

A heuristic-based approach for systematic error correction of gaze data for reading

Abhijit Mishra Michael Carl Pushpak Bhattacharya

(1) Indian Institute Of Technology, Bombay

(2) CRITT, Copenhagen Business School

(3) Indian Institute Of Technology, Bombay

abhijitmishra@cse.iitb.ac.in, mc.isv@cbs.dk, pb@cse.iitb.ac.in

ABSTRACT

In eye-tracking research, temporally constant deviations between users' intended gaze location and location captured by eye-samplers are referred to as systematic error. Systematic errors are frequent and add a lot of noise to the data. It also takes a lot of time and effort to manually correct such disparities. In this paper, we propose and validate a heuristic-based technique to reduce such errors associated with gaze fixations by shifting them to their true locations. This technique is exclusively applicable for reading tasks where the visual objects (characters) are placed on a grid in a sequential manner; which is often the case in psycholinguistic studies.

KEYWORDS: EYE-TRACKING, FIXATION CORRECTION, GAZE DATA MANIPULATION, SYSTEMATIC ERROR

1 Introduction

In psycholinguistic studies, eye tracking experiments have often been conducted to study the human way of analysing and synthesizing text. During reading, eye movement significantly relates to the cognitive load on participants. So, analysing gaze data is useful in proving/disproving hypotheses and extracting features for training and tuning machines. But eye trackers, after all, have certain limitations and they exhibit error in capturing gaze points of individuals. Such errors could be classified into variable and systematic errors (Hornof and Halverson, 2002). Variable error is nothing but dispersed gaze-points around the intended fixation which indicate lack of precision of eye-trackers. Systematic error, on the other hand, is the drift between the gaze-point locations captured by the eye-trackers and the intended fixation. It may be caused by imperfect calibration, head movement, astigmatism and other sources (LC Technologies, 2000). With the advent of sophisticated eye-trackers (Tobii, SR Research Eyelink etc.) it has been possible to reduce variable errors. But yet there is still a demand of tools and techniques to handle systematic errors which often imposes adverse impact on gaze-point analysis.

Various methods have been proposed to handle systematic error associated with fixations. Abrams and Jonides (1988) and Juhasz et.al (2006) proposed recalibration in the course of experiment which may not be applicable for linguistic analysis since such interruptions would reduce the quality of task. For example: during translation process studies participants cache contextual evidences in their short term memory, which could be lost by such interruptions.

Hornof and Halverson (2002) introduced Required Fixation Location (RFL) technique in which they identify RFLs i.e some points on the screen which indicates the actual fixation of the candidates at a specified time. In some of the experiments they record RFLs by asking participants to place the mouse cursor over the objects they were looking at. Then they measure the discrepancies between RFLs and fixations recorded by eye-trackers and shift the fixations to the true locations. This method is not very useful where one cannot ask the user to indicate RFLs. For example, during translation studies the participant might be busy typing the translations and reading the text simultaneously. Similar is the case with annotation tasks where the user has to read and annotate a text.

The Gaze to Word Mapping (GWM) modules introduced by Špakov, (2007) is a heuristic based approach. The underlying algorithm does not make a simplistic link between the x-y coordinates of a fixation and the location of a word on the monitor, but rather tries to account for certain documented effects, closely resembling to our technique. While is it quite reasonable to believe that participants tilt towards the end of reading lines; it doesn't clearly show us a way to determine the line which the participant is looking at; given initial few fixations are nonlinear in nature. Our algorithm tries to overcome this by introducing a scoring function which guesses which line a participant is focusing on; given N initial non-linear/linear fixations starting at time T.

The Mode-of-disparities error correction technique proposed by Zhang and Hornof (2011) is useful when the visual objects are arranged in an irregular manner but fails when objects are placed on a grid such as placing a paragraph for reading.

Intuitively, for reading and writing tasks vertical displacement of fixations contribute more to the noise than that of horizontal. So in this article, we focus more on vertical directional adjustment.

Initially, before processing fixations, a set of virtual horizontal lines are drawn by joining the centre coordinates of character belonging to the respective textual lines. Fixations are extracted from the noisy data and stored sequentially in a temporal order¹. Then they are processed and corrected in three stages. In first stage, fixations are shifted to lie on the nearest virtual lines. In the second stage transient fixations are corrected. Finally, participant's Reading Line (RL) is predicted and deviating fixations are shifted to the corresponding RLs.

This technique is applied on the Translation Process Research (TPR) database (Carl, 2012) recorded by Tobii eye-tracker using Translog-II (Carl 2012) software. Then validation is done across manually corrected fixations. Qualitative analysis is done by replaying the recorded and corrected data in Translog. In all the cases we have assumed left to right reading and writing but the technique could be slightly modified to support for languages adopting Arabic scripts.

2 Heuristics for Fixation Correction

In order to hand code rules for fixation correction, we have extensively studied the fixation sequences in TPR database. The database contains more than 450 recordings for translation, post-editing and reading experiments in 7 languages and are collected over last 5 years by a following a systematic initial experimental setup (Carl, M. and Jakobsen A.L. 2009); the eye-tracker used being Tobii, a remote eye-tracker. However, this does not bias our heuristics since many of the psycholinguistic experiments involving reading and writing tend to follow similar set-up. Moreover, other state of the art remote eye-trackers (such as SR research, SMI vision) report same or more accuracy as Tobii.

Fixations in the recorded data are corrected in three phrases as described below.

2.1 Shifting fixations to the nearest line

First of all, recorded fixations could be dispersed over the screen whereas the intended fixation should only possibly lie on visual objects such as characters. A fixation lying on the blank space between two lines is nothing but an indication of error. So the first step is to shift the fixations vertically to the nearest line. To come up with discrete lines we have taken the cursor coordinates of each character in a line and joined them to draw a virtual line. Figure 1 illustrates a set of virtual lines going through the text. These lines serve as Reading Lines (RLs) in the later processing stages.

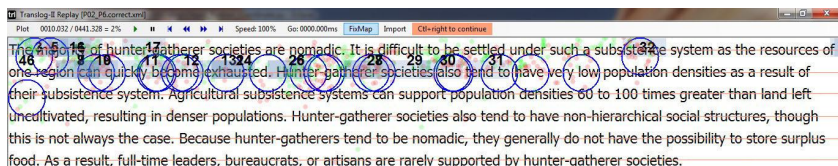


FIGURE 1 – Shifting fixations to nearest virtual lines

¹ Fixation sequencing is done on the basis of time of occurrence of the fixations. For example, if we say a particular fixation (say F2) follows/precedes another fixation (F1), we mean, F1 occurs sooner/later than F2 even if F2 appears to the left/right of F1 co-ordinate wise.

Figure 1 is a screen dump of Translog II, The orange lines represent virtual lines (Reading Lines). The red and green dots represent gaze samples of left and right eyes and blue circles represent fixations.

Sometimes, shifting fixations to the nearest virtual line is not enough. Upon closely looking at figure-1, one would predict that the participant is trying to read line1. But after shifting the fixations most of the fixations fall on line2.

After this step, it becomes easy to obtain systematic patterns which reduces the randomness and hence, the number of rules to be used for correction.

2.2 Discarding transient fixations

Transient Fixations (TFs) are very short duration fixations which occur in between two fixations falling nearer to each other (on the same line or just a line apart) and located far away from each of them. In other words, upon joining three fixations if we observe a spike and the tip of the spike is a short duration fixation, it is said to be transient. Figure 1 illustrates one TF.

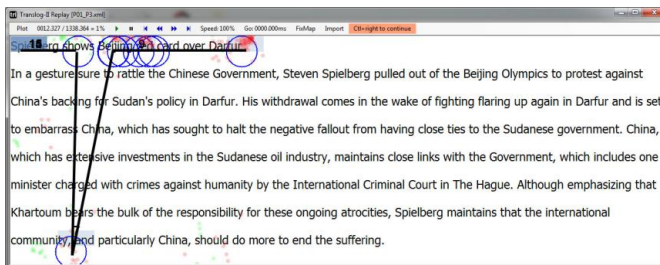


FIGURE 2 – Transient Fixations

Figure 2 shows one transient fixations. Upon joining 3 consecutive fixations involving one TF, we observe a spike.

In some studies, we do not need TFs to be present in our data as the fixation count un-necessarily grows on account of TFs. Transient fixation may also add noise to the data in some cases where, for example, fixation count for a region is a part of our study. Suppose, for our translation studies if we want to count fixations in source text window (src) and target text window (tgt) during an interval of 20 seconds and a lot of transient fixations fall on tgt, the distribution will be completely different from that of if we discard transient fixations. Such cases would require discarding TFs.

2.3 Correcting continuous abnormalities in fixation sequences

In this stage we try to predict the Reading Line (RL) of the participant at a specified time period and try to shift way-ward fixations within that time period to the corresponding RL. For instance, consider the case where the user starts reading the text from left to right and the eye-tracker records F fixations within the timespan of T. After shifting those fixations to the nearest lines, it is observed that first N out of the F fixations lie on line1. Here we can, to some extent, believe that the RL for the participant for the timespan T is line1. Now suppose the rest (F-N) fixations

lie on line2 and the X co-ordinate of these fixations are greater than those of first N fixations. In this case, it is unlikely that the RL of the user has changed from line 1 to line2. Hence those (F-N) fixations have to be relocated to line1.

Assuming that the initial calibration is perfect enough for a particular experiment session and the line spacing width significant (which is often the set up in linguistic studies) , it is reasonable to believe that most of the first N (co-ordinate wise) fixations decide the RLs. The intuition behind such an assumption is that, if the participant is reading from left to right, after reading certain words from left, there will be a gradual head movement and tilting which might contribute to shifting of fixations to the next/previous line.

The value of N is decided by taking samples from the recorded data and observing it by replaying the recordings. It is highly possible that the first N fixations could be distributed amongst different lines; each being a candidate RL. In such cases we infer the RL by ranking the candidates as follows

$$RL = \operatorname{argmax}_{r \in R} \sum_{f \in \text{first}N} \sum_{r \in R} (\delta_r(f.Y) \times \text{dur}(f))$$

where R is the set of RLs, δ is Dirac Delta function and $\text{dur}(f)$ is duration of fixation f

The first part of the summation represents fixation frequency distribution amongst the RIs. The intuition behind taking such a function is that during reading/writing, fixation duration and frequency are measurable factors providing evidences regarding participant's attention. The rationale behind taking Dirac Delta is that one particular fixation at time T could lie only on one Reading Line.

If the scores of two potential RLs match, RL is assigned to the line having maximum fixation. If that still matches, random assignment has to be done. Once the RL for a particular time period has been detected, the following two types of deviations are corrected.

Type A: This is a case when the user tries to read M^{th} line from left to right. A few fixations (say P) lie on line M spatially followed by a number of fixations (say F) on line M+1. The x-coordinates of those F fixations are greater than those of P. In such cases those F fixations are shifted upward to line M unchanging x-coordinates. (Figure 3 Type A)

Type B: Here, the user tries to read M^{th} line from left to right. A few fixations (say P) lie on line M spatially followed by a number of fixations (say F) on line M-1. The x-coordinates of those F fixations are greater than those of P. In such cases those F fixations are shifted downward to line M unchanging x-coordinates. (Figure 3 Type B)

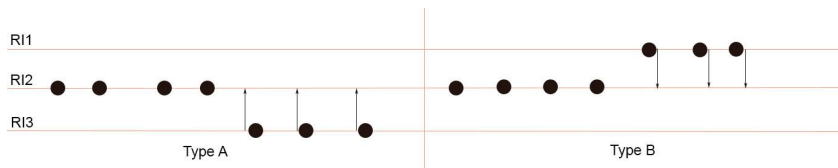


FIGURE 3 – Type A and Type B deviations

3 Algorithm

correctFixations (N, loggedData):

```

fixationSet := extractFixations(loggedData)
fixationSet = sortByTimeOfOccurrence (fixationSet)
RL_Set := extractDistinctYCoordinate(loggedData)
Foreach fixation in fixationSet:
    Re-assign the y-coordinate of the fixation to that of the closest RL
correctTransientFixations (fixationSet)
correctAbnormalities (fixationSet,N,RL_Set)
Update logged data with fixationSet
return

```

correctTransientFixations (fixationSet):

```

averageFixationDuration := ComputeAverageFixationDuration(fixationSet)
Foreach fixation in fixationSet:
    IF previousFixation doesn't exist OR nextFixation doesn't exist
        Continue
    IF abs(previousFixation.Y -nextFixation.Y) << abs(previousFixation.Y -fixation.Y)
        AND fixation.duration << averageFixationDuration
        Delete fixation from fixationSet

```

correctAbnormalities (fixationSet,N,RL_Set):

```

startingPoint := 1
firstN: = selectNFixations(fixationSet, startingPoint,N)
RL:= getRLWithMaximumScore(firstN,RL_Set)
X: = getLargestXCoordinate(firstN,RL)
targetSet: = setDifference(fixationSet,firstN)
Foreach fixation in targetSet -:
    startingPoint+=1
    L1 = getLineNumber(fixation.Y)
    L2 = getLineNumber (RL)
    IF previousFixation doesn't exist OR nextFixation doesn't exist
        Continue
    IF (previousFixation.X > fixation.X and previousFixation.X>nextFixation.X)
        RL = getRLWithMaximumScore(firstN,RL_Set)
        X = getLargestXCoordinate(firstN,RL)
        targetSet = setDifference(fixationSet,firstN)
        Continue
    IF (abs(L2-L1)==1 and fixation.X >X)
        fixation.Y = RL

```

getRLWithMaximumScore (firstN,RL_Set)

```

RL = argmaxr∈RL_Set ∑f∈firstN ∑r∈RL_Set δr (f.Y) × dur(f)
Return RL

```

The subroutines selectNFixations returns N fixations from the starting index. Similarly, getLargestXCoordinate returns the right-most fixation lying on an RL.

4 Validation

This technique was applied on Spanish and Danish translation and post-editing recording sessions from Translation Process Research (TPR) database. Qualitative analysis of the corrected data showed improvement.

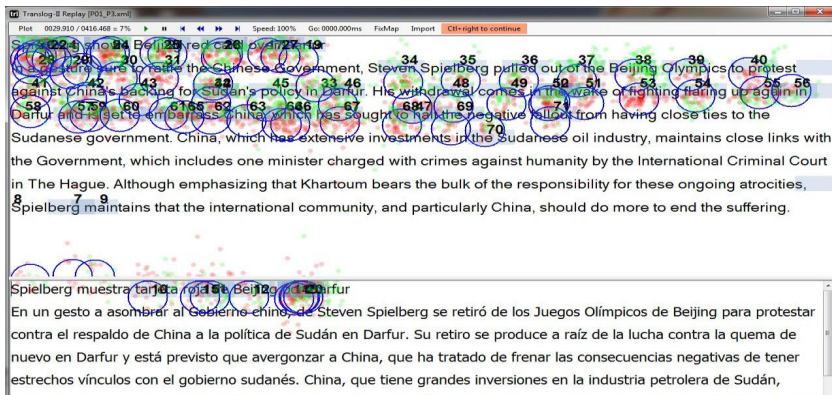


FIGURE 5 – Uncorrected fixations

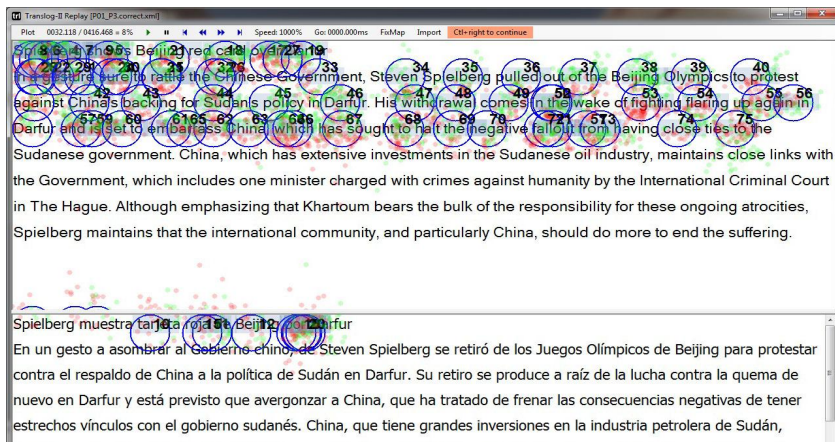


FIGURE 6 – Automatically corrected fixations

As we can see in the initial data (Figure 5), the fixation distribution is noisy and there is an overlap among fixations lying on line 3 and 4. After correction (Figure 6) the noise is significantly reduced. Fixations are labelled as per their temporal ordering.

5 Comparison with manual correction

We compared our output with manual corrections done for Spanish and Danish TPR data. Since our method shifts most of the fixations and manual correction only involves correcting only certain badly shifted fixations by mapping an appropriate word to the fixation, we checked for what fraction of manual correction could be successfully carried out by our method.

First, we mapped our fixations to the words on which they lie. Then from the original data we took the timestamp of those fixations which were corrected manually. For those timestamps we collected Fixation-to-word mapping for both the corrected versions and produced the Longest Overlapping Subsequence (LOS) between the mapped words. If the length of the LOS is more than the sum of the character counts of those two corresponding words, it is considered to be a valid correction.

For different values of N, we checked for the percentage of correction done with respect to manual correction. The results are shown by the following table

	N=3	N=6	N=10
Danish (10 sessions)	63%	83%	79%
Spanish (40 sessions)	55%	81%	81%

TABLE 1 – Automatic Vs Manual Correction

6 Conclusions

In this article, we presented a mechanism to correct systematic error associated with fixations by applying certain heuristics. The advantage of this method is, it can be applied both online (in the course of experiments) and offline. But the correction quality depends on the value of N and other parameters like initial experimental set-up and degree of randomness of fixations etc. It works best for shallow visualization studies; making it quite useful in studies like Translation Process Study, Sentiment Analysis etc.

There are certainly several factors for drift and imprecision apart from what we have taken into account. For instance, if the eye-tracker maps all gaze sampled, say 3cm below the intended location (because the head was permanently moved), all gaze samples are 3cm distorted, including the ones on the first N words in a line. Our algorithm fails to detect this. Of course, for the studies involving writing, we can get this constant drift (3cm) by comparing the cursor and the fixation positions during writing and finding out the average deviations. This is somewhat similar to RFL techniques assuming that a person's region of interest should not be very far away from the cursor position.

Our technique also fails if fixations are highly randomly distributed; which might be a case for studies involving detailed reading. In such cases, we also do not know the all the causes of the deviating fixations. Future work includes exploring and involving other case than just the two types of deviations that we took into account here. More cases and heuristics have to be included. A better validation technique has to be introduced as well.

References

- Abrams, R. A., & Jonides, J. (1988). *Programming saccadic eye movements*. *Journal of Experimental Psychology: Human Perception and Performance*, 14, 428–443.
- Hornof, A. J., & Halverson, T. (2002). *Cleaning up systematic error in eye-tracking data by using required fixation locations*. *Behavior Research Methods, Instruments, & Computers*, 34, 592–604.
- Zhang, Y., & Hornof, A. J. (2011). *Mode-of-disparities error correction of eye-tracking data*. *Behavior Research Methods*, 43, 834–842. doi:10.3758/s13428-011-0073-0
- Technologies, L. C. (2000). *The Eyegaze Development System: A tool for eyetracking applications*. Fairfax, VA
- Carl, Michael (2012). *Translog-II: A Program for Recording User Activity Data for Empirical Reading and Writing Research*, In Proceedings of the *Eight International Conference on Language Resources and Evaluation, European Language Resources Association (ELRA)*
- Carl, M. and Jakobsen A.L. (2009). *Towards statistical modelling of translators' activity data*. *International Journal of Speech Technology*, 12(4).
<http://www.springerlink.com/content/3745875x22883306/>.
- Carl Michael (2012). *The CRITT TPR-DB 1.0: A Database for Empirical Human Translation Process Research*. AMTA 2012 Workshop on *Post-Editing Technology and Practice (WPTP-2012)*
- Špakov, O. (2007). *GWM – the Gaze-to-Word Mapping Tool*, available online at <http://www.cs.uta.fi/~oleg/gwm.html>.

