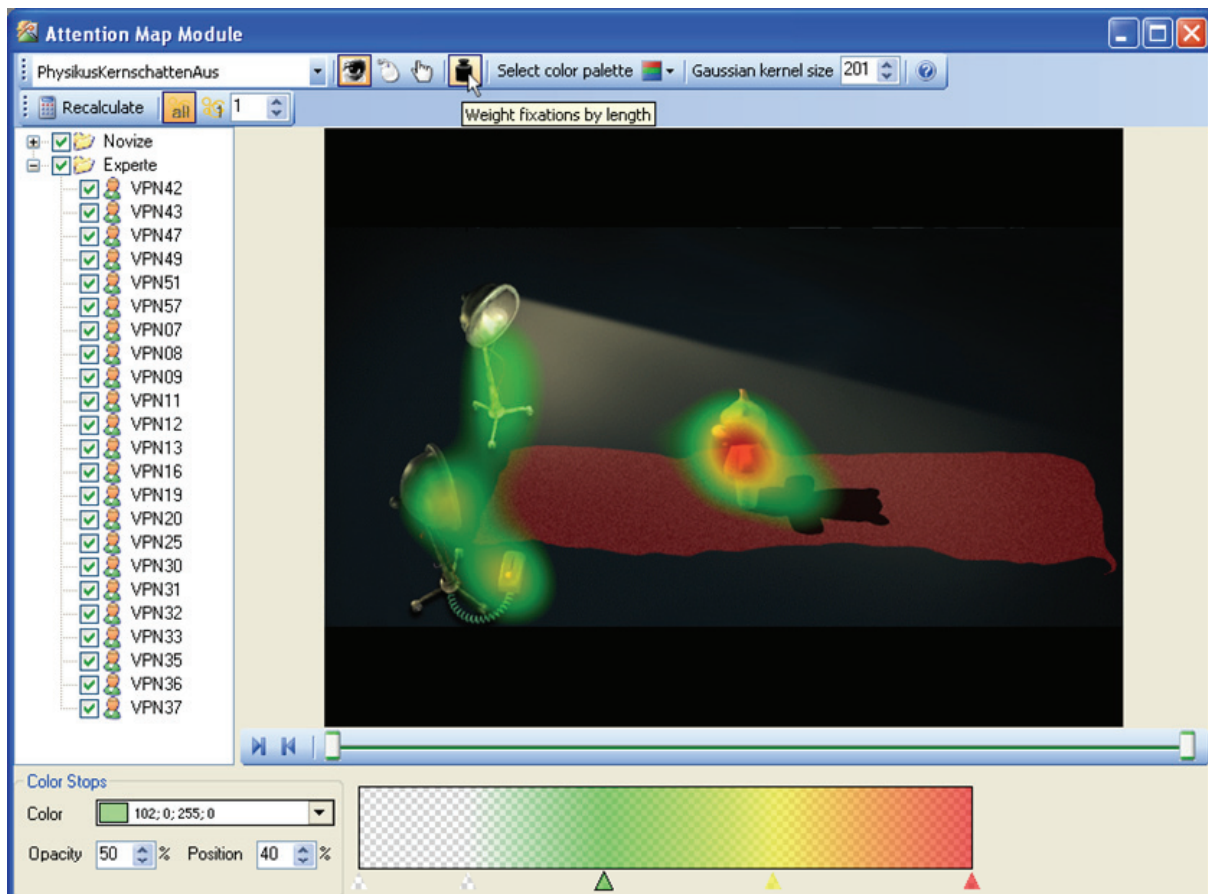
The Attention Maps module is designed to visualize merged fixation data from selected subjects (see Figure 12Figure 14 for a screenshot). This can be used to identify regions of special attention in a specific stimulus slide. In addition to the attention maps computed in the Fixation module, the Attention Maps module allows to build averaged maps across a subset of (or all) subjects.

The attention maps are calculated as aggregated Gaussian distributions of each fixation in a stimulus slide. The summed Gaussian distributions are then overlaid on the original stimulus slide. This allows a quick visualization of a landscape of visited and unvisited locations on the stimulus slide (Spakov & Miniotas 2007). The attention map calculation uses a two dimensional Gaussian kernel with an editable size to enable adjustable smoothing of fixation distribution landscapes. With a mean value of  $\mu = 0$ , a standard deviation of  $\sigma$  and an isotropic distribution it is given by the following formula.

$$f(x, y) = \frac{e^{-\frac{x^2+y^2}{2\sigma^2}}}{2\pi\sigma^2}, x, y \in [-s, s]$$

---

<sup>6</sup> Numerous other comparison algorithms are proposed; see section **Fehler! Verweisquelle konnte nicht gefunden werden.** for a review. OGAMA currently only supports the Levenshtein distance; the Malsburg distance is already coded, but not available from the frontend.



**Figure 12: Screenshot of OGAMAs attention map module. (Windows XP style)**

The standard deviation is internally set to  $\sigma = s/5$ . Each value of this template kernel is multiplied for each fixation with a factor that is given by the duration of the fixation. This factor represents the weighting of this fixation. All multiplied kernels in a trial are then added to a stimulus sized array at the positions of the respective fixations. Finally, the whole array is normalized. The result is a stimulus sized height map or landscape (e.g. Spakov & Miniotas 2007). This landscape can be easily transformed into a colour map by a predefined or custom gradient. Figure 12 shows an example image that uses a ‘traffic sign’ gradient where the hot spot is overlaid in red. Another pre-defined gradient is called ‘black mask’ which hides unseen areas of the original image with a black mask.

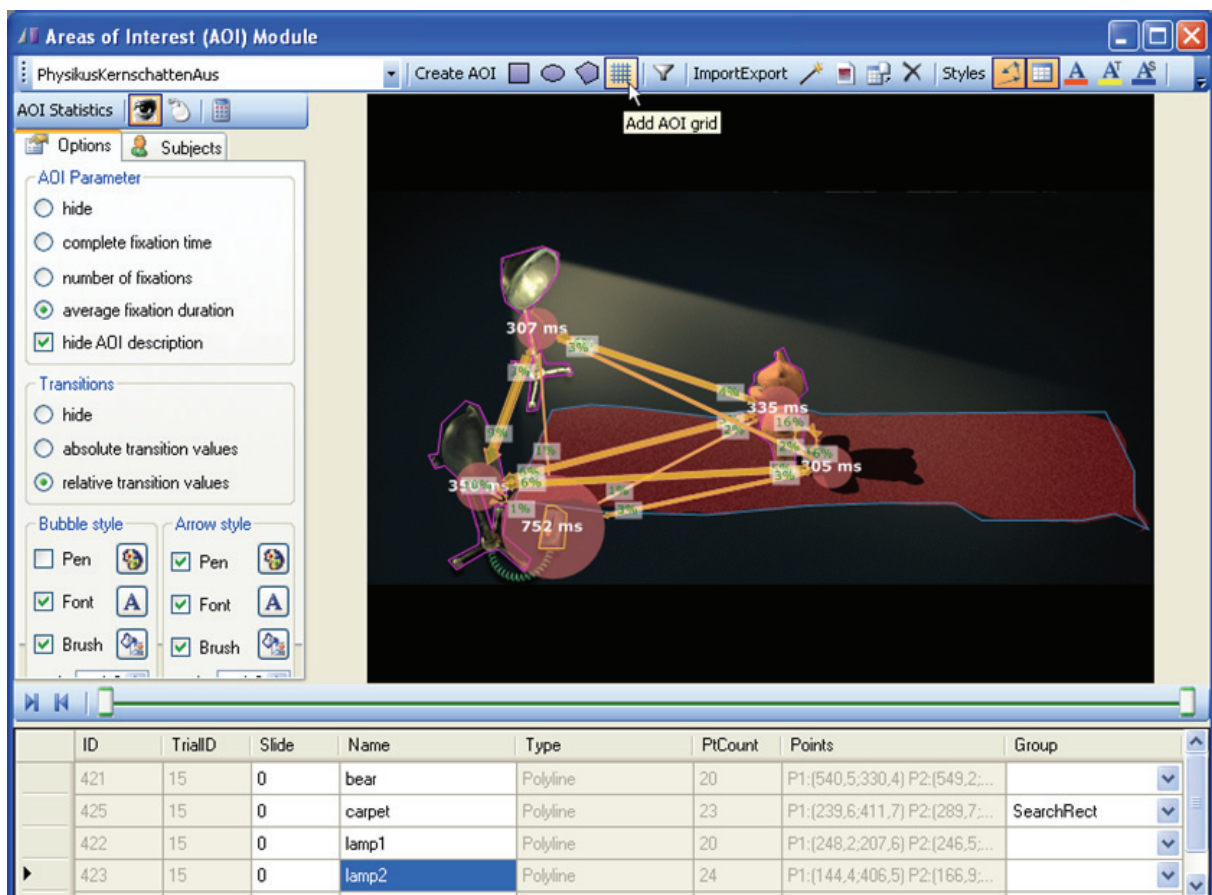
The user can choose the subjects to include in the calculation and whether the calculation should be based only on a specific (e.g. first) or on all fixations along with the option to modify the size of the Gaussian kernel in pixel. So one can easily answer questions like “Where the average subject does has its first look” and “What locations of the stimulus are not noticed by the average subject”.

## 2.3. Statistical data analysis

The above described modules are primarily intended to qualitatively survey the recorded data. Replay videos, heat maps, fixation distributions are mainly used to provide first insight in the experiment, find hints for usability issues and hopefully propose pathways for successful statistical analyses. They can be done with the modules described in the next section.

### 2.3.1. Areas of Interest

The Areas of Interest module is designed to define and display different areas of interest (AOI) (see Figure 13 Figure 14 for a screenshot). Three types of AOI shapes are available: simple rectangles, ellipses and polygons. After selecting the type of the AOI, one specifies location and size of the AOI with the mouse on the stimulus slides. Another option is to import target areas defined in the slideshow. Shape and size can still be modified in later processing stages with the help of grab handles.



**Figure 13: Screenshot of OGAMAs areas of interest module. (Windows XP style)**

The AOI data is stored in a separate database table together with the trial id and slide number they belong to. The databases entries are shown in a data table, where editing is allowed for renaming and fine tuning the size and position, or deleting. In the data table the AOI can be

categorized by grouping them with specific names. These categories are used in the statistical output. Predefined values are “Target” and “SearchRect” which are used for example to generate variables like “time left until first fixation in ‘Target’ AOI” or “number of fixations in ‘SearchRect’ “.

Similar to the foregoing modules, the user can change the pen style to allow visible differentiation. This is possible for the predefined shape groups. Further, the Areas of interest module provides the possibility of importing a list of rectangular AOI from a text file through an import assistant similar to section 2.1.3 or import/export AOI from/to another OGAMA project.

After defining the AOI an area with controls for the calculation of AOI related statistics can be shown. After selecting the subjects to be included into the calculation, three parameters concerning each AOI can be displayed:

1. *Complete fixation time* – This is the sum of all fixation durations for the selected subjects on the given AOI
2. *Number of fixations* – This is the count of all fixations for the selected subjects on the given AOI
3. *Average fixation duration* – This is the sum of all fixation durations divided by the number of all fixations for the selected subjects on the given AOI

These parameters are displayed with bubbles centred at the AOI centre and sized by the parameters value, which is written at the centre of the bubble overlaid on the slide (see Figure 13). They cannot be displayed concurrently.

Additionally transition arrows are calculated and shown which show the number of transitions of fixations from one AOI to another. The transitions from an AOI to nowhere or self-transitions are currently not displayed but can be calculated in the statistic module. Lots of style controls ensure the adaption needed to provide readable statistical overlay outputs. Note: Pressing Ctrl-C in a module copies the content of the canvas in the original high resolution of the recording to the clipboard to be used in other applications.

## 2.3.2. Statistics

The Statistics module is designed to calculate empirical parameters that are useful for further analyses (see Figure 14 for a screenshot). There are four different scopes of variables available: subject parameters, trial parameters, gaze parameters, and mouse parameters. The parameters are marked for calculation by selecting checkboxes. Once finished the parameters are calculated and submitted to a results data table. This table can be exported to an ASCII-file which is importable in post processing software like SAS or SPSS.

Figure 14: Screenshot of OGAMAs statistics module. (Windows XP style)

### a) Subject parameters

Subject parameters are objective parameters to describe subject's properties. These predefined variables refer to the subject and subject parameter table of OGAMAs database. In detail, there are the unique subject name (e.g. a subject ID, 'AZ012'), a subject category (e.g. 'physicist'), Age, Sex, Handedness, custom defined parameters which can be assigned by the user and a comments column for additional notes.

### b) Trial parameters

Trial parameters categorize the different trials. Trial parameters are split in five sections: 'identification', 'timing', 'areas of interest', 'loss of data' and 'further options'.

The section "identification" consists of the unique trial id and sequence, the trial name and a descriptive stimulus category (which can be used for grouping trials, like "first seen" or "second visit").

The section "Timing" contains the trial duration and the starting time (in milliseconds) with respect to the whole experiment.

"Areas of interest" contains the sum of the sizes of all AOI in percent (relative to the presentation screen size) and the relative size of the 'Target' AOI in percent (in relation to the presentation screen size). This size can be used for calculation of Fitts law (Accot & Zhai, 2003). "Areas of interest" further contains the AOI of response, which describes the AOI the user clicked if the slide was finished with a mouse click.

The fourth section "loss of data" provides a count of the data lost due to blinks and out of monitor samples.

Finally the 'further options' section contains three parameters, the first is the subjects response recorded during the experiment, the second indicates the correctness of the response (if it is applicable and has been defined in the slideshow design, see 2.1.1) and the third column indicates manually excluded trials. The latter column contains "yes" values when the trial is manually marked for exclusion in the Fixations module.

### **c) Gaze parameters**

Gaze parameters are split in five sections: 'Fixation general', 'Gaze path', 'Gaze Mouse interaction', 'Gaze at AOI (predefined)' and 'Gaze at AOI (custom)'.

General fixation parameters in the 'Fixation general' section include number of fixations, fixation count per second, average fixation duration and the fixations/saccades ratio.

'Gaze path' refers to the gaze path length in a trial. It is calculated by adding the lengths of the fixation connections to exclude micro saccades and data collection noise. The user can select between two output parameters: absolute gaze path length in pixels and gaze path velocity in pixel per second.

The section 'Gaze mouse interaction' contains two parameters. The first reports the number of fixations until the first mouse click and the second calculates the average distance between gaze and mouse path.

'Gaze at AOI (predefined)' refers to categorized areas of interest. This section evaluates the time until the subject first fixated the AOI referred

as “Target” or “SearchRect” and the amount of time that the subject fixated the ‘Target’ AOI at all. Additionally, the time until the second fixation at the target AOI can be calculated.

‘Gaze at AOI (custom)’ provides a custom definition of AOI related parameters. The four parameters ‘complete fixation time’, ‘time until X fixation’, ‘number of fixations’ and ‘average fixation duration’ can be calculated referring to a selected AOI group or single AOI. This allows even artificial variables like the calculation of the “time until the third fixation at the AOI named ‘painting’ ”.

#### **d) Mouse parameters**

In the mouse parameters tab of the Statistics module the gaze parameters are also calculated for mouse movements (except the average gaze and mouse path distance). Additionally, the user can select the evaluation of click counts and click rates for left and right mouse button and the time until the first mouse click event. Consequently the custom parameters section is extended with a fifth parameter “clicks of button: X” at selected AOI group or single AOI. To calculate variables like “Number of clicks of left mouse button at AOI ‘click me’ “.

### 3. Conclusion

We introduce OGAMA to expand the research capabilities in screen-based eye tracking studies. OGAMA is a stand-alone application running on Microsoft® Windows® platforms and designed to handle data from a wide range of experiments in psychology and usability research. Thus, it is optimized for the analysis of slide show based stimulus presentations including flash movies. It handles mouse movements just the same as gaze movements and has the ability to calculate connecting parameters between mouse and gaze data. It also supports most of the commonly used qualitative analysis and visualization tools in the field of eye tracking research like replay, attention maps, saliency maps and areas of interest along with tools for a quantitative look at the data.

Researches from the community are encouraged to further develop their own statistical variables by adapting OGAMAs source code. To support this process, we provide OGAMA with extensive source documentation. Its open source character allows also developing custom modules, for example a specialized reading graphs analysis module. At this point, OGAMA is flexible and powerful and we would be happy to see it being useful in the eye tracking research community. OGAMA is available along with an extensive demo project via the URL: <http://didaktik.physik.fu-berlin.de/OGAMA>.



## 4. References

- Brandt, S. A., & Stark, L. W. (1997). Spontaneous Eye Movements During Visual Imagery Reflect the Content of the Visual Scene. *Journal of Cognitive Neuroscience*, 9 (1), 27-38. MIT Press. doi: 10.1162/jocn.1997.9.1.27.
- Chen, M., Anderson, J., & Sohn, M. (2001). What can a mouse cursor tell us more? Correlation of eye/mouse movements on web browsing. In *Conference on Human Factors in Computing Systems* (p. 281–282). New York, USA: ACM Press.
- Gitelman, D. R. (2002). ILAB: A program for postexperimental eye movement analysis. *Behavior Research Methods, Instruments, & Computers*, 34 (4), 605-612.
- Karsh, R., & Breitenbach, F. (1983). Looking At Looking: The Amorphous Fixation Measure. In R. Groner, C. Menz, D. Fisher, & R. Monty, *Eye Movements And Psychological Functions: International Views* (pp. 53-64). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Kirstein, J., & Nordmeier, V. (2007). Multimedia representation of experiments in physics. *European Journal of Physics*, 28 (3), 115-126. doi: 10.1088/0143-0807/28/3/S11.
- LC Technologies. (2006). *Fixation Functions Source Code*. Fairfax, Virginia, USA: LC Technologies.
- Levenshtein, V. (1966). Binary Codes Capable of Correcting Deletions and Insertions and Reversals. *Soviet Physics Doklady*, (8), 707-710.
- Salvucci, D. D., & Goldberg, J. H. (2000). Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the symposium on Eye tracking research & applications - ETRA '00* (pp. 71-78). New York, USA: ACM Press. doi: 10.1145/355017.355028.
- Shneiderman, B., & Plaisant, C. (2004). *Designing the user interface* (4th Edition.). Boston, MA: Addison-Wesley.
- Spakov, O., & Miniotas, D. (2007). Visualization of Eye Gaze Data using Heat Maps. *Electronica and electrical engeneering*, 2 (74), 55–58.
- Tan, P., Steinbach, M., & Kumar, V. (2005). *Introduction to data mining*. Boston, MA, USA: Addison-Wesley Longman Publishing Co.