The VERP Explorer: A Tool for Exploring Eye Movements of Visual-Cognitive Tasks Using Recurrence Plots

Çağatay Demiralp, Jesse Cirimele, Jeffrey Heer and Stuart K. Card

**Abstract**— Evaluating the effectiveness of the visual design of an interface is an important yet challenging problem. In this paper, we introduce The VERP (Visualization of Eye movements with Recurrence Plots) Explorer, a visual analysis tool for exploring eye movements during visual-cognitive tasks. The VERP Explorer couples conventional visualizations of eye movements with recurrence plots that reveal patterns of revisitation over time. We contribute a set of methods for the analysis of eye movement sequences, including recurrence motifs for identifying behavioral eye movement patterns. We apply the VERP Explorer to the domain of medical checklist design, analyzing eye movements of doctors searching for information in checklists under time pressure. We use these results to introduce the notion of visual micro-foraging, which generalizes information foraging theory to visual design.

**Index Terms**— Visual search, eye-movements, recurrence plots, quantified recurrence, analysis,  sequential behavior, time series, information foraging, theory, visual micro-foraging.

# Related Work

## Eye-tracking Data Visualization

Increased popularity of eye-tracking studies brought anew interest for developing better visual analysis methods for eye-movement data [1]. Most of the methods adapted in practice have been, however, around for sometime [2]. There are several standard techniques for visualizing eye-tracking data, including heat maps, focus maps, and gaze plots (scan paths).

Heat map visualizations are widely used for displaying aggregated patterns of eye movements. However, they suffer from over saturation when the underlying data is dense. Heat maps can also be misleading if the color map isn’t chosen carefully [3]

Related to heat maps are focus maps. A focus map is an image mask that shows the underlying stimulus image at eye-tracking points. The degree of visibility at a particular stimulus region is proportional to the density of the tracking points at that region.

Missing from heat maps and focus maps is a temporal view of the data. Understanding temporal patterns in eye tracking is important as they change not only by the visual stimulus but also by the task [4]. Animated heat maps and gaze plots are widely used to visualize the time ordering in eye tracking data. Animations may, however, cognitively overload viewers trying to grasp the temporal context [5].

Using gaze plots is probably the most common approach to visualizing the temporal behaviour of eye movements.

Understanding differences and similarities in eye movements across subjects is an important goal in eye-tracking studies. However, basic configurations of heat maps and gaze plots suffer from visual clutter when the underlying eye tracking data is dense.

Prior work uses several techniques to reduce visual clutter and support multi-subject comparisons. Raschke *et al.* [6] introduce a parallel scan-path visualization to facilitate the comparison of eye-tracking data across users. Earlier research also proposes the space-time cube visualization for eye movements [7,8]. Originally introduced for geographic movement analysis [9,10], the space-time cube visualization treats time as the third—spatial—dimension, enabling a static visualization of multiple eye-movement trajectories in 3D.

Reducing visual complexity often requires aggregating and sampling the data, while introducing simpler abstract representations without losing the original context in the data. Burch *et al.* introduce saccade plots that combine a heat map and a graph-based matrix representation for aggregated movement directions [11]. Experts try to capture semantics of eye movements by tagging areas of interest (AOIs) on the stimulus image and associating them with fixations. This also reduces the data complexity as the experts care more about what the viewers look at then where they look at. Prior research borrows from text visualization techniques to visualize AOIs. Tsang *et al.* [12]applies the WordTree visualization [13] to AOI tags concatenated based on their in eye-movement trajectories. Similar to ThemeRiver [14], AOI Rivers [15] visualizes fixation frequencies of AOIs as flow maps. While the spatial context is accessible only indirectly, the flow map visualization reduces the visual clutter that would be otherwise caused by use of gaze plots.

Recurrence plots are a type of non-linear analysis that has been used in the study of dynamical systems and other areas (Eichmann et al., 1978; Marwan, 2008). Recently it has been applied to eye-movements (Anderson et al, 2013). Recent research applies recurrence plots to eye-movements (Anderson et al, 2013).

The VERP Explorer couples several of the standard eye-tracking visualizations above, including heat maps, focus maps, gaze plots, with alpha patches and recurrence plots through interaction and quantification.

# Design of The VERP Explorer

Our primary goal for developing the VERP Explorer was to provide an interactive tool for applying our recurrence-based analysis. To this end, we integrate several existing and new visualizations into our tool.

We identified several high-level analysis questions that we would like the VERP Explorer to facilitate.

1. Separate fixations and saccades.
2. Identify and quantify visual search and consumption behaviours
3. Determine re-visitation patterns of eye movements

We now discuss the design of visual representations and interaction techniques in the VERP Explorer to support these tasks.

## Overview

Figure 1 shows a screenshot from our tool. Two main views are the scene view and the recurrence view. Both views are coordinated through brushing and linking.

## Visual Encoding

We use five visual representations for eye movement data: scatter plots, heat maps, focus maps, scan paths, alpha patches and recurrence plots. Scatter plots, heat maps, and focus maps enable to visualize different aspects of spatial aggregate eye-movement behaviour while

### Heat Maps, Focus Maps, and Scatter Plots of Eye Movements

The VERP Explorer enables users to visualize eye-tracking positions as heat maps, focus maps, and scatter plots. While all the three methods primarily encode eye-movement positions and are typically overlaid on the stimulus scene, they have complementary strengths.

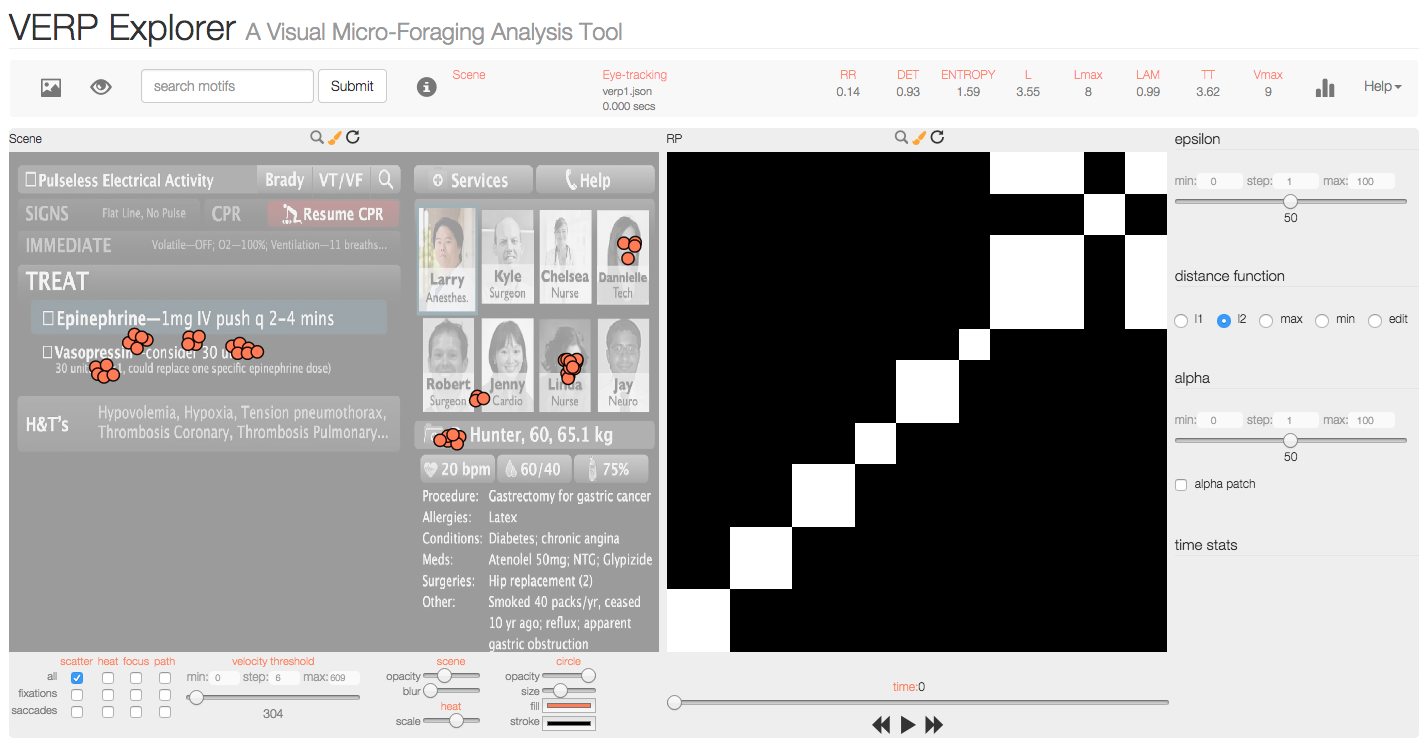


Figure The VERP Explorer interface.

Heat maps and focus maps are two related standard visualization techniques, which are useful for providing a synaptic view of eye movements aggregated over time and subjects. The VERP Explorer creates the heat map visualizations by drawing semi-opaque disks centred at eye-tracking positions. The disks are filled with a color gradient and their opacity is modulated (decreased) with the distance from the disk center (Figure 2)

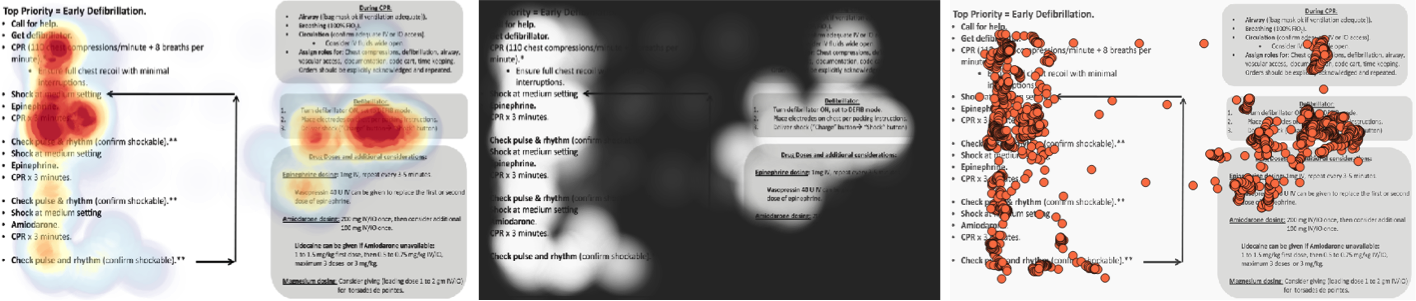


Figure Three spatial eye-tracking visualizations from the VERP Explorer: Heat map (left), focus map (middle), and scatter plot (right).

By painting eye-movement point densities, heat maps obscure, however, the areas of attention when overlaid on the stimulus image. Focus maps visually “invert” heat maps to enable the visibility of the areas of viewer attention. To create a focus map, we first create a uniform image (mask) that has the same size as the underlying stimulus image. We then vary the opacity at each pixel inversely proportional to the opacity of the corresponding heat map pixel. Focus maps are essentially negative space representations, visualizing the negative space of the corresponding heat maps (Figure 2).

Heat maps and focus maps visualize eye movements indirectly, facilitating visual aggregation. On the other hand, scatter plots provide a discrete view by representing eye-movement positions directly. The VERP Explorer creates scatter plot views by directly drawing each eye tracking position as a circular node in the plane (Figure 2)

Scatter plots are useful for seeing patterns and outliers in eye-movements, while enabling the inspection of individual eye-movement positions. We also use the scatter plot view for the timeline animation (Section), as it provides a direct, discrete representation of the tracking positions.

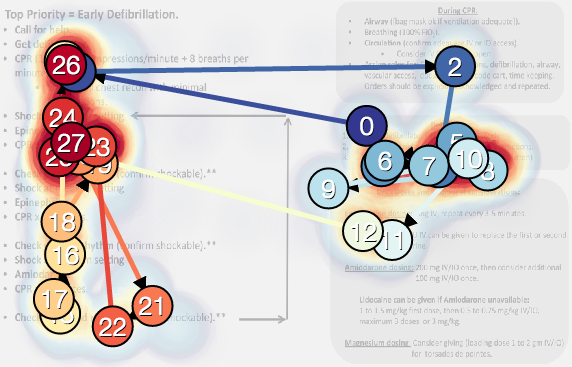


Figure . Scan path.

### Scan Paths

In their basic, static configuration, neither heat maps nor focus maps convey the temporal order of eye movements. The VERP Explorer uses scan paths (gaze plots) to provide an aggregate temporal view of eye movements. It creates scan path views by drawing circles centred at the centroids of fixation clusters and connecting two consecutive clusters with arrows. The VERP Explorer numbers the nodes sequentially. It also encodes the temporal order of fixations by coloring the nodes and the arrows using a color map ranging from dark blue to red [16] (Figure 3)

We generate the fixation clusters using the velocity-threshold fixation (I-VT) algorithm [17]. I-VT is a fast and robust algorithm for identifying fixations and saccades based on their point-to-point velocities. I-VT operates under the assumption that low velocity eye movements correspond for fixations while high velocities for saccade. We now describe the algorithm briefly. See [17] for a detailed, comparative discussion of fixation identification algorithms.

Using I-VT, we compute clusters of fixations in three steps. We first calculate point-to-point velocities for each tracking point. Note that velocities can be computed using spatial or angular distance between consecutive points. We use angular velocities if the head position is provided in the tracking data.

We then classify each point as a fixation or saccade using a velocity threshold. If the point’s velocity is below the threshold, it becomes a fixation point, otherwise it is considered a saccade points. The VERP Explorer lets users interactively modify the velocity threshold using a sliding bar.

In the final step, we gather consecutive fixation points into identical clusters and compute associated measures such as the cluster centroid and duration. We set the minimum cluster size to twenty. Clusters with fewer than twenty fixation points are discarded.

### Saccade Paths

Saccades are the rapid eye movements between fixations and represent the search behaviour of the eye. The VERP Explores creates saccade paths by drawing circular nodes at saccade position and connecting consecutive saccades with arrows. Similar to scan paths, colors of the nodes and directed edges encode the temporal order of the points.



Figure . Alpha patches with increasing alpha values from left to right.

### Alpha Patches

Visual clutter is often a concern in eye tracking data visualization. We introduce alpha patches, alpha-shapes of fixation points, to provide a cleaner view of underlying eye-movement locations. Alpha patches enable users to visualize fixated regions as filled polygonal patches providing visually cleaner representation. Figure 4 shows alpha shapes of a point set with varying alpha values.

The alpha-shape is a generalization of the convex hull of a point set. The primary advantage of alpha shapes over the convex hull is that alpha shapes can recover disconnected, non-convex spaces with holes. Specifically, for a given real parameter ), alpha balls are balls of radius α centered at the points in P. The alpha shape of P is then the union of the convex hulls of the points whose alpha balls intersect. The VERP Explorer enables users to automatically create alpha shapes of eye-movement positions, which we call alpha patches. We now discuss our algorithm for deriving alpha patches.

Given a eye tracking point set (e.g., fixations) and an alpha value, we generate the alpha shape for the point set in three steps. First, we create the Delaunay triangulation of the set. Note that the boundary of the Delaunay triangulation is the convex hull of the points in the set. Second, we extract, from the Delaunay triangulation, the triangles whose vertices are within the alpha distance. The union of the extracted triangles is known as the alpha complex of the point set. In the final step, we determine the boundary of the alpha complex and draw them as simple closed polygons.

In our implementation, we create the Delaunay triangulation once and extract alpha complexes for varying—user determined—alpha values as needed.

### Recurrence Plots

## Interaction Techniques

### Viewing Interactions: Zooming & Panning

The VERP Explorer provides zooming and panning interactions on all of the visualizations that it generates. Both zooming and panning are forms of dynamic visual filtering and essential for exploring dense eye-movement datasets.

### Brushing & Linking

We use brushing & linking in the VERP Explorer to coordinate the scatter plot view of the eye-tracking data with the recurrence plot view. This is the main mechanism that allows users to inspect recurrence space and spatial eye movements simultaneously. Brushing over a location on the scene highlights all the corresponding entries in the recurrence view. Conversely, brushing on the recurrence plot highlights corresponding eye movement positions represented as circular scatter plot nodes. Brushing regions can be resized or moved using mouse as well as keyboard.

### Dynamic Filtering

The ability to interactively aggregate, sample and filter data is key to exploring and untangling complex datasets. The VERP Explorer enables to users dynamically change the visualization and analysis parameters.

Epsilon filtering enables users to explore custom ranges of epsilon values for recurrence plots. These changes are also reflected in measures calculated. Users can also select different distance measures. We provide the Euclidean (L2 Norm), the city block (L1 Norm), the maximum (Linfinty Norm) and the minimum of the absolute differences along data dimensions and the edit distance.

Similar to epsilon filtering, alpha filtering enables to change how the alpha parameter of the alpha shapes.

The VERP Explorer also allows users to change the threshold for fixation-saccade classification dynamically. This is particularly useful when angular velocity calculations are not possible or reliable.

### Timeline Animation

While the scan path visualization provides an aggregate temporal view of the eye movement, it is desirable to be able to directly examine the timeline of the complete data. The VERP Explorer enables users to animate the appearance of eye tracking points in the scatterplot view. Users can set the speed of the animation or manually control it by dragging the animation slider’s handle.

### Motif Search

Recurrence plots facilitate pattern-based analysis of time varying data. One of the motivations of the current work is to relate behavioural eye-movement patterns to visual design through recurrence patterns. The VERP Explorer enables users to search for predefined patterns in the recurrence plot.

## RQA Measures

One of the advantages of recurrence plots is that they enable the quantitative analysis of the underlying data based on the visualization itself. The VERP Explorer provides several of these measures.

Recurrence Rate (RR):

Determinism (DET):

Entropy (ENTR): Shannon entropy based on diagonal line histograms.

Average Diagonal Line Length (L):

Maximum Diagonal Line Length (Lmax):

Laminar Phases (LAM): Analogous to DET but uses vertical line lengths)

Trapping Time (TT): Trapping Time is the average length of the vertical recurrence structures:

Maximum Diagonal Line Length (Vmax):

## Implementation

The VERP Explorer is a web-based application, available online at http://hci.stanford.edu/~cagatay/projects/verp/. It currently accepts eye tracking input in two file formats: ASCII IDF, as exported by iViewX, and JSON. We implemented the VERP Explorer in Javascript using D3 [18] and AngularJS [19] libraries. The source code of the VERP Explorer along with example datasets are available at https://www.github.com/uwdata/verp/.

Acknowledgments

We thank members of Interactive Data Lab at the University of Washington for their feedback on earlier drafts of this paper. We also thank Scott Klemmer for sharing the eye tracking data used.

References

1. T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl. State-of-the-art of visualization for eye tracking data. In EuroVis, 2014
2. K. Holmqvist, M. Nystrom, R. Andersson, R. Dewhurst, H. Jarodzka, and J. Van de Weijer. *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford University Press, 2011.
3. Borland, D., Taylor II, R.M.: Rainbow color map (still) considered harmful. IEEE Computer Graphics and Applications 27 (2007) 14–17
4. Yarbus, A. L. (1967), Eye Movements and Vision, New York: Plenum. (Originally published in Russian 1962)
5. Tversky, Barbara, Julie B. Morrison, and Mireille Bétrancourt. 2002. “Animation: Can it facilitate?” International Journal of Human-Computer Studies 57: 247–262.
6. RASCHKE M., CHEN X., ERTL T.: Parallel scan-path visualization. In *Proceedings of the 2012 Symposium on Eye Tracking Research & Applications* (2012), ETRA’12, pp. 165–168.
7. X. Li, A. Çöltekin, M.-J. Kraak. Visual exploration of eye movement data using the space-time cube. In Proceedings of GIScience 2010, pp. 295-309, Springer, 2010.
8. KURZHALS K., WEISKOPF D.: Space-time visual analytics of eye-tracking data for dynamic stimuli. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2129 – 2138.
9. Hägerstrand, T. (1970) "What about people in Regional Science?" Papers in Regional Science. 24, 1, pp. 6-21.
10. M.-J. Kraak. The space-time cube revisited from a geovisualization perspective. In Proceedings of the 21st International Cartographic Conference, Durban, South-Africa, pp. 1988-1995, Aug. 2003
11. BURCH M., SCHMAUDER H., RASCHKE M., WEISKOPF D.: Saccade plots. In Proceedings of the 2014 Symposium on Eye Tracking Research & Applications (2014), ETRA’14, pp. 307–310.
12. H.Y.Tsang, M.Tory, and C.Swindells. eSeeTrack–visualizingsequen- tial fixation patterns. *IEEE Transactions on Visualization and Computer Graphics 16*, 6(6):953–962, 2010.
13. M. Wattenberg and F.B. Viégas. “The Word Tree, an Interactive Visual Concordance”, IEEE Trans. Visualization and Computer Graphics, 14(6): 1221-1228, Nov.-Dec. 2008.
14. Havre S. Hetzler B. Nowell L. (2000) ThemeRiver: Visualizing Theme Changes over Time.*Proceedings of the IEEE Symposium on Information Visualization.*
15. BURCH M., SCHMAUDER H., RASCHKE M., WEISKOPF D.: Saccade plots. In *Proceedings of the 2014 Sympo- sium on Eye Tracking Research & Applications* (2014), ETRA’14
16. M. Harrower and C. Brewer. ColorBrewer.org: an online tool for selecting colour schemes for maps. *The Cartographic Journal*, 40(1):27–37, 2003.
17. D. D. Salvucci and J. H. Goldberg. Identifying fixations and saccades in eye-tracking protocols. In Proceedings of the Symposium on Eye Tracking Research and Applications, pages 71–78, 2000.
18. M. Bostock, V. Ogievetsky, and J. Heer. D3 : Data-driven documents. IEEE TVCG (Proc. InfoVis), 17(12):2301–2309, 2011.
19. AngularJS, https://angularjs.org