EyeTrackerJS

Final Project for CS582: Data Visualizations

Basic Info

Github Link: https://github.com/cjchagnon/EyeTrackerJS

Chris Chagnon	@cjchagnon	cjchagnon@wpi.edu
Jeffrey Estrada	@jestradaramos	jestrada@wpi.edu
Tiffany Vo	@ttranvo	ttvo@wpi.edu

Background and Motivation

In the User Experience Community Eye-tracking has become a ubiquitous and well utilized commodity for tracking user interactions with a website. It tells a story of what the user does from beginning to end and defines trends on what the user looks for in a website. Unfortunately consuming, searching, querying, and sharing this data is difficult and, often times, restricted to proprietary (and expensive) software platforms.

The WPI User Experience and Decision Making (UXDM) Laboratory, in the Foisie School of Business extensively uses eye tracking when running studies. Through his role in the UXDM laboratory, Chris has access to the eye trackers and the eyetracking data, and has encountered these limitations first hand.

The main motivation for this project is to explore innovative and novel ways of interacting with eye tracking data on the web, to expose its value to a wider market. It will look at ways to recreate the story from the given data and perform analysis for users to come up with their own hypothesis from the data. It will also be important to study the feasibility of using the web and if there are limitations in performance or scalability when working with these data sets.

Project Objectives

The main research question we are asking is can eyetracking data be rendered in easier to use and more accessible formats on the web. This includes other components such as shareability, and interaction design.

Objective	Benefit
To Create a tool / platform that allows for upload and interaction with eye-tracking data in an easy to use (and share) manner.	Wider-access to eye tracking data outside of proprietary software and formats.
To allow for customization of views and renderings.	There is a lot to be gained by allowing the users to customize their views, apply filters, and change rendering methods. In doing so, new information may be able to be derived.

Data

The data will be obtained using Tobii eye trackers in the WPI UXDM lab for various tasks or recordings to get multiple samples to work with. The data itself includes time/index data as well as x, y, and z positional data and other information about the eye and subject. The main concern with this data is the volume. Some of the UXDM eye trackers read ad 300hz which means that they are gathering extensive data.

Data Processing

While the UXDM lab does have multiple data processing filters and options that could be run to clean up the data. Most data cleanup will be done on problem areas such as blank data gaps (if the user looks away, or blinks etc). It is possible that we could attempt to implement these into the visualization engine, but it can be easier to do this manually.

The code itself will need to process the data to prepare it for the charts, such as gathering averages and interpolating for the heat maps and gaze plots. User inputs such as filters, time ranges, and customized views will determine how the data is processed. It will be an important research topic to explore the effects of resolution on performance in rendering these.

Must-Have Features

- Uploading of a CSV of eye tracking data for viewing on the tool.
- Viewing of eye tracking data in a ‘player’ format. As a heatmap, a gaze plot, and a eye position playback.
- Uploading scene information. For our player this will most likely be limited to static scenes such as images and wireframe mockups, and *not* video, click through and site data.
- Timing controls. Scrubbing through a playback, but also changing or flattening the period (Ex viewing a heatmap all at once, vs over time).

Optional Features

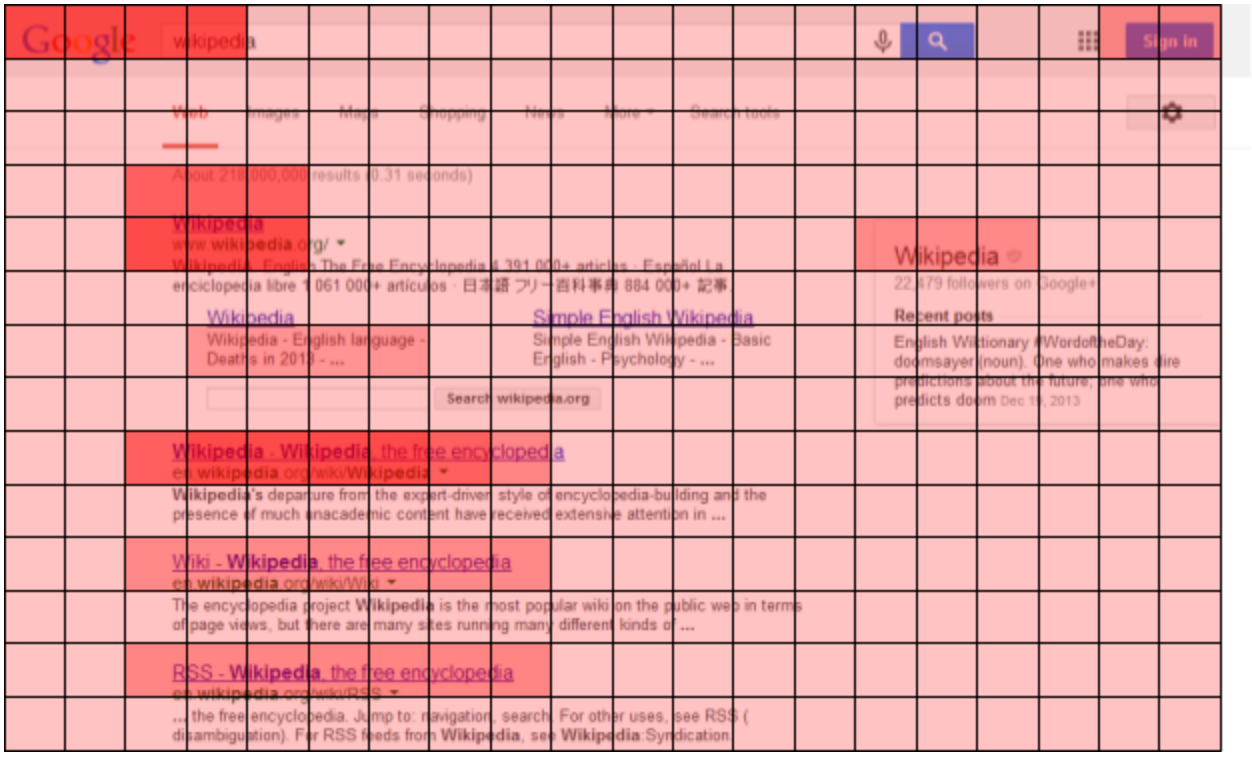
- Sharing & Custom URLs (EX: once uploaded site.domain/vis/1293219gd)
- Exporting (rendering) images of vis for saving or printing.
- Are of interest creation (track intersections in a given area).
- Multiple session tracking, EX combine data from multiple users for comparison
- Automatic collection of some data: EX uploaded scene picture is 500x300px so fields are auto filled.
- Resolution and customization options for heat maps and gaze plots
    - Heat map: Change how large an area is for the heatmap grid
    - Gaze plot: change how long something must gaze for it to count.

Project Schedule

Week 1: 2/12-2/18	Project Planning and Design 02/15 - Proposal Due
Week 2: 2/19-2/25	Build 02/23 - Prototype Presentation Due
Week 3: 2/26-3/02	Present 03/02 - Final Project and Video Due

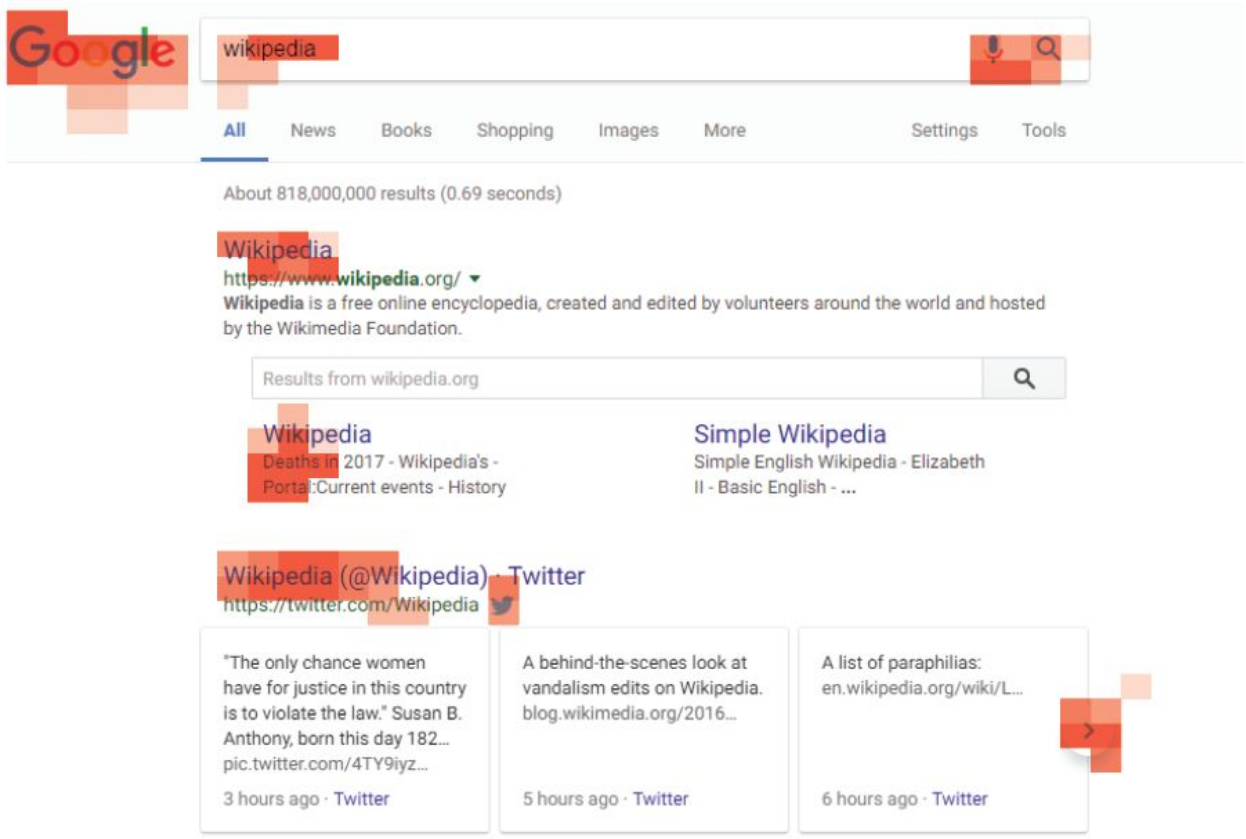
Sketches

Sketch 1: Low Resolution Heatmap



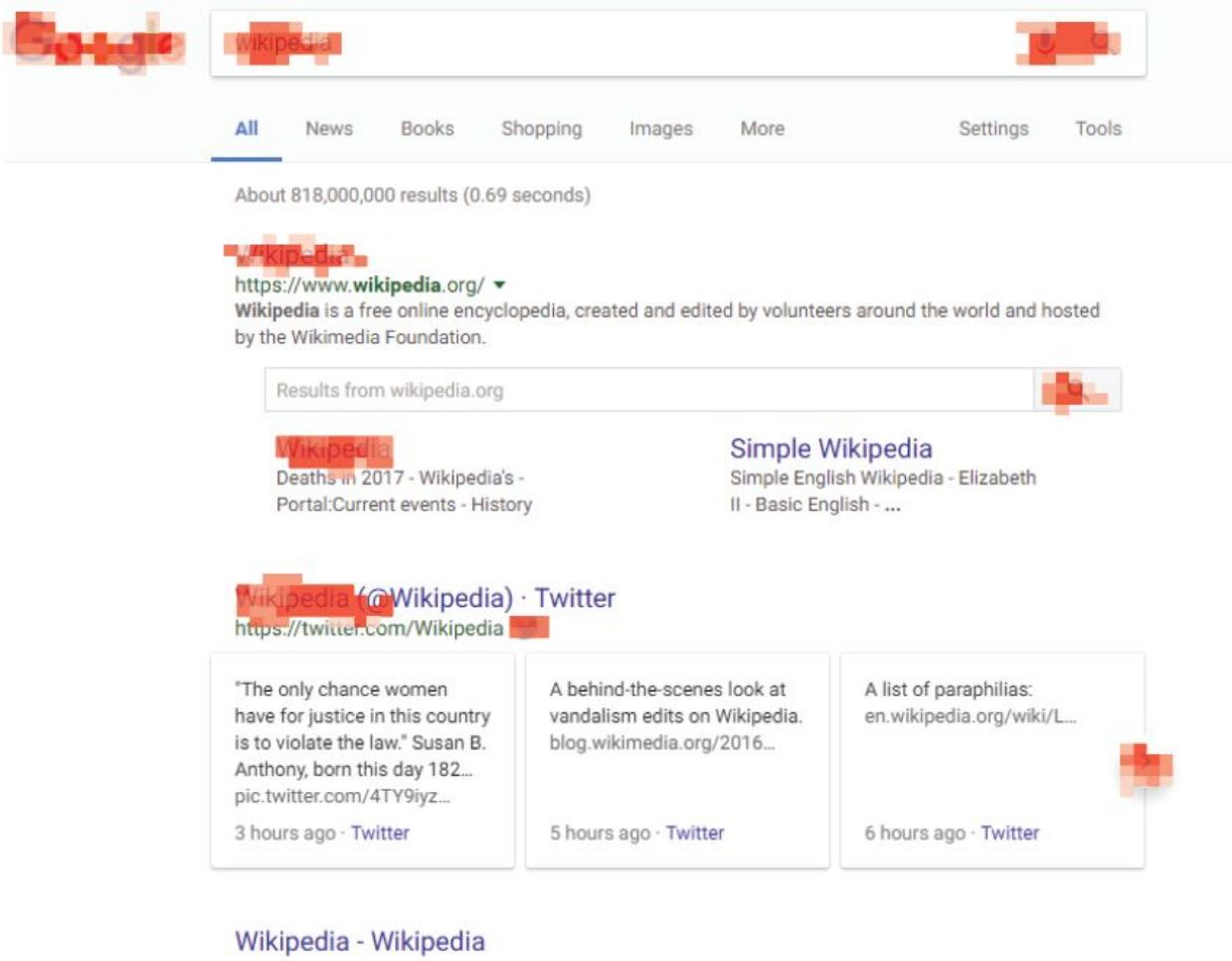
Heatmaps are a way of representing data through summarizing values. They are commonly seen as higher resolution representations using rainbow coloration. This diagram shows a lower resolution abstraction showing focused areas.

Sketch 2: Slightly Higher Resolution Heatmap



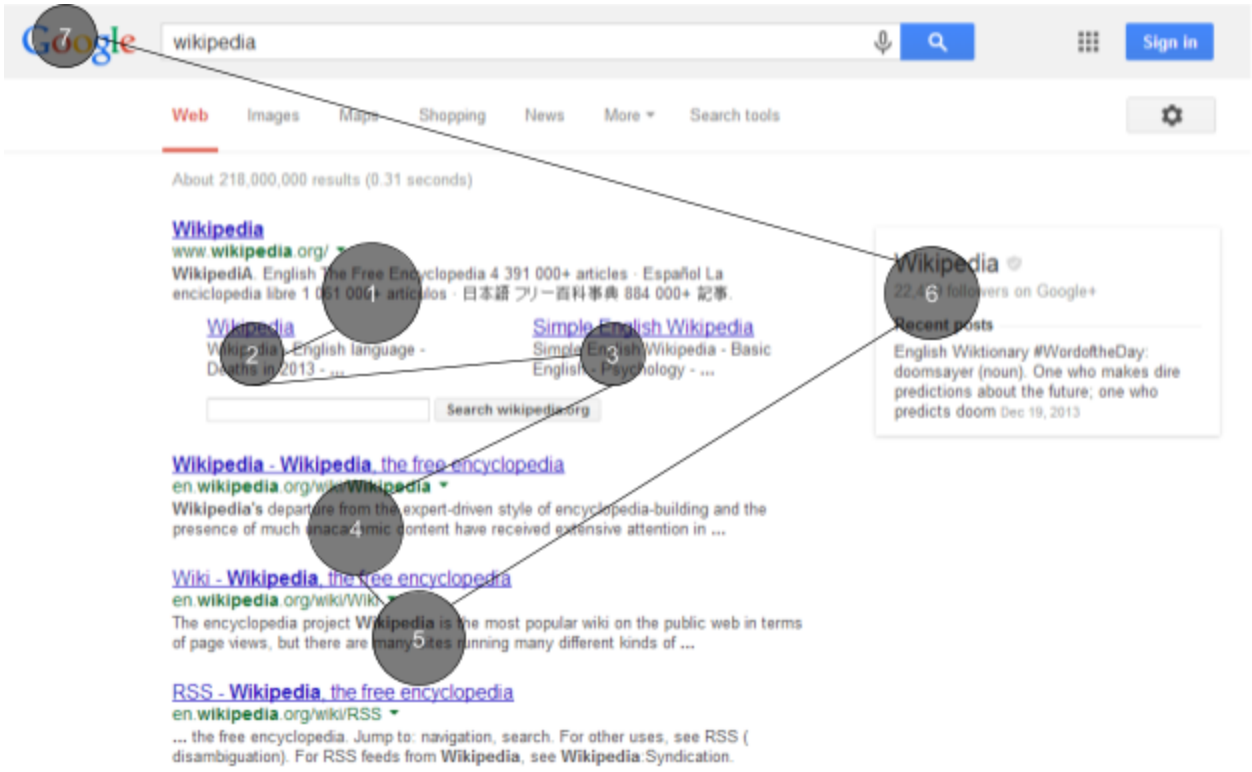
This heatmap is a slightly higher resolution grid (50x50) and you can more clearly see the mapping of focus. This requires more elements to be drawn to the DOM which could cause performance issues

Sketch 3: Higher Resolution Heatmap



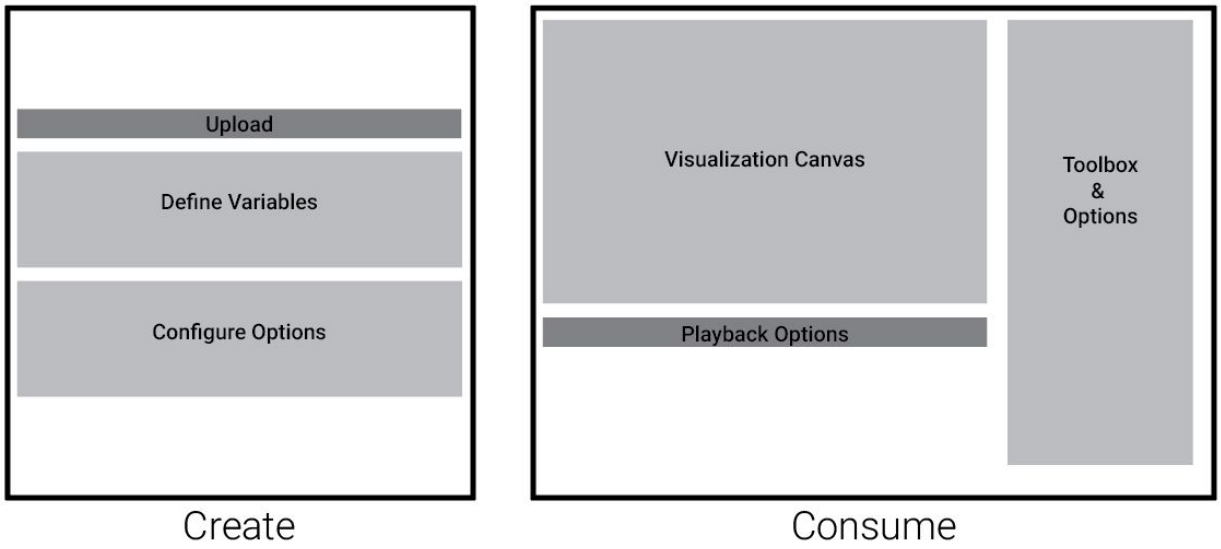
This map features a 100x100 grid with a continues hue but frequency mapped to luminance. The higher resolution grid allows for clarity on elements and exactly which parts drew focus.

Sketch 4: Gaze Plot



This plot features the path a user gaze traverses. Length of time at an area is mapped to bubble size, and the links are representative of the order and path user goes through. Numbers help clarify ordering

Sketch 5: Wireframing



There are two main parts of the site, creating (uploading) and then viewing/consuming that data