Experimenting with polylines on the visualization of eye tracking data from observations of cartographic lines

Sophia Karagiorgou[1], Vassilios Krassanakis[1], Vassilios Vescoukis[2], Byron Nakos[1]

[1] National Technical University of Athens (NTUA).
School of Rural and Surveying Engineering.
[2] Eidgenössische Technische Hochschule Zürich (ETH).
Institute of Cartography and Geoinformation.

Corresponding authors: sokaragi@mail.ntua.gr, vvescoukis@ethz.ch

Abstract. Several visualization methods for eye tracking data exist to help researchers from many disciplines depict data collected in eye tracking experiments. Focusing on eye tracking data from observations of cartographic lines, in this paper we discuss early findings on a new visualization that uses inferred polylines instead of more traditional techniques such as heat maps to visualize eye tracking data. This visualization depicts the average line that is actually seen by subjects, which can be useful in the study of cartographic concepts such as the assessment of the effects of alternative cartographic lines presentations in maps, of distractions, abstraction levels and more.

Keywords. Eye tracking, visualization, cartography.

1 Introduction and related work

Eye tracking is a widely used methodology in many scientific fields, as it reveals important findings about the human cognitive processes during the observation of a visual stimulus. In cartographic research, eye tracking is a valuable tool for the execution of experiments related to the study of map reading and cartographic design evaluation. An important element of eye movement analysis is the visualization of eye tracking data using techniques referred to the gaze behavior of either individuals or all the subjects in an experiment. Considering that the amount of data collected can blur the reference with the visual stimulus, visualization techniques are usually applied after clustering the gaze recordings in fixations and saccades. A typical visualization is the scan path graph, where fixations are depicted as circles with radical values related to their durations, while saccades are presented as connector line segments among fixations. Other techniques include heat maps and scan path graphs, using variables such as duration, number of fixations, participant percentage etc. [1]

In this paper we report early progress on the depiction of the gaze route history using a polyline, which is feasible, as the visual trace is generated from sequential raw eye tracking data [2]. The nodes of such a polyline contain information about the durations of fixations or other statistical values, which can also be attributed to line sections that represent saccadic movements. Generally, the reconstruction of gaze route history can be very useful in the study of several cartographic concepts as a gaze polyline depicts the line that is actually perceived from subjects.

The motivation for this work stems from methods used in the inference of graph geometries such as transportation networks, from GPS tracking data. Several such methods rely on trajectory clustering. Some of the algorithms in the literature [3] [4] operate on point data and do not take the temporal aspect into consideration. Others infer curved paths using K-means clustering of raw tracking data along with distance measures [5]; others transform tracking data to discretized images using Kernel Density Estimation (KDE). They function well for frequently sampled and redundant track data [6], but are sensitive to noise. Other approaches relying on computational geometry techniques [7] operate on tracks of high-resolution and accuracy. The final category involves trace-clustering approaches that derive a connected road network from vehicle trajectories [8] or different movement types. This work applies such a technique in eye tracking data to automatically extract “hubs” and construct a polyline that corresponds to the observed geometry of cartographic lines.

2 Inference of Polylines from eye tracking data

The aim of this work is to derive a single polyline geometry from sampled eye tracking data from multiple users. **Fig. 1** plots data used in our experiment in blue color with the actual cartographic line that the subjects have been asked to follow, shown in black.

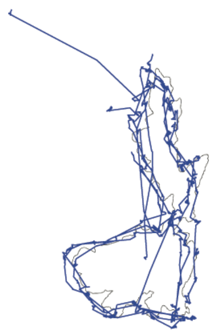

Fig. 1. Eye tracking data example

2.1 A first version of a proposed algorithm

The proposed algorithm to derive the polylines from eye tracking data involves three steps; (i) identifying hubs, (ii) connecting hubs, and (iii) reducing the links into a single geometry, which are discussed in the sequel.

Phase 1: Hubs and Spatial Fixation. A hub represents the spatial fixation that the eye creates near an area of interest. Indicators for hub recognition are the number of tracking samples, the number of different users and the coverage of an extended area of focus. The algorithm takes as input the eye tracking data and determines the k-NNs of each tracking sample, which are subsequently filtered according to the number of users. On these filtered tracking samples, we apply the DBSCAN clustering algorithm using a distance threshold and a minimum number of samples, which depend on the specifics of the experiment. The centroids of the resulting clusters are the hubs. **Fig. 2** shows the hubs derived after applying the hubs inference algorithm in our test dataset.

Phase 2: Connecting Hubs. Next, we connect hubs by links. A fringe benefit of the hubs computation based on spatial fixation is that for all data we know which samples helped in identifying hubs. To derive links we exploit this knowledge: for each hub we record the outgoing and/or incoming tracking portions connecting this hub to others by scanning all eye tracking data to discover sequences of hubs. The result of this step is the creation of a sample polyline set that connects hubs with links. In our representation of eye tracking data, all tracking samples that are also hubs are marked as such. Hence, performing a linear scan of all tracking data reveals the respective tracking portions that connect hubs.

Phase 3: Compacting Links. To this point, we have hubs connected by links derived from eye tracking data that exhibit spatial fixation at these hubs. In a nutshell, the algorithm identifies tracking portions that are close to existing links by means of a buffer region and merges their geometry into the existing link geometry. The size of the buffer region depends on the specifics of the data; in our case we used 15 pixels as buffer region. In this step, we neither introduce new hubs nor do we add new links. We only adjust the geometry of existing links using a three-step algorithm: (i) sort existing link samples, (ii) determine relevant tracking portions using a buffer region around link samples, and (iii) adjust the geometry of links based on the tracking data geometry.

In our experimentation so far we first sort all links according to their length so as to process longer links first as they may be more significant for polyline construction, which remains to be further tested future work. In step (ii) the algorithm uses a buffer region around the examined link sample and retrieves all intersecting portions of other links. New links are created by interpolating link samples and introducing hubs. New links are assigned a weight that is the sum of the weights of the merged links. Link samples are updated several times during this phase. While the examined links are reconstructed, new link samples are created to replace links in previous iterations.

2.2 Polylines Inference Results

The cartographic line that we try to infer consists of 6595 links (edges) and 6607 nodes. The edges have a length of 4041 pixels, as the reference system is in pixels. Sampling of eye tracking data was at 60 Hz (0.017 sec). Data comes from 3 different users with a total length of 89880 pixels (**Fig. 1**). Following the various stages of the polylines inference algorithm, the following output is produced. During the first phase, i.e., hubs extraction and connection, 109 hubs and 300 link samples are generated. The second polylines inference phase, i.e. compacting links, produces 119 hubs, 79 links and a length of 2990 pixels. This result shows that during the second phase of the algorithm, the number of hubs remains largely constant but only the length of the links connecting them is significantly reduced since we radically merge links during this phase. **Fig. 3** visualizes the inferred polylines in blue and the actual cartographic data in grey color.

Fig. 2. Hubs Inference from Spatial Fixation

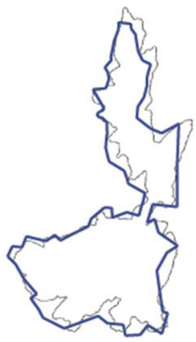

Fig. 3. Inferred Polylines

3 Further work

We briefly presented a polyline-based visualization of eye tracking data that depicts the "average" cartographic line observed by subjects, along with the algorithm that is used to infer this polyline. Clearly, such a visualization is of little use in cases where the context of eye tracking experiments has no lines of some kind that subjects are required to follow. It is, however, quite interesting in cases that such a line really exists, as is the case in cartography where borders, navigation routes and all kinds of curves, are used to represent useful information on a map. Studying the effects of different visualization attributes of cartographic lines in the concentration of the eye's attention to a central linear entity can benefit from using the representation of eye tracking data introduced in this paper.

This visualization can be further improved by adding color attributes to the inferred polyline using calculations such as eye tracking samples data density near the line, or other statistical metrics. Considering that it is the mind that actually does the cognitive interpretation of lines observed, it is rather impossible to infer a polyline that very closely matches the initial cartographic line. However, studying the deviations of individual observers' tracks from the average polyline, and combining the results with semantics from the experiment and subject context may produced some interesting results, too.

Application of the proposed visualization in other kinds of lines whose eye tracking makes sense, as is the case with some medical images, is another area that is definitively worth exploring. Last but not least, the production algorithm of the polyline needs further experimentation on bigger data sets and possibly improvement in few operational aspects. All of the above are future directions of this research.

4 References

1. Bojko, A.: Informative or Misleading? Heatmaps Decostruction. In: Jacko, J.A. (Eds.) Human-Computer Interaction. pp. 30-39, Springer-Verlag, Berlin (2009)
2. Bargiota, T., Mitropoulos, V., Krassanakis, V., Nakos, B.: Measuring locations of critical points along cartographic lines. In: 26th International Cartographic Confer-ence, Dresden (2013)
3. Zhang, T., Ramakrishnan, R., Livny, M.: Birch: An efficient data clustering method for very large databases. In: Proc. of the 1996 ACM SIGMOD Int'l conf. on Management of Data, pp 103-114, Montreal, CA. (1996)
4. Ester, M., Kriegel, H.P., S, J., Xu, X.: A density-based algorithm for discover-ing clusters in large spatial databases with noise. In: Proc. of the 2nd Int'l conf. on Knowledge Discovery and Data Mining, pp 226-231, Portland, OR. (1996)
5. Edelkamp, S. and Schroedl, S.: Route planning and map inference with global positioning traces, In: Computer Science in Perspective, Springer-Verlag, pp 128-151, New York. (2003)
6. Biagioni, J. and Eriksson, J.: Map inference in the face of noise and disparity. In: Proc. of the 20th ACM SIGSPATIAL GIS conf., pp 79-88, Redondo Beach, CA. (2012)
7. Ahmed, M. and Wenk, C.: Constructing street networks from gps trajectories. In: Proc. of the 20th Annual European Symposium on Algorithms, pp 60-71, Ljublja-na, Slovenia. (2012)
8. Karagiorgou, S. and Pfoser, D.: On vehicle tracking data-based road network generation. In: Proc. of the 20th ACM SIGSPATIAL GIS conf., pp 89-98, Redondo Beach, CA. (2012)