Fusing Eye-gaze and Speech Recognition for Tracking in an Automatic Reading Tutor

A Step in the Right Direction?

Rasmussen, Morten Højfeldt; Tan, Zheng-Hua

Published in:
SLaTE 2013 Proceedings

Publication date:
2013

Document Version
Early version, also known as pre-print

Link to publication from Aalborg University

Citation for published version (APA):
Rasmussen, M. H., & Tan, Z.-H. (2013). Fusing Eye-gaze and Speech Recognition for Tracking in an Automatic Reading Tutor: A Step in the Right Direction? In *SLaTE 2013 Proceedings* (pp. 112-115). ISCA. http://www.slate2013.org/images/SLaTE%202013%20Abstracts%20Book%202.0.pdf

Fusing Eye-gaze and Speech Recognition for Tracking in an Automatic Reading Tutor – A Step in the Right Direction?

Morten Højfeldt Rasmussen [1], *Zheng-Hua Tan*[2]

[1]SpeechOp ApS, Aalborg, Denmark
[2] Department of Electronic Systems, Aalborg University, Aalborg, Denmark

mr@speechop.com, zt@es.aau.dk

Abstract

In this paper we present a novel approach for automatically tracking the reading progress using a combination of eye-gaze tracking and speech recognition. The two are fused by first generating word probabilities based on eye-gaze information and then using these probabilities to augment the language model probabilities during speech recognition. Experimental results on a small dataset show that the tracking error rate of the system using only speech recognition is 34.9% whereas the tracking error rate for the system that incorporates eye-gaze tracking into the speech recognizer is 31.2% – a relative improvement of 10.6%.

Index Terms: automatic reading tutor, eye-gaze tracking, speech recognition

1. Introduction

The tasks of automatic reading tutors (ART) are many. It might provide live feedback to the reader if he or she pauses during a reading session, which could indicate that the reader has trouble reading a word. It might also provide corrective feedback if the reader misreads a word. The feedback provided could be in the form of a picture (if possible), reading the word in question out loud, or a number of other ways [1]. It could also be a platform for reading assessment [2] or provide a speech driven interface to the reader [3].

In order to automatically provide feedback or assess reading proficiency the reading tutor needs to detect some level of reading activity. At least three modalities can be used to this end: eye-gaze (or gaze), speech, and manual feedback requests. For example in [4] the authors use an eye-gaze tracker to provide assistance when the reader looks at a word for more than 360 msec and in [1], [2], and [3] the authors use automatic speech recognizers (ASR) in three different reading tutors. Manual requests for feedback can be done using the mouse; however, systems that provide this as the only way to get feedback fall in the category of interactive books rather than ART.

In this paper we focus on the domain of adult dyslexic read speech. Tracking the reading progress of people with dyslexia is challenging as they produce more miscues (misread words and other disfluencies [5]) than people without this developmental reading disorder.

Reading usually occurs in a progressive way. However, sometimes the reader returns to previously read words in order to revise or remember what was read. This is especially true for people with dyslexia who struggle with reading [1]. This means that an ART should be able to determine which word the reader is supposed to read next, in order to provide assistive feedback if necessary. In [6] we detailed a speech recognition based tracking system. In this paper we extend that work on tracking by using an eye-gaze tracker and fusing gaze and speech.

To the authors' knowledge, no prior research has been done with regards to fusing gaze information and speech recognition for tracking reading. In [7] N-Best lists generated from a speech recognizer are rescored based on gaze points for a visual-based goal-driven task. In the experiment the participant describes a geographical map with landmarks to another person. The landmarks are placed relatively far from each other. The N-Best lists were used as a substitute for implementing the rescoring in the Viterbi decoding.

The rest of the paper is organized as follows: In Section 2 we present our tracking methods, in Section 3 we describe the data collection and transcription, in Section 4 we show and discuss the results, and in Section 5 we conclude the work.

2. Tracking

In this section we'll explain the relationship between gaze and reading and how we track the reading progress using an eye-gaze tracker, an ASR and a fusion of gaze and speech.

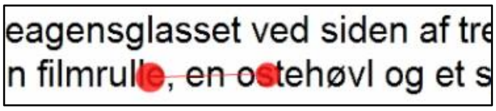
2.1. Eye-gaze tracking and reading

During reading the eyes move in a sequence of fixations and saccades [4]. Usually the saccades move from left to right but sometimes they do the opposite – for example when the reader revises what was previously read.

2.1.1. Gaze events

Gaze events can be ordered into a number of categories. The authors of [8] list the following: fixation, glissade, saccade, smooth pursuit, and blink. For this paper we will focus on fixations as they can be thought of as the anchor points of the gaze events.

A fixation can be defined in different ways. One way is to define it as a given period where the gaze points generated from an eye-gaze tracker falls within a region. Another way is to define it as gaze points that are not classified as saccades, glissades or noise as in [8]. For this paper we use the Tobii Fixation Filter [9], which finds fixations by grouping (or clustering) gaze points using the method described in [10]. An example of a saccade and two fixation events mapped onto the text being read is shown in Figure 1.



eagensglasset ved siden af tre
n filmrulle, en ostehøvl og et s

Figure 1: *Three gaze events: two fixations (the red discs) and one saccade (the red line connecting the two discs). The reader is reading the upper line, but the eye-gaze tracker offsets the gaze to the lower line.*

2.1.2. Sources of errors

Ideally the fixations will map onto the lines being read but tracking errors add noise to the gaze points. According to [11] there are at least three sources of errors when determining what is being focused on using an eye-gaze tracker: the quality of the setup, drift of the calibration, and the biological characteristics of the eye. The quality of the setup depends on the eye-gaze tracker used and how successful the session calibration is. But even if the calibration is successful the accuracy will usually degrade during a session due to drifting errors that can occur e.g. when changes in head-position are incorrectly compensated for. The biological characteristics of an eye give a visual field of focus of around 1°.

The effect of the described errors is an offset plus noise on the gaze points. An example of offset error is shown in Figure 1. The reader is reading the text in the upper line, but the eye-gaze tracker maps the gaze onto the lower line.

2.1.3. The word being focused on

The task of mapping the gaze points to text words is not just a matter of finding the nearest word given a gaze point due to the errors described in Section 2.1.2. The authors of [11] introduce the notion of "sticky" and "magnetic" lines, where sticky lines keeps the focus to the (assumed) correct line and magnetic lines sets the focus to the next line when a gaze point jump from the end of a line to the beginning of the next is detected. Sticky and magnetic lines alleviate some of the errors in the vertical direction. Horizontal errors are less pronounced – which is fortunate since they would be harder to detect.

2.1.4. Tracking the reading using eye-gaze tracking

The magnetic lines method described above was used as inspiration to our eye-gaze tracking algorithm. However, we only use the information of which line is being focused on to estimate the per-line offset error in the vertical direction. We calculate the offset for each line, since the magnitude of the offset varies as a function of the y-position.

We define "line-index" as the index of the line we believe that the reader is focusing on. We assume that the reader starts reading from the first word and onwards and sets line-index to 1. A "next-line event" is detected, when the center of the i'th fixation cluster (the collection of fixation points belonging to one fixation) is near the end of a line and the i+1'th fixation cluster is near the beginning of the next line. The procedure for determining the word being focused on can be described like this:

1. *Get the next fixation cluster from the eye-gaze tracker.*
2. *Update the per-line offset estimate for the y-axis.*
3. *Calculate the center point (or mean) of the fixation points belonging to that cluster.*
4. *Subtract the y-axis offset from the center point.*
5. *Find the text word closest to the value calculated in 4.*
6. *If a next-line event is detected: increment the line-index.*
7. *GoTo 1.*

2.2. Speech recognition and tracking

Our previous work [6] within the area of tracking in ARTs involved using speech as the only modality for tracking. In that paper we showed that a language model that models the expected reading behavior of children – allowing for jumping back and forth in the text – works better than changing the task from trying to follow the child to forcing a strict left-to-right reading policy.

In this paper we also use a language model that allows for jumping back and forth in the text. This model is further relaxed to a word-loop; essentially giving word transitions equal probability. This relaxation has been chosen in order to accentuate the effect of including gaze information in tracking. Each word was given a unique ID in order to differentiate words in the same text segment with the same spelling.

2.3. Fusing gaze and speech for tracking

This section describes how we handle synchronization issues between gaze and speech, calculate word probabilities from gaze points and apply them in the speech recognizer.

2.3.1. Synchronizing gaze and speech

No matter how accurately the gaze points and the audio stream are synchronized, there will always be a non-deterministic delay from the time the reader focused on a word to when it was uttered. This delay is at least the time it takes for the reader to see, interpret, and utter the word. The authors of [7] report this delay to be typically between 430 and 902 msec. In this paper we don't explicitly try to synchronize gaze and speech. Instead we delay the gaze points by 430 msec because we want to avoid overshooting the reference changing points due to the way we calculate the word probabilities from the gaze points.

2.3.2. Word probability from gaze points

Similar to [11] we want to update the language model based on the gaze information. Instead of making a hard decision on which word is being focused on as in Section 2.1.4 (bullet point *5.* in the list) we choose to assign a probability to all the text words at a given time. Given a fixation cluster, the probability for each text word is calculated from a bivariate normal distribution with the probability density function given as:

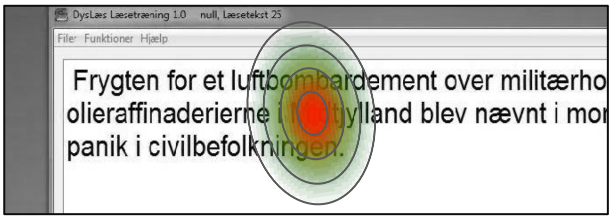
$$f(\boldsymbol{x}_n, \boldsymbol{\mu}_t) = \frac{1}{2\pi}\ \det(\boldsymbol{\Sigma})^{-\frac{1}{2}}\ e^{\left(-\frac{1}{2}(x_n-\boldsymbol{\mu}_t)^T\boldsymbol{\Sigma}^{-1}(x_n-\boldsymbol{\mu}_t)\right)}, \qquad (1)$$

where $\boldsymbol{x}_n$ is the coordinate for the center-point of the *n*'th word we want the probability for, $\boldsymbol{\mu}$ is the mean vector (center point) consisting of the mean of the x- and y-coordinates of the gaze points belonging to the fixation cluster. $\boldsymbol{\Sigma}$ is the covariance matrix of all fixation clusters and is estimated as a running average over an entire session. An illustration of the normal distribution can be seen in Figure 2. Since, however, the speech lags behind the gaze we estimate the probability of word *n* as:

$$P_t^*(w_n) = f(\boldsymbol{x}_n, \boldsymbol{\mu}_t) + f(\boldsymbol{x}_n, \mathbf{c}_{n-1}) \qquad (2)$$

where P_t^* is the un-normalized word probability for word *n* at time *t* calculated from the gaze information and $\mathbf{c}_{n-1}$ is the center point of word *n*-1. $\mathbf{c}_{n-1}$ is estimated as the word index of the word closest to $\boldsymbol{\mu}_t$ minus one. The word probabilities are then normalized in order to ensure that their sum equals 1.

Linear interpolation of the word probabilities is used for periods with no fixations.



Frygten for et luftbombardement over militærho olieraffinaderierne i [illegible]tjylland blev nævnt i mor panik i civilbefolkningen.

Figure 2: *Illustration of the estimated bivariate normal distribution.*

2.3.3. *Integrating gaze information in the ASR*

With the word probabilities calculated based on the gaze information the only thing left is to apply them during recognition. To that end, we have modified Sphinx-4 slightly so that whenever it encounters a word in the recognition lattice, it multiplies (which becomes an addition in the log domain) the gaze probability (P_t^*) of that word at the given time by the ASR probability of the search path that ends in the word.

3. Data collection and transcription

In this section we will describe the data collection and transcription.

3.1. Data collection and experiment setup

The setup for the data collection can be seen in Figure 3. The participant was equipped with a headset and was told to read the displayed text out loud. The eye-gaze tracker was calibrated before each session which took roughly half a minute.

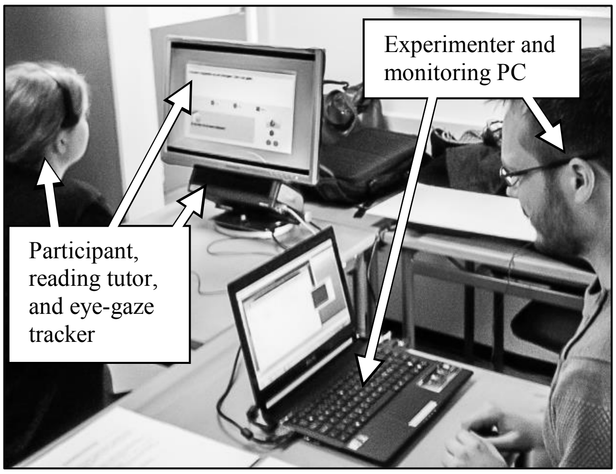

Experimenter and monitoring PC
Participant, reading tutor, and eye-gaze tracker

Figure 3: *The experiment setup.*

After the calibration the participant was given control of the mouse and began the reading session. Each participant read the same 23 short text segments of varying difficulty [1]. The experimenter was monitoring a live view of the eye-gaze tracking during each session and would tell the participant to adjust his or her position if the head had moved too far away from the optimal zone. Some dropped gaze points were observed, which was expected.

The data was collected for four adults with dyslexia.

3.2. Transcribing the data

Each session was time-segmented into words and miscues. Each speech event was assigned a target word index, indicating the word position in the prompt (or target) text. These word indices were then used to generate the tracking reference. Reference changing points were placed just after correctly read words were uttered. Miscues were filtered out.

4. Results and discussions

In this section we describe the evaluation method and present and discuss the results.

4.1. Evaluation method

The evaluation method used in this paper builds on [6] but moves away from the notion of speech events (segments with either a word or miscue) to word index changes – that is whenever the focus moves from one word to another. Since the tracking reference was generated by a human who applies judgment when placing changing points, a tolerance interval is introduced similar to that in [12]. Another reason for using a tolerance interval is that even though the reading tutor might be slightly off in detecting a changing point this has no practical significance in most settings.

Given a tolerance interval, the tracking error rate is calculated as the total number of changing points in the reading tutor's output minus correct changing points that fall within ±tolerance of the reference point. In this work a tolerance of ±250 msec was applied.

An illustration of the tolerance interval is shown in Figure 4; where *T* is the tolerance value, the two blue discs are the reference changing points, and the three red diamonds are the changing points detected by the system (hypothesized changing points). The two rightmost hypothesized changing points are tracking errors in the example. The leftmost hypothesized changing point falls within the tolerance interval of a reference changing point with the same word ID and is therefore marked as being correct. No hypothesized changing point falls within the tolerance interval of the leftmost reference point, which results in a tracking error. The tracking error rate for the example is 3/2 = 1.5.

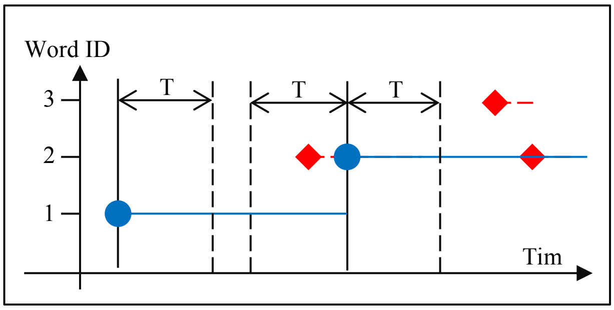


Figure 4: *Illustration of the tolerance interval. T is the tolerance value. The two blue discs are the reference changing points; the three red diamonds are the changing points detected by the system.*

4.2. Tracking error rate

The test data contains 1099 changing points in 92 utterances. The tracking error rates (TER) can be seen in Table 1. Note that the relative improvement of 10.6% of the system using gaze information and speech recognizer compared to the system that only uses the speech recognizer is significant, as the P-value of the matched-pairs test described in [13] is 0.042.

Table 1: *Tracking error rates.*

Tolerance	ASR	Gaze+ASR	P-value
±250 msec	0.349	0.312	0.042

Furthermore, a matched-pairs test of the distribution of errors (missed changing points vs. wrong and inserted changing points) was conducted and gave a P-value of 0.02 indicating that the distribution of errors is different as well.

The approach to tracking the reading position using only gaze information as presented in 2.1.4 performs poorly. The TER for this approach is 1.26. This was expected, as the reference is based on the words being read out loud and the readers will not always utter the words they look at.

The experiment has been conducted offline but the algorithms that track the reading and calculate word probabilities from gaze points (Sections 2.1.4 and 2.3.2) are causal and would perform in the same way in a live setting. Moving from an offline ASR setup to one where partial hypotheses would be used, however, would most probably result in performance degradations similar to those documented in [14].

5. Conclusions

In this paper we presented our work on fusing gaze information and speech for tracking the reading progress of people with dyslexia. The proposed fusion method achieved a 10.6% decrease in tracking error rate over the baseline ASR-only method.

In the course of doing the experiment we found that there is a multitude of parameters to tweak and equally many design options to consider – and since this was the first step in fusing eye-gaze information and speech recognition for tracking reading progress – we are confident that there'll be ways to fuse the two that improves the performance more significantly than what we have presented here.

6. Acknowledgements

The authors would like to thank master student Julia Alexandra Vigo for her help in collecting the data and Anders Olesen Sigh and Karina Fuhr Pedersen at VUC Nordjylland for facilitating the contact to the test participants.

7. References

[1] Pedersen, J. S., "User Centred Design Of a Multimodal Reading Training System for Dyslexics", Ph.D. thesis, Aalborg University, 2009.

[2] Duchateau, J., Kong, Y. O., Cleuren, L., Latacz, L., Roelens, J., Samir, A., Demuynck, K., Ghesquière, P., Verhelst, W. and hamme, H. V., "Developing a reading tutor: Design and evaluation of dedicated speech recognition and synthesis modules", Speech Communication, Volume 51, Issue 10, pp 985–994, 2009.

[3] Mostow, J., "Why and How Our Automated Reading Tutor Listens", International Symposium on Automatic Detection of Errors in Pronunciation Training (ISADEPT), KTH, Stockholm, Sweden, pp 43–52, 2012.

[4] Sibert, J. L. and Gokturk, M., "The Reading Assistant: Eye Gaze Triggered Auditory Prompting for Reading Remediation", Procedings of ACM Symposium on User Interface Software and Technology, pp 101–107, 2000.

[5] Rasmussen, M. H., Lindberg, B., Tan, Z.-H., "Combining Acoustic and Language Model Miscue Detection Methods for Adult Dyslexic Read Speech", International Speech Communication Association Special Interest Group, Workshop on Speech and Language Technology in Education, Venice, Italy, 2011.

[6] Rasmussen, M. H., Mostow, J., Tan, Z.-H., Lindberg, B., & Li, Y., "Evaluating Tracking Accuracy of an Automatic Reading Tutor", International Speech Communication Association Special Interest Group, Workshop on Speech and Language Technology in Education, Venice, Italy, 2011.

[7] Cooke, N. J., "Gaze-contingent automatic speech recognition", Ph.D. thesis, University of Birmingham, 2006.

[8] Nyström, M. and Holmqvist, K., "An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data", Behavior Research Methods, Volume 42, Issue 1, pp 188–204, 2010.

[9] User Manual, "Tobii Studio Version 3.2", Online: http://www.tobii.com/en/eye-tracking-research/global/library/manuals/, accessed on 8 April 2013.

[10] Olsson, P., "Real-time and offline filters for eye tracking", Msc. thesis, KTH Royal Institute of Technology, 2007.

[11] Hyrskykari, A., "Utilizing eye movements: Overcoming inaccuracy while tracking the focus of attention during reading", Computers in Human Behavior, Volume 22, Issue 4, pp 657–671, 2006.

[12] Koh, C.-W. E., "Speaker Diarization of News Broadcasts and Meeting Recordings", Msc. Thesis, Nanyang Technological University, 2009.

[13] Gillick, L., Cox, S. J., "Some statistical issues in the comparison of speech recognition algorithms", International Conference on Acoustics, Speech, and Signal Processing, pp 532–535, 1989.

[14] Li, Y., and Mostow, J., "Evaluating and improving real-time tracking of children's oral reading", Florida Artificial Intelligence Research Society Conference, pp 488–491, 2012.