User Gaze Predictions in Web Browsers from Mouse Movement Events

Joshua Toenyes[1], Samuel Vange[2], and Saif Chaudhry[3]

I. Abstract

In this assignment we attempted to predict where a user is looking on the screen (their "gaze" position) based only on their mouse movements. A large dataset consisting of mouse movements and corresponding gaze positions was analyzed and a predictive model was developed. We show that the user's gaze coordinates can be fairly well-predicted even by a naive model, and in some portions of the screen, far better by the model we developed.

II. Introduction

The goal of our predictive project was to develop a model that could predict where the user was looking based only on their mouse movements, in actual *x* and *y* coordinates. To approach this problem, we decided to break the problem in to two separate predictive tasks that could be attacked separately. So instead of directly predicting the cartesian screen-coordinates, we decided to essentially predict polar coordinates based on the data available. As one sub-problem we would predict radius and another the direction (see Figure 1).

III. Related Works

To our knowledge, there has not been a significant amount of published research directly predicting user gaze coordinates from only mouse movements. However, there have been several studies in this general area of interest.

As early as 2001, Chen, Anderson and Sohn showed that mouse movements and gaze coordinates are highly correlated in the web browser [2]. This particular paper is cited quite often by data analytics firms when describing the merits of recording mouse movements as part of web site analytics products.

*This work was not supported by any organization

[1]Joshua Toenyes is a Student in Computer Science, University of California: San Diego

[2]Samuel Vange is a Student in Computer Science, University of California: San Diego

[3]Saif Chaudhry is a Student in Computer Science, University of California: San Diego

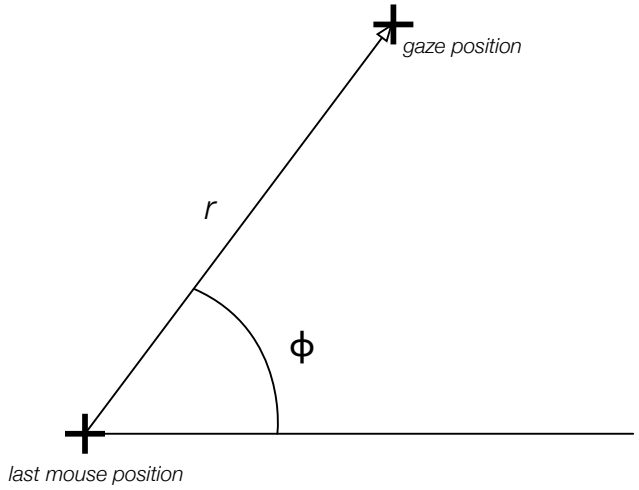

Fig. 1. Generally, we are predicting radius and direction.

Later, Guo and Agichtein showed that it is possible to predict when the gaze is within 100 pixels of the mouse [1]. While Guo and Agichtein's approach is impressive (it can predict with 77% accuracy when the gaze is within 100 pixels of the mouse) their model required a great deal of user-specific information. For example, their predictive model was customized on a per-user basis, and predictions were time sensitive (i.e. "trial-time elapsed").

We endeavored to develop a more generalized approach. One that used only mouse data and was indifferent to the user being predicted, or the specific environment in-which the predictions were being made.

IV. Dataset Description

The dataset consists of time-sequenced gaze coordinates (*where* the user is looking on the screen) and mouse coordinates (*where* the user's mouse is positioned on the screen). Both items are measured in pixels and timestamped to the millisecond. The dataset consists of approximately 255,000 mouse-coordinate data points and 3,000,000 gaze-coordinate data points collected from 17 individuals. In total, the dataset

represents approximately 8.5 hours of user interaction with a web browser.

The dataset was collected by Joshua Toenyes as part of a two-quarter independent-study research project at UCSD under the guidance of Professor Nadir Weibel. The goal of the project was to construct a generic framework for bringing eye-interactivity to the web environment. The dataset for this assignment is a subset of the data collected for this research trial.

V. Data Analysis

Analyzing our data, we sought to discover features which could provide a predictive link between where the user is currently looking on the screen and where the mouse is positioned (or had previously been positioned). Unfortunately, much of the data collected does not include mouse usage in close temporal proximity to recorded gaze coordinates. This is because much of the data was collected while the user was using *only their eyes* to control the web browser, so no mouse movements were recorded.

We made the decision to limit ourselves to using a subset of the data which consisted of only gaze and mouse movement samples that occurred very-near the same time. This was motivated by the assumption that only mouse-movements close in time to gaze measurements are related. Also, data samples could occur at irregular intervals, i.e. mouse coordinates and eye coordinates are not record at fixed, measurable intervals. Mouse coordinates are only recorded when the mouse is moved, so each datum represents the result of a very small *mouse-movement* (1 or 2 pixels in some direction), *not* the position of the mouse at some point in time. As a result of distilling our dataset to only gaze and mouse data points within close temporal proximity, the number of usable samples was reduced.

For our predictive task (described in a following section) we decided to consider the previous **10** mouse coordinate samples that occurred within 1000 milliseconds of a recorded gaze measurement. Gaze measurements that had fewer than 10 mouse coordinate samples within the last 1000 milliseconds were discarded and mouse samples which did not occur within 1000 milliseconds of a gaze sample, or that occurred within 1000 milliseconds but were not one of the ten most-recent, were also discarded.

Analyzing the data from this perspective allowed us to view gaze sample points and their associated previous 10 mouse samples (within the last second) as a single entry in a reformatted data set (see Figure 2). In this configuration our dataset contained 374,257 records of gaze coordinates and their associated ten preceding mouse coordinates.

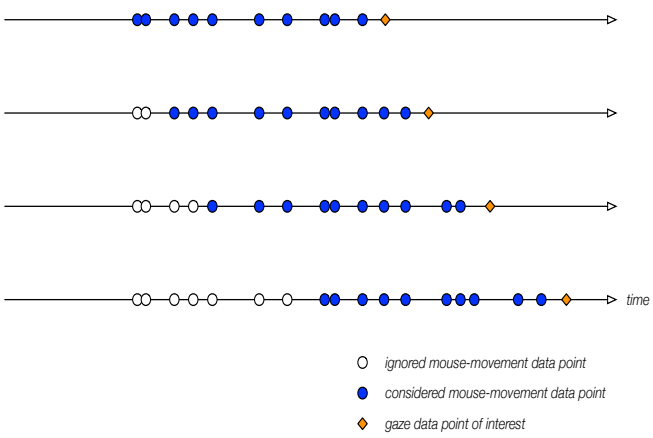

Fig. 2. Mouse and gaze coordinate samples over time.

VI. Predictive Tasks

We endeavored to predict two things from the dataset. First, the gaze position radius, defined as the distance in any direction from the gaze position to the most recent mouse position). Second, the direction to the gaze position from the most recent mouse position. Using these two predictions our ultimate goal was to accurately predict the x and y coordinates of the user's gaze. It is important to note, that the radius prediction could equivalently be interpreted as the euclidean distance from the most recent mouse position to the user's gaze.

A. Prediction Baseline

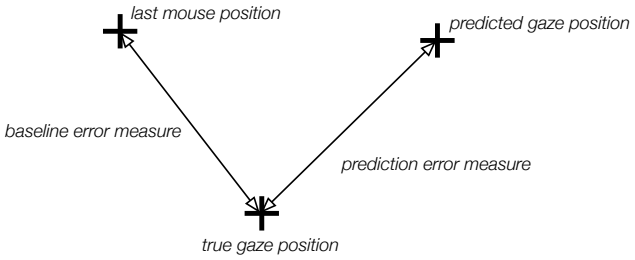

Fig. 3. Baseline comparison for predicting actual coordinates.

For quality comparison of our predictions we will use as a baseline the immediately preceding mouse-coordinate as the prediction of the user's gaze. We will compare our prediction's Mean Absolute Error (MAE)

with that of the baseline model's to evaluate the quality of our predictions. We feel this is a good baseline, as eye and mouse movements have previously been show to be well correlated [2]. Essentially, this will compare the euclidean distance from the predicted *x* and *y* coordinates and the actual *x* and *y* coordinates.

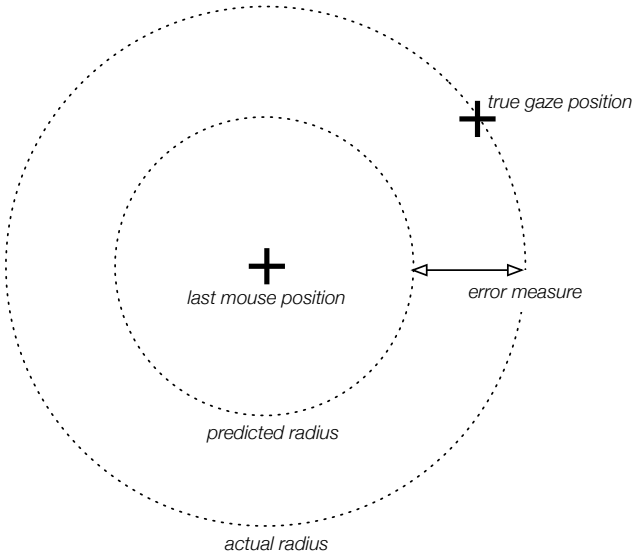

Fig. 4. Error measure for radius prediction

For the radius prediction baseline we will simply always predict the radius to be 100 pixels. That is, we will always predict that user's gaze is 100 pixels from their most recent mouse position. The MAE will also be used to compare the baseline and predicted radius.

B. Dataset Conditioning and Partitioning

Since our dataset was time-sequence and thus biased by the particular user, how long they had been using the machine and the task they were performing, as well as the adjacent data points, we randomly shuffled our dataset to remove these biases. We then partitioned the set into three separate subsets: a *test set* (20%), *validation set* (10%) and *training set* (70%).

C. General Approach

We decided to use two separate linear regressions: one to predict the radius of the gaze position and a second to predict the direction. We used the same features for both linear regressions and changed the trained truth values from radius to direction. We partitioned the screen real estate into some number of buckets, using our validation set to determine the optimum size (in pixels) of each buckets.

D. Data Features

The features we chose to use for our predictors were as follows (each feature was calculated for all ten mouse data points in a record from the dataset). The first feature is a binary array with a **1** in some position corresponding to the *x bucket* in-which this sample falls. This array is the number of *x* buckets minus one in length, so-that if the mouse position falls in to the first bucket no **1** is recorded to avoid overfitting. For example, if the x-coordinate of some mouse sample is 121 pixels and the bucket size in the x-direction is 30, then a **1** will be recorded in the third entry of the array. The second feature is an array exactly as described previously, but corresponding to the *y bucket* in-which some mouse sample falls. Third, the numerical *x-direction gradient* of all mouse samples. Fourth, the numerical *y-direction gradient* of all mouse samples. Finally, the second order *x gradient* and *y gradients* for all mouse samples.

We can list our features as:

$$B_x := \text{x-bucket membership vector} \quad (1)$$
$$B_y := \text{y-bucket membership vector} \quad (2)$$
$$\nabla M_x := \text{gradient in x direction} \quad (3)$$
$$\nabla M_y := \text{gradient in y direction} \quad (4)$$
$$\nabla M_x^2 := \text{2nd gradient in x direction} \quad (5)$$
$$\nabla M_y^2 := \text{2nd gradient in y direction} \quad (6)$$
$$(7)$$

The radius is predicted using the trained value of θ_r as:

$$r = \theta_r[B_x \ B_y \ \nabla M_x \ \nabla M_y \ \nabla M_x^2 \ \nabla M_y^2] \quad (8)$$

Similarly, the direction is predicted using the trained value of θ_ϕ as:

$$\phi = \theta_\phi[B_x \ B_y \ \nabla M_x \ \nabla M_y \ \nabla M_x^2 \ \nabla M_y^2] \quad (9)$$

While our model does not explicitly take in to account the mouse's velocity and acceleration, we believe the gradient to be an analogous feature. Future work could explicitly use velocity and acceleration using datum timestamps, which are available in the dataset.

E. Parameter Tuning

Using our validation set, we determined that our approximately optimum bucket size was 35 by 80 pixels. The smaller the buckets, the more fine-grained our prediction could become. We believe that with more data the bucket size could be decreased, and hence the quality of our predictions could increase. Tuning was accomplished by retraining and evaluating the model (using the validation set) on a very large number of bucket shapes and sizes. This task was computationally intensive and took quite some time accomplish.

VII. Evaluation and Results

Using the previously described predictive approach and features, our radius-predictor was able to predict the radius of the gaze (from the most recent mouse coordinates) with a mean absolute error (MAE) of 120.6 pixels. Meaning, on average the actual radius was within 120.6 pixels of the predicted value. The baseline radius MAE was 128 pixels, so our prediction slightly outperformed the baseline.

When plotting our radius predictions as a heatmap in Figure 5 (each square represents a "bucket"), we notice that we have much smaller predicted radii when the mouse is near the top middle and top left portions of the screen. We speculate that these portions of the screen are likely what the user is interacting with the most, therefore their gaze and mouse coordinates are near each other and the predicted radius is smaller. In contrast, the predicted radius is quiet large when the mouse is near the far right portion of the screen. This is possibly because the user is likely not interacting with that region of the screen, since there are seldom controls or navigation elements on the right side of web pages. We would likely see a reversal in this pattern however, if the data was collected in a right-to-left reading language such as Arabic.

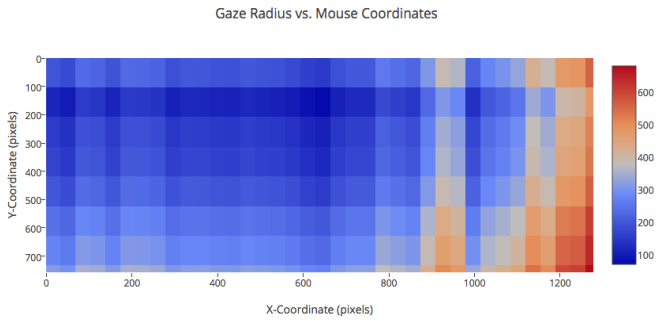

Fig. 5. Predicted gaze radius is smaller near top left and top middle portions of the screen.

Incorporating the direction component into our model for predicting actual x and y coordinates yielded less impressive results. In practice, simply predicting that the user's gaze is the same as their mouse coordinate delivered better results than our radius and direction approach. The MAE of our model was 241 pixels, compared to just 200 pixels for the baseline. We also analyzed the portion of model predictions that were within some number of pixels, and compared that to the true gaze coordinates in Table I. Interestingly, the user's gaze was within 100 pixels of their mouse for 39.5% of samples and within 200 pixels for 67% of the samples (mouse samples as described above). Our model fell far below the baseline for being within 100 pixels at only 15.3%, but closer when within 200 pixels at 57.5%.

TABLE I
Portion of predictions within 100, 150 and 200 pixels.

	100px	**150px**	**200px**
Model Prediction	15.3%	38%	57.5%
Baseline	39.5%	56.4%	67%

In Figure 6 we plot our model's MAE by "bucket" when predicting actual x and y coordinates, revealing a smaller MAE in portions of the screen where we also predict smaller radii. This distribution of model-accuracy is interesting, because in many portions of the screen our model significantly outperforms the baseline.

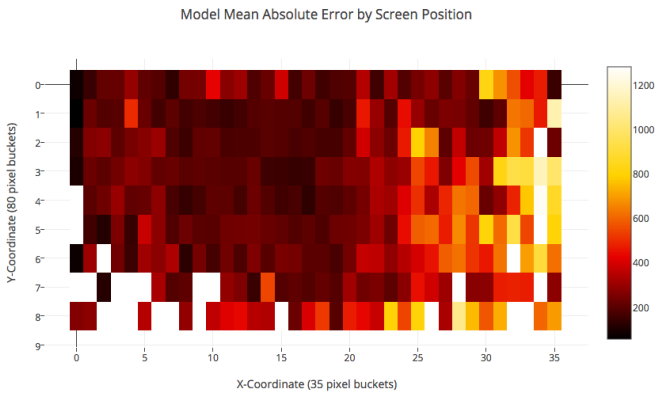

Fig. 6. Mean Absolute Error by Screen Position

This distribution of our model's accuracy led us to investigate our test data to see how it was distributed over the screen. In Figure 7 we plot every data point in our test set (please ignore the plotted arrow direction, as it is not important for this analysis). It's clear to

see that our model's accuracy performs much better in areas of the screen where there is more data.

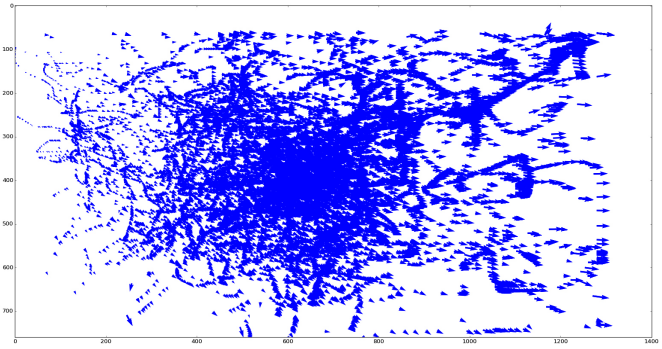

Fig. 7. Distribution of Test Data

VIII. Conclusion

Unfortunately, we defined robust baselines late in the process. We failed to define what constituted "quality" predictions until after we had spent a great deal of time analyzing the data and developing a predictive model. In retrospect, it would have been prudent to define reasonable baselines prior to developing a model. This would have allowed us to make a more informed comparison between our predictions and the chosen baseline.

Although our approach was outperformed by the baseline in general, we see significant promise over areas of the screen where more data was available. We believe that with enough data, it may be possible for our model to outperform the baseline in general, instead of only in particular portions of the screen.

References

[1] Qi Guo and Eugene Agichtein. 2010. Towards predicting web searcher gaze position from mouse movements. In CHI '10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10). ACM, New York, NY, USA, 3601-3606. DOI=10.1145/1753846.1754025 http://doi.acm.org/10.1145/1753846.1754025

[2] Mon Chu Chen, John R. Anderson, and Myeong Ho Sohn. 2001. What can a mouse cursor tell us more?: correlation of eye/mouse movements on web browsing. In CHI '01 Extended Abstracts on Human Factors in Computing Systems (CHI EA '01). ACM, New York, NY, USA, 281-282. DOI=10.1145/634067.634234 http://doi.acm.org/10.1145/634067.634234