

FeaturEyeTrack: A Vector Tile-Based Eye Tracking Framework for Interactive Maps

Fabian Göbel
ETH Zürich
Institute of Cartography and
Geoinformation
Stefano-Franscini-Platz 5
Zurich, Switzerland
goebelf@ethz.ch

Peter Kiefer
ETH Zürich
Institute of Cartography and
Geoinformation
Stefano-Franscini-Platz 5
Zurich, Switzerland
pekiefer@ethz.ch

Martin Raubal
ETH Zürich
Institute of Cartography and
Geoinformation
Stefano-Franscini-Platz 5
Zurich, Switzerland
mraubal@ethz.ch

Abstract

A promising way to gain more insights into how people read and understand a map is by analyzing their eye gaze while working with a map. However, eye gaze logging on interactive maps is challenging due to various reasons. One of the biggest challenges comes from the interaction with a map that can drastically change the visual representation of the depicted content, making it difficult to determine what exactly is visible at a given screen coordinate in a given moment. This results in labor-intensive data analysis and study preparation. This paper presents FeatureEyeTrack, a framework for logging a user's visual attention while working with interactive online maps, which is based on vector tile maps. Instead of logging only gaze coordinates, we determine which visual map feature is located at the gaze coordinate, allowing for very detailed analyses on which features, when, for how long, and in which sequence have been looked at.

Keywords: Eye Tracking, User Logging, Interactive Maps

1 Visual Attention and Cartography

Map reading is a visual task that can strongly vary between individuals depending on aspects such as background knowledge, mental model, and task. Furthermore, the visual design of a map also has a great impact on how it is visually explored (Slocum *et al.*, 2009). Understanding how people read maps is important for the optimization of map design which can make map usage more effective and efficient (Montello, 2002; Lobben, 2004).

The logging and analysis of visual attention has been part of cartographic methodology for quite some time (see Kiefer *et al.*, 2017, for an overview). Nowadays, however, as online cartography and web mapping services, such as Google Maps or Open Street Maps, are omnipresent new challenges arise. Compared to static maps, the depicted content of such maps can change through interaction like panning and zooming. Consequently, the mapping of gaze coordinates on the screen to map content (i.e., which feature is looked at?) becomes dynamic and more challenging.

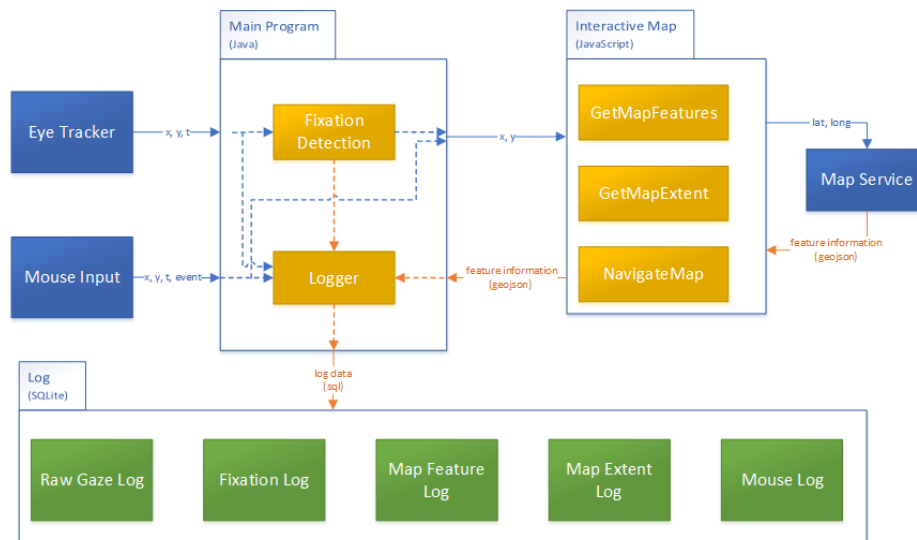
In this paper, we present a prototype of a framework called FeaturEyeTrack for logging a user's gaze and mouse input during the interaction with digital interactive maps. As a particular novelty, FeaturEyeTrack enables the matching of gaze with a vector model of the current map visualization. This allows for very detailed analyses of gaze sequence and visited map features without the requirement for manual annotation.

Figure 1: Setup for an eye tracking study on an online vector map with the FeaturEyeTrack framework.



In section 2, we review previous work on eye gaze logging and analysis on digital maps, and describe the involved challenges. Our framework, a prototype implementation, and a demonstration of its feasibility are presented in section 3. Section 4 provides a conclusion and ideas for future work.

Figure 2: Overview of the eye tracking framework with the three components “Main Program”, “Interactive Map” and “Log”



2 Eye Tracking on Maps

Eye trackers are camera-based devices, which measure an individual’s visual attention (see Figure 1 for an example of a Tobii TX300 eye tracking system). The raw gaze data stream delivered by an eye tracker consists of triples in the form of (x,y,t) , which are usually aggregated spatio-temporally to eye movement events called *fixations* and *saccades*. A fixation appears when the eye rests relatively still over a certain time, whereas saccades denote the rapid movements of the eyes between fixations. Here, we are mainly interested in raw gazes and fixations, more particularly: what was visible on the screen when a certain fixation happened?

In cartography, there are generally two types of eye tracking analyses: content-independent and content-dependent analyses. The first type considers statistics on eye movements only, without taking into account what has been looked at. For instance, the average duration of fixations may correlate with the size of a map area a user has interpreted (Ooms *et al.*, 2012).

The latter type of analyses (content-dependent) investigates which features the user has looked at on the map. A common approach for this type presents (aggregated) fixations as an overlay on the map for visual analysis by an expert who interprets and reasons about what the user has been focusing (Andrienko *et al.*, 2012; Fabrikant *et al.*, 2008).

As an alternative, the investigator may use standard eye tracking software to create polygons (called Areas of Interest, AOI) on the stimulus which enables statistical analyses about the distribution and transition of visual attention on and between these AOIs (Holmqvist *et al.*, 2011). One drawback of this approach is that all AOIs must be created manually, which can be very time and labor intensive, depending on the level of detail required by the research question. For interactive web maps, manual AOI creation becomes virtually impossible because the depicted content may change rapidly and dramatically through user interaction (e.g., zooming, panning, layer toggling).

Therefore, it has been proposed that – for interactive maps – geo coordinates on the screen should be transformed into, and logged as, geo-coordinates based on the current map extent (Giannopoulos, Kiefer and Raubal, 2012; Ooms *et al.*, 2015).

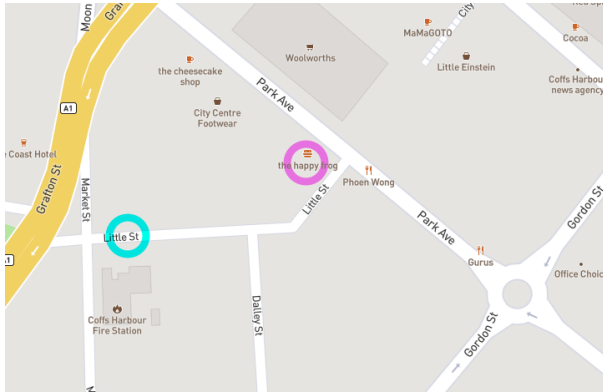
However, later reproduction of what exactly has been looked at is only possible when the map visualization during the study and the evaluation is identical. In particular, it is not sufficient to match the geo-coordinates with the features stored in a geographic information system (GIS) database since the visualization of these features depends on the zoom level and the properties of the cartographic rendering algorithm, such as styles, aggregation, label placement rules, etc. Therefore, determining geo-coordinates of fixations alone is not suitable for reproducing what the user has been seeing on the map.

Besides the manual coding of a screen video, which is very labor intensive, another approach would be to log also zoom levels and active map layers. However, the resulting log can only help, if one has access to the cartographic renderer applied during the study.

3 Gaze Tracking Framework for Interactive Vector-Based Maps

In this section, we propose *FeaturEyeTrack* as framework that applies the idea of Gaze Map Matching (Kiefer and Giannopoulos 2012) to interactive web maps, and which works with a vector model of the cartographic visualization instead of a GIS vector model. First, we discuss the design considerations and derive four criteria our framework must meet. Then we describe the framework in detail and afterwards discuss the results from a preliminary proof of concept.

Figure 3: Example of a map extent. The cyan and magenta circles highlight fixations.



3.1 Design Considerations

The missing capability of previous systems to determine what exactly has been looked at motivates our first design consideration: the framework should be able to map fixations to the *visual* features of a map, i.e., to what is actually shown by the cartographic renderer. This includes, for instance, taking into account labels, the visualization style, or occlusion if features overlap.

This mapping should be done in an automated fashion, thus avoiding extensive AOI annotation (before the study) and manual coding (after the study), as would be necessary by other approaches.

As today web map services are ubiquitous, the proposed framework should work with interactive online maps. Not only does this allow testing user behavior with realistic map interfaces, but it also makes it much easier to test different map styles as well as custom map material.

Furthermore, the suggested framework should log other interaction data, such as mouse movements and clicks, and the current map extent. Raw gaze data and fixations should both be logged to allow in-depth analyses of the eye movements (e.g., testing different fixation algorithms or statistics on saccades).

To summarize, the proposed framework should satisfy the following criteria:

- Platform for user studies on interactive web maps.
- Log of interaction events, in particular zooming, panning, mouse events, raw gaze and fixations.
- Logging of *visual* map features at the position of the fixation.
- Low effort for stimulus preparation and data analysis

According to these criteria, we propose the framework described in the following, which is based on vector tile maps. These are particularly suited because they offer direct access to all visual map features, including labels.

Figure 4: Example feature log. The cyan and magenta highlights correspond to the fixations in Figure 3.

timestamp	ID	name	type	icon	json
22896143	695468191	Little St	road-label-medium	null	{"geometry":{"coordina
24026062	52037611	null	road-secondary-ter	null	{"geometry":{"coordina
24121425	43730830940	the happy frog	poi-scalerank4-115	fast-food	{"geometry":{"coordina
24275622	4429805531	Park Ave	road-label-large	null	{"geometry":{"coordina
24376876	52037611	null	road-secondary-ter	null	{"geometry":{"coordina
24445645	52037611	null	road-secondary-ter	null	{"geometry":{"coordina
24832267	4429805531	Park Ave	road-label-large	null	{"geometry":{"coordina

3.2 The FeaturEyeTrack Framework

Figure 2 shows a schematic overview of the proposed framework FeaturEyeTrack. It consists of three main parts. The *Main Program* is the “brain” of the framework and collects the gaze data stream as well as the mouse input. It routes the interactions towards the second component and writes the log. In our prototype, we used Java for the main program. The second component (*Interactive Map*) is the bridge between the map service and the main program. It consist of a html webpage with JavaScript that connects to the vector tile map service and provides the functionality to request map extents and identify the features at a given location. In our case, we use the Mapbox service¹ because it allows to upload one’s custom map data, provides an open source JavaScript API, and there exists a third party standalone webserver² for hosting the service locally. The third component of FeaturEyeTrack (*Log*) is a SQLite database for storing the log data of the eye tracker, mouse and map.

The main program sets up the connection to the data stream of the mouse and the eye tracker and loads the webpage to display the vector tile map. Its logger class writes raw gaze data and mouse events directly to the database. At the same time, the program calculates fixations and applies the mouse events that manipulate the map extent. Every time this happens, the API of the map service is queried to return information on the map extent and the features in the proximity of the current fixation. The buffer can be adjusted before a study to compensate for the inaccuracy of the eye tracker. In our test run, we used a value of 0.5° of visual angle (see Figure 3). The result is a geoJSON object containing all information about the features within this buffer stored on the server.

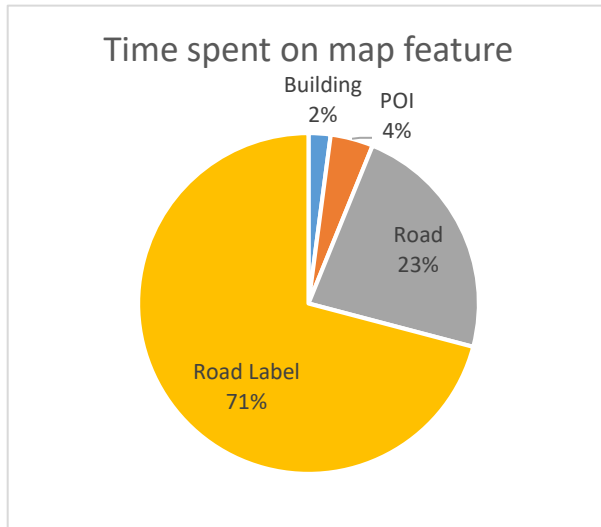
Our framework produces five different tables in the log database. All contain the subject ID, trail number and a timestamp. This triple serves as identifier in corresponding tables (primary key). Additionally the following data are stored:

- **Raw Gaze Log:**
 - X and Y coordinates in pixel (raw gaze at the full recording frequency of the eye tracker, in our case 300 Hz)
- **Fixation Log:**
 - X and Y coordinates in pixel (fixations, as computed by the fixation module, on average approx. 4 per second)
 - Fixation duration in milliseconds,
 - Latitude and longitude

¹ Mapbox Inc, (Accessed: 20.01.2017), <https://www.mapbox.com>.

² Klokantech Technologies GmbH, (Accessed: 20.01.2017) <https://github.com/klokantech/tileserver-gl>.

Figure 5: Pie chart of one participant showing the percentages of fixation duration spent on certain map features during the free exploration.



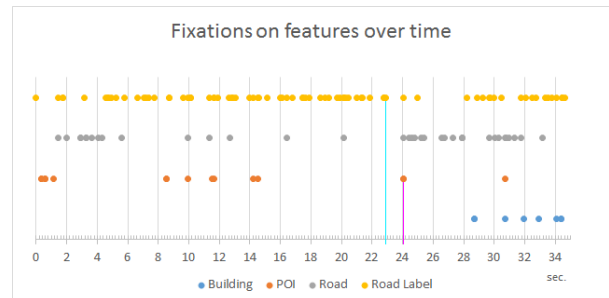
- Map Feature Log:**
 It contains the following data for each feature within the center of a fixation (see Figure 3):
 - Feature ID
 - Name of the feature (e.g. street name)
 - Type of the feature (e.g. building, label)
 - Icon name, if it is a POI with special symbology (e.g. a cup for a cafe)
 - GeoJSON String, with all available information, also the geometry
- Map Extent Log:**
 For every pan/zoom event, an entry with the following data is created:
 - Latitude and longitude of the north-east point of the extent
 - Latitude and longitude of the south-west point of the extent
 - Pitch of the map extent
 - Bearing of the map extent
- Mouse Log:**
 An entry is created, whenever a mouse event is detected. A time threshold can be set for *mouse move* events.
 - X and Y coordinates in pixel
 - Mouse event type (e.g. left click,...)

As we use a SQLite database for the log, one can run traditional SQL queries for further analyses. Furthermore, it allows importing data to a GIS or a spreadsheet software for visual analytics.

3.3 Preliminary Results

We ran a test study as a proof-of-concept of our framework. Figure 3 shows a part of the map extent that was presented to the participant. Her task was to explore the map freely for 35

Figure 6: The chart depicts the time, when a fixation happens on a certain map feature during free exploration. The cyan and magenta highlights depict the fixations from Figure 3.



seconds (free exploration). Two fixations (cyan and magenta circles on the map) in the middle of the trail are highlighted for demonstration purposes. Figure 4 shows the relevant part of our map feature log. As one can see, the cyan fixation was on a street label, whereas at the magenta the participant fixated on the name of a restaurant.

In addition, the data log was imported in Microsoft Excel for further analysis, resulting in the illustrations depicted in Figures 5 and 6. The former illustrates the distribution of fixations over four different feature categories. It becomes apparent that, for instance, in the free exploration task the participant spent 71 % of her overall fixation time on road labels. The latter, Figure 6, illustrates switches of visual attention between feature classes over time. For instance, the attention switch between the road and the restaurant can also be observed here. Notably, it also reveals that only at the end of the trail, the user fixated on buildings.

4 Discussion and Future Work

Logging visual attention on interactive maps with previous approaches has been cumbersome due to the large effort connected with annotating AOIs and the problem of limited reproducibility of the study situation in the analysis phase. Our framework addresses this issue by leveraging vector-based maps and querying the rendered features as soon as a fixation has been detected. To the best of our knowledge, this is the first approach that allows efficient online user logging on interactive maps with automatic Gaze Map Matching. Furthermore, compared to other tools, the proposed framework does not struggle from map projection distortion or synchronization problems between eye tracking data and map interaction logging, as a single tool collects all the data.

FeaturEyeTrack facilitates not only the detection of map features that are currently focused on the user for post-study analyses, but also tracks these data in real time. This is an important requirement for future extensions of the framework in which it could be used for interactive and adequate adaptations of map interfaces (refer to Göbel et al. 2016).

One common challenge with gaze tracking is the inaccuracy of the data due to of various reasons (mainly microsaccades, calibration, and hardware issues). Especially for maps with high information density, this can lead to many false positives, meaning features are detected despite the user not having

focused on them. Future work has to develop intelligent algorithms to filter out these false positives. One idea could be to reason about a sequence of fixations instead of only single ones, similar to the Hidden Markov Model approach described in Kiefer and Giannopoulos (2012).

5 Acknowledgements

The Swiss National Science Foundation has supported this research (grant number 200021_162886).

References

- Andrienko, G., Andrienko, N., Burch, M. and Weiskopf, D. (2012) ‘Visual Analytics Methodology for Eye Movement Studies’, *IEEE Transactions on Visualization and Computer Graphics*, 18(12), pp. 2889–2898.
- Fabrikant, S. I., Rebich-Hespanha, S., Andrienko, N., Andrienko, G. and Montello, D. R. (2008) ‘Novel Method to Measure Inference Affordance in Static Small-Multiple Map Displays Representing Dynamic Processes’, *The Cartographic Journal*, 45(3), pp. 201–215.
- Giannopoulos, I., Kiefer, P. and Raubal, M. (2012) ‘GeoGazemarks: Providing Gaze History for the Orientation on Small Display Maps’, in *Proceedings of the 14th ACM international conference on Multimodal interaction*. New York, New York, USA: ACM Press (ICMI ’12), pp. 165–172.
- Göbel, F., Giannopoulos, I. and Raubal, M. (2016) ‘The importance of visual attention for adaptive interfaces’, in *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. New York, New York, USA: ACM Press (MobileHCI ’16), pp. 930–935.
- Holmqvist, K., Nyström, M., Andersson, R., Richard, D., Jarodzka, H. and Weijer, J. van de (2011) *Eye Tracking: A comprehensive guide to methods and measures*. 1st edn. Oxford: Oxford University Press.
- Kiefer, P. and Giannopoulos, I. (2012) ‘Gaze Map Matching: Mapping Eye Tracking Data to Geographic Vector Features’, in *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*. New York, New York, USA: ACM (SIGSPATIAL ’12), pp. 359–368.
- Kiefer, P., Giannopoulos, I., Raubal, M. and Duchowski, A. (2017) ‘Eye Tracking for Spatial Research: Cognition, Computation, Challenges’, *Spatial Cognition & Computation*, 17(1–2), pp. 1–19.
- Lobben, A. K. (2004) ‘Tasks, Strategies, and Cognitive Processes Associated With Navigational Map Reading: A Review Perspective’, *The Professional Geographer*, 56(2), pp. 270–281.
- Montello, D. R. (2002) ‘Cognitive Map-Design Research in the Twentieth Century: Theoretical and Empirical Approaches’, *Cartography and Geographic Information Science*, 29(3), pp. 283–304.
- Ooms, K., Coltekin, A., De Maeyer, P., Dupont, L., Fabrikant, S., Incoul, A., Kuhn, M., Slabbinck, H., Vansteenkiste, P. and Van der Haegen, L. (2015) ‘Combining user logging with eye tracking for interactive and dynamic applications’, *Behavior Research Methods*, 47(4), pp. 977–993.
- Ooms, K., De Maeyer, P., Fack, V., Van Assche, E. and Witlox, F. (2012) ‘Interpreting maps through the eyes of expert and novice users’, *International Journal of Geographical Information Science*, 26(10), pp. 1773–1788.
- Slocum, T. A., McMaster, R. B., Kessler, F. C. and Howard, H. H. (2009) *Thematic Cartography and Geovisualization*. 3rd edn. Upper Saddle River, New Jersey, USA: Pearson.