Vocabulometer: a Web Platform for Document and Reader Mutual Analysis

Olivier Augereau, Clément Jacquet, Koichi Kise, Nicholas Journet

▶ **To cite this version:**

HAL Id: hal-01885247

https://hal.science/hal-01885247

Submitted on 1 Oct 2018

Vocabulometer: a Web Platform for Document and Reader Mutual Analysis

Olivier Augereau, Clément Jacquet, Koichi Kise
Osaka Prefecture University
IDAKS
Sakai, Japan
Email: augereau.o@gmail.com

Nicholas Journet
Bordeaux University
LaBRI
Talence, France
journet@labri.fr

Abstract—We present the Vocabulometer, a reading assistant system designed to record the reading activity of a user with an eye tracker and to extract mutual information about the users and the read documents. The Vocabulometer stands as a web platform and can be used for analyzing the comprehension of the user, the comprehensibility of the document, predicting the difficult words, recommending document according to the reader's in order to increase his skills, etc. Since the last years, with the development of low-cost eye trackers, the technology is now accessible for many people, which will allow using data mining and machine learning algorithms for the mutual analysis of documents and readers.

Keywords-eye tracking; document analysis; reader analysis; mutual analysis; web platform

I. INTRODUCTION

Using an eye tracker device allows detecting where a user is looking at in real time. Thanks to this tool we can determine precisely the position where a user is looking on a screen, analyze the read words and deduce his reading behavior.

Different kinds of eye trackers are available on the market: such as electro-oculography (EoG) and video-oculography (VoG). The EoG systems are composed of several electrodes placed on the skin around the eye measuring the eye movement. The disadvantage of this technology is that the user must wear sensors and the position of the gaze is unknown (only horizontal and vertical movement are quantified). By contrast, VoG systems can estimate the position of the eye gaze quite accurately. There are two different kinds of VoG: the mobile ones, which are wearable glasses; and the stationary ones, which are attached to a screen. Ones of the cheapest eye trackers on the market are Tobii Eye X and Tobii 4C, so we selected these eye trackers for our platform. Figure 1 shows the Tobii 4C. In the near future, eye trackers might be available by using a simple web camera on a tablet or a smartphone [1] which will make this web platform even more easily usable.

Since the last few years, much research about eye-trackers had been published about the three following research areas: (1) Biometry [2], [3], [4]: eye-trackers are used as an alternative to identifying people by analyzing the difference of eye movements. (2) Handicap [5], [6], [7]: eye-trackers are used to assist humans by controlling interfaces using the eye movement. It is especially useful for people who cannot move easily their hands or arms.
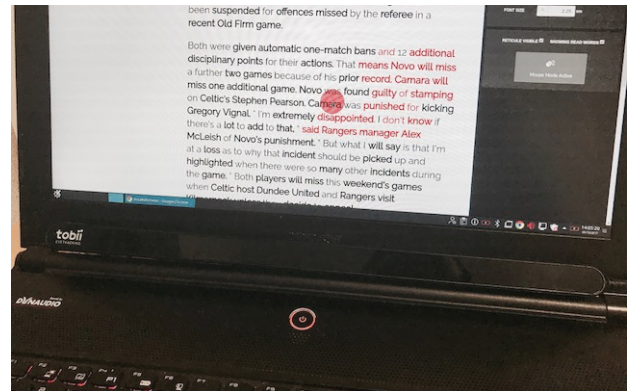

Figure 1: Tobii 4C stationary eye tracker attached beneath a laptop screen.

(3) Cognition [8], [9], [10]: research about autism or schizophrenia and their link to the eye movement.

More recently, the DIA (Document Image Analysis) research community get interested to use eye-trackers to correlate the document image understanding with the readers. For example, it has been proved that by analyzing the reader's eye movement it is possible to determine the objective comprehensibility of texts [11] or to find the subjective difficult words of a reader [12]. Analyzing the way people are interacting with documents is a promising way to understand the document in a better way.

The idea of the "reading-life log"[13] is to record the position of the eye gaze of the reader while reading, in order to save which word has been read at which time. This basic information can be used for several applications such as counting the number of read words [14], creating a reading summary [15], analyzing how people read comics [16] etc. In this paper, we go further this idea by proposing a web platform and services to help readers to improve their reading skill in a new language.

In the next section, we will present the architecture of the web platform and its interface. Then we will give some details about how the read words are detected by the application using eye gaze positions. At the end of the paper, we introduce a new feature for the "reading-life log": a reading recommender system which suggests documents containing similar vocabulary like the one the user knows or documents which contains vocabulary that the user should review.
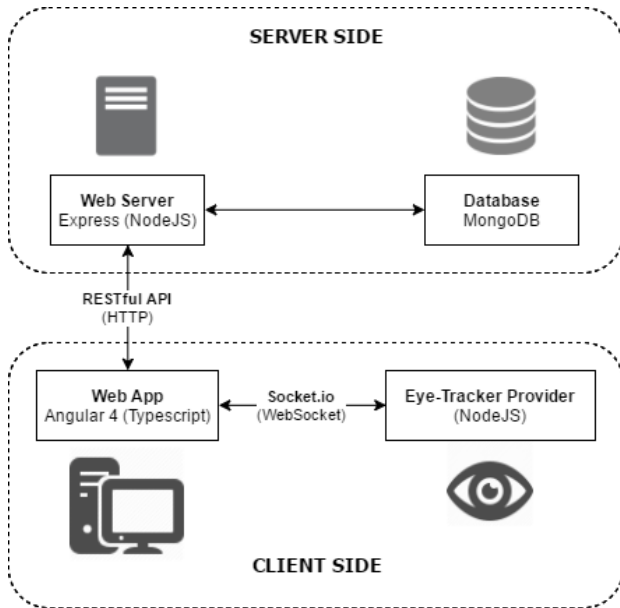
Figure 2: Server/client architecture of the Vocabulometer. This architecture is based on RESTful API.

II. EYE-TRACKING WEB PLATFORM

The web platform[1] can be freely accessed by anyone who wants to test the Vocabulometer. For now, it is compatible with Tobii Eye X and Tobii 4C eye-trackers. After installing the eye tracker SDK on a client computer, the user just has to plug the eye tracker and can start a reading session on the website. For a testing purpose, we also developed a "mouse" mode which emulates the eye gaze through the pointer of the mouse. It is a convenient way for testing the basic use of the Vocabulometer.

Several users can be managed, a new user can create an account and then records his reading behavior on the server. The user can thus have an access to a history of all his previous reading sessions and get useful statistics about his reading activity. The user can access different kinds of documents to read which are copyright free. He can also add his own documents to his personal database. The Vocabulometer contains also two recommender systems which are described in the section V.

A. Architecture

An overview of the Vocabulometer architecture is displayed in Fig.2. It is composed of four main parts:

1) the Web application, which contains the pages and the logic of the website,
2) the Web server, the gate to access the Database in a secure and easy way,
3) the Database, which stores the texts and the vocabulary (read words) of the users,
4) the Eye-tracker provider, which communicates through the Tobii API to the eye-tracker and sends to the web application the gaze positions.

The application communicates through AJAX calls to a RESTful API provided by this server in real time. This strategy offers the advantage of having an application and a web API totally independent enabling to build different types of applications (such as mobile applications) without having to change any part of the web server. The web application and the eye-tracker provider communicates through the real-time library socket.io[2], leveraging web sockets. The Database can be exported in JSON or CSV files.

B. Interface

A screenshot of the website is shown in fig. 3. The main Vocabulometer application consists of logging the readers' read words. In order to save only the words which have been really read, we save the words of a paragraph after a predetermined percentage of the words of the paragraph n have been read. We choose to make the log in this way in order to avoid recording words that have just been skimmed and to compensate the errors of the eye tracker due to calibration and noise.

Depending on the accuracy of the eye tracker, the line space and size of the font can be changed. If the eye tracker has a larger error, a larger line spacing and font size are recommended. In the future, this can be done automatically when more eye trackers will be available to be used within the application.
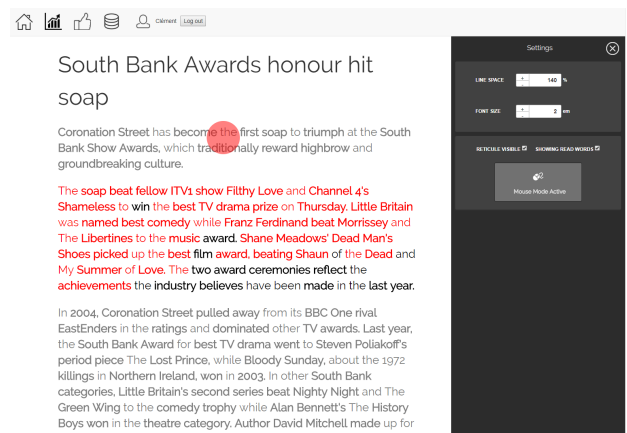

Figure 3: Illustration of the proposed web interface. The colors are used only in the debug mode for demonstration purpose. The red circle corresponds to eye gaze position of the reader. The words in red are the ones which have been detected as read by the application. If enough words of a paragraph are read, all the bold words of the paragraph are saved and added to the reader's profile. The non-bold words are stop words and are not saved.

The interface also contains a page where the user can see some statistics about his reading activity such as displayed in Fig. 4.

The website contains recommendation systems that help the user during the learning process :

Number of words read last week
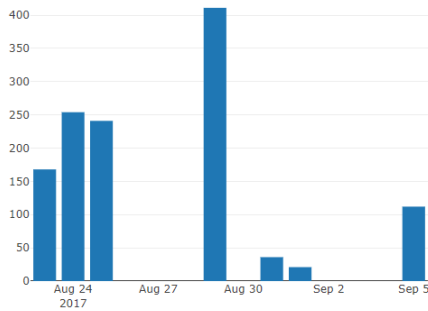
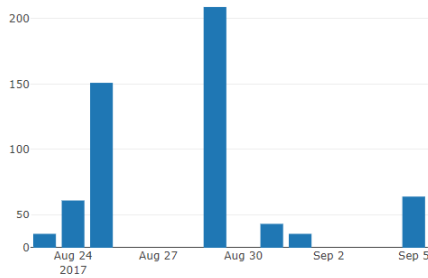## Number of new words encountered last week

Figure 4: Reading statistics. The user can see his progress through this page (reading sessions, amount of words read, etc.).

- "Revise your vocabulary". This system suggests documents with words that the reader read before but might have forget if he did not read them recently. It is based on the "Spaced repetition system" (see section V-B).
- "Easy texts for you". The aim of this system is to provide the user a new text for reading that contains a balanced amount of new vocabulary. This can be achieved by suggesting a new document from the Vocabulometer database according to the words that he read in previous reading sessions and the ones present in the documents that the user has never read. The propositions are based on the "Jaccard similarity" (see section V-A).

III. READING ANALYSIS

When one reads, his eyes are not having a smooth movement from the left to the right. The eyes are staring at some part of the words (fixations) and then jump to another part of the text (saccades). As presented in Fig. 5, the reading behavior can then be seen as a sequence of fixation and saccades.

The eye-tracker provides a raw signal corresponding to the estimated coordinates where the eyes are looking at on the screen, at a certain frequency (70Hz for Tobii Eye X and 90Hz for Tobii 4C). This signal is processed with
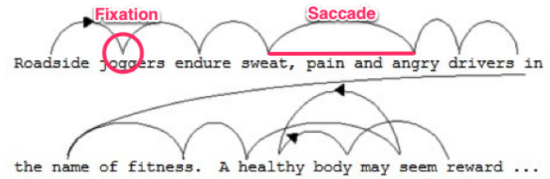

Figure 5: Fixations and saccades: typical eye movement while reading.

Buscher et al. algorithm [17] to extract the fixations and saccades.

A. Text model

A text contains words, punctuations, numbers and needs to be parsed in order to define what is considered as a word. For this purpose a tokenizer such as "Stanford Tokenizer[3]" is used. After this step, stemming and lemmatization can be applied depending on the application.

After a text has been displayed in a web browser, a user can change many parameters that leads to change the position of each word (interline space, character size, zoom, ...). He can even resize the window. The algorithm which is matching the eye gaze with the words needs to know in real time the position of each word. The Vocabulometer client part handles this problem by analyzing in real time the Document Object Model[4] of the displayed web page.

B. Detecting read paragraphs

Due to the limited accuracy of the eye trackers, a post processing such as the one proposed by Lima Sances et al. [18] would be necessary to align the eye gaze with the text. However this solution cannot be applied in real time. So, instead of trying to track which word is read or not, we propose to track if a paragraph as been read or not.

To do so, we analyze all the fixations, create a circle around them and check if they intersect with the rectangle of the bounding box of a word, the corresponding words are marked. Then, A paragraph is considered as read when n percents of its words are marked.

This threshold can be changed depending on the accuracy of the eye tracker; if the accuracy of the eye tracker is higher, n can be higher too. This could be easily set during a parametrization step when the user read a text for the first time. For now, the threshold has been chosen experimentally in order to optimize the performance of the system.

When a paragraph is considered as read, all the words in the paragraph are saved in the user's vocabulary database. Stop words such as "the", "a", "in", etc. are removed.

IV. EXPERIMENTS

When a user reads a document, every time he finishes a paragraph, this one is supposed to be detected as read by

---

[3]https://nlp.stanford.edu/software/tokenizer.html
[4]https://en.wikipedia.org/wiki/Document_Object_Model

the system. In order to assess the current accuracy of the system, tests have been conducted on reading detection. Data were collected on five users.

Two principal measurements have been assessed:

- full reading: how many paragraphs are correctly detected as read when the user read the full document?
- partial reading: how many paragraphs are correctly detected as read when the user read some paragraphs, but not all?

Therefore the test protocol has been split into two parts: one for the overall test and one for per-paragraph test. For both experiments, the threshold of the number of words to be read in order to detect a paragraph as read has been fixed to $n = 75\%$.

A. Full reading

In this experiment we let the participants choose three texts in the database. Then, they have been asked to read the document from the beginning until they wanted to stop (as some documents are very long, we did not want to force them to read for too long). Then the participant tells where he stopped in order to obtain the ground truth.

For each user and each of the three texts we estimated which paragraph were read and computed three metrics:

- the number of read paragraph detected as read: true positives `TP`
- the number of read paragraph not detected as read; false negatives `FN`
- the number of unread paragraph detected as read: false positives `FP`
- recall defined such as: $R = \frac{TP}{(TP+FN)}$
- precision defined such as: $P = \frac{TP}{(TP+FP)}$

Table I shows the number of well-detected read paragraphs by the system along with recall and precision metrics.

	TP	FP	FN	R	P
Count of paragraphs	96	2	6	94%	98%

Table I: Results of the full reading experiment (True Positives TP, False Positives FP, False Negative FN, Recall R and Precision P). Most of paragraphs are well detected.

We can see that in this scenario, the system achieves 98% of precision and 94% of recall. Very few unread paragraphs are detected as read, and only few of read paragraphs are not detected. In order to understand more where these problems comes from we run the next experiment, under more controlled conditions.

B. Partial reading

In this experiment, three specific documents have been selected. We especially selected a good balanced of "short" (less than 25 words), "medium" (between 25 and 50 words) and "long" (greater than 50 words) paragraphs. The user chooses two documents among the three and reads the text partially, i.e. he has been asked to skip one or two paragraphs from time to time. At the end of the experiment the reader reports which paragraphs he actually read.

As for the first part, true positives, true negatives, false positives, recall, and precision are estimated.

	TP	FP	FN	R	P
Overall	138	6	4	97%	96%
Short Paragraphs	48	4	4	92%	92%
Medium Paragraphs	48	2	0	100%	96%
Long Paragraphs	42	0	0	100%	100%

Table II: Count of paragraphs of the partial reading. The "Short paragraphs" are less than 25 words, the "Medium paragraphs" between 25 and 50 words and the "Long paragraphs" greater than 50 words.

The overall performances of the system are similar to the previous experience. Most of the paragraphs are well detected as read. We can see that for short paragraphs the detection is slightly less accurate. If few fixations are not detected or if some noise is included in the signal, then the short paragraphs might be mis-detected. However, for long paragraphs, the reading detection is working very well.

V. VOCABULOMETER : TOWARD A READING RECOMMENDER SYSTEM

More than just providing reading statistics, the Vocabulometer can be used to identify which words are difficult to read for a given user and then recommends some texts that are prone to train him efficiently. Eye tracking can be used to predict the reader's TOEIC (Test of English for International Communication) score [19] or Japanese reading understanding [20]. In these applications, the reading understanding is computed without using any information about the document such as the difficulty of the words or the vocabulary of the reader.

Thanks to the Vocabulameter, we can track the reading activity of the reader and especially, we can determine which words he reads and when he reads them. Thus, it is possible to estimate the knowledge of the reader through its vocabulary. A feedback of each learning sessions can be used to analyze the progress of the reader through the words he read. For example, the Vocabulameter stores which new words have been read, which words have not been read recently, etc.

We propose two recommendation systems based on thess information, one based on the Jaccard similarity measure and another based on the spaced repetition method.

A. Recommending documents with increasing difficulty

A simple approach to propose a document to a reader is to analyze the vocabulary he already read and the one contained in the set of documents he never read. If the reader already knows most of the words of a specific document, he might understand easily this document. On the contrary, if he does not know most of the words of the document it might be very hard for him to understand it. We propose to compute a similarity measure that allows to find the next document to read while guaranteeing

a progressive and increasing difficulty of the reading difficulty. To do so, we use the Jaccard similarity which computes a similarity between two sets of words. It is defined in equation (1), where A is the set of words learned by the user and B the set of words contained in a specific document.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \qquad (1)$$

Dividing by $|A \cup B|$ affects the scale of the similarity measure; the larger A set is, the smaller the similarity will be.So we slightly modified the formula as follows:

$$J'(A, B) = \frac{|A \cap B|}{|B|} \qquad (2)$$

Using equation (2), the similarity corresponds to the proportion of words known by the user in the text. This change does not have an impact on the recommendation results, the similarity scale is still between 0 and 1: 0 when the user knows no word of the text, and 1 when he knows all of them.

We finally suggest the user to read a text within a ranked list computed with equation 2.

B. Recommending a document based on the memory abilities of the user

The spaced repetition approach has been proposed by Piotr Wozniak [21] who is a researcher in the theory of spaced repetition and the author of SuperMemo[5], a software helping to learn vocabulary. It has been proved that we tend to forget a new word we learned according to an exponential formula, as the following:

$$F(t) = e^{-\alpha t}, \qquad (3)$$

where α is a positive real number representing the stability of memory, i.e. it models how fast the memory capacity $F(t)$ falls over time (when t increase) in the absence of reviewing. Wozniak then proposed a geometric progression in order to determine when a word needs to be reviewed[6]. Instead of computing the next time when a word needs to be read, we changed the equation to compute a score representing how much a word might be forgotten. The new equation is defined as follow:

$$\begin{cases} u_0 = 1 \\ u_n = \beta + F(g_n) * u_{n-1} \end{cases} \qquad (4)$$

where g_n is the inter-repetition interval and u_n the score for review need of a word, after the nth repetition.

As an example, g_1 is the time interval between the first and second time the user encounters a word. However a special case occurs on the last review. If the user read a word N times, then g_N would be the time interval between the Nth review (which is logically the last one) and the present moment, when the system performs the recommendation task.

[5] http://supermemo.com
[6] https://www.supermemo.com/english/ol/sm2.htm

β is the long-term memory residue, a value designed to favor words that have been read several times over a long period. The values of α and β change the slope of the "memory" curve and are fixed to $\alpha = 0.1$ and $\beta = 1$ in the case of The Vocabulometer. As Wozniak indicate, these values must be adapted based on the performance of the user's memory.

The interval of the time duration between two reminders increases exponentially. So the more a word is read, the less it needs to be reviewed. At the opposite, the less the user has read a word, the more he needs to review it.

In our context, every time a word is read, it's considered as reviewed and the value of u_n is updated. We compute a $\sum u_n$ for all the words of a document and normalize it by the number of words. We thus obtain the "reviewing score" of the document: the larger is the value, the less the user needs to review the document (and the smaller the value is, the most the user need to review the words of the document). We are finally able to suggest to the user which document should be the next to read in order to review as many words as possible in one document.

VI. Conclusion and future works

We presented the Vocabulometer, a web platform for recording the reading behavior. Such platform is easy to use and could greatly help to analyze readers and documents and providing new kind of services such as learning a new language.

The first experiments show that the designed algorithm is able to accurately detect words read by the user. The Vocabulometer can thus provide many services like reading statistics or useful recommendation systems for selecting the best document to read based on reader's vocabulary. The recommendation features need to be tested. This is a hard task because we have to take into account cognitive and perceptual factors. We are currently working on a test process based on both perceptive and objective features. After a reading session, the Vocabulometer will ask the user to answer few questions. First questions will be related to the document, in order to measure the user's comprehension. Other questions will be related to his personal feeling about the text difficulty and his language level.

Understanding better the way we read documents will help to understand better the documents themselves. We are currently starting research that aims to define automatically the reading mode of a user (superficial reading, in-depth reading, etc). State of the art models like [22], [23] need to be adapted to measure such behavior. In the future, eye-tracking with web camera will certainly be democratized thanks to the advance of artificial intelligence and deep learning approaches such as [1], [24] and will allow anyone to use eye tracking based platforms from a laptop, tablet or smartphone.

REFERENCES

[1] K. Krafka, A. Khosla, P. Kellnhofer, H. Kannan, S. Bhandarkar, W. Matusik, and A. Torralba, "Eye tracking for everyone," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2176–2184.

[2] V. Cantoni, C. Galdi, M. Nappi, M. Porta, and D. Riccio, "Gant: Gaze analysis technique for human identification," *Pattern Recognition*, vol. 48, no. 4, pp. 1027–1038, 2015.

[3] C. Holland and O. V. Komogortsev, "Biometric identification via eye movement scanpaths in reading," in *Biometrics (IJCB), 2011 International Joint Conference on.* IEEE, 2011, pp. 1–8.

[4] A. George and A. Routray, "A score level fusion method for eye movement biometrics," *Pattern Recognition Letters*, vol. 82, pp. 207–215, 2016.

[5] M. Adjouadi, A. Sesin, M. Ayala, and M. Cabrerizo, "Remote eye gaze tracking system as a computer interface for persons with severe motor disability," *Computers helping people with special needs*, pp. 628–628, 2004.

[6] P. Biswas and P. Langdon, "A new input system for disabled users involving eye gaze tracker and scanning interface," *Journal of Assistive Technologies*, vol. 5, no. 2, pp. 58–66, 2011.

[7] O. Augereau, M. Matsubara, and K. Kise, "Comic visualization on smartphones based on eye tracking," in *Proceedings of the 1st International Workshop on coMics ANalysis, Processing and Understanding*, ser. MANPU '16. New York, NY, USA: ACM, 2016, pp. 4:1–4:4.

[8] O. Palinko, A. L. Kun, A. Shyrokov, and P. Heeman, "Estimating cognitive load using remote eye tracking in a driving simulator," in *Proceedings of the 2010 symposium on eye-tracking research & applications.* ACM, 2010, pp. 141–144.

[9] J. F. Hopstaken, D. van der Linden, A. B. Bakker, M. A. Kompier, and Y. K. Leung, "Shifts in attention during mental fatigue: Evidence from subjective, behavioral, physiological, and eye-tracking data." *Journal of Experimental Psychology: Human Perception and Performance*, vol. 42, no. 6, p. 878, 2016.

[10] M. Sabatos-DeVito, S. E. Schipul, J. C. Bulluck, A. Belger, and G. T. Baranek, "Eye tracking reveals impaired attentional disengagement associated with sensory response patterns in children with autism," *Journal of autism and developmental disorders*, vol. 46, no. 4, pp. 1319–1333, 2016.

[11] R. Biedert, A. Dengel, M. Elshamy, and G. Buscher, "Towards robust gaze-based objective quality measures for text," in *Proceedings of the Symposium on Eye Tracking Research and Applications.* ACM, 2012, pp. 201–204.

[12] U. Garain, O. Pandit, O. Augereau, A. Okoso, and K. Kise, "Identification of reader specific difficult words by analyzing eye gaze and document content," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR).* IEEE, 2017, pp. 1346–1351.

[13] O. Augereau, K. Kise, and K. Hoshika, "A proposal of a document image reading-life log based on document image retrieval and eyetracking," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on.* IEEE, 2015, pp. 246–250.

[14] O. Augereau, C. Lima Sanches, K. Kise, and K. Kunze, "Wordometer systems for everyday life," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 1, no. 4, pp. 123:1–123:21, Jan. 2018. [Online]. Available: http://doi.acm.org/10.1145/3161601

[15] S. Xu, H. Jiang, and F. Lau, "User-oriented document summarization through vision-based eye-tracking," in *Proceedings of the 14th international conference on Intelligent user interfaces.* ACM, 2009, pp. 7–16.

[16] O. Augereau, M. Iwata, and K. Kise, "An overview of comics research in computer science," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR).* IEEE, 2017, pp. 54–59.

[17] G. Buscher, A. Dengel, and L. van Elst, "Eye movements as implicit relevance feedback," in *CHI'08 extended abstracts on Human factors in computing systems.* ACM, 2008, pp. 2991–2996.

[18] C. Lima Sanches, O. Augereau, and K. Kise, "Vertical error correction of eye trackers in nonrestrictive reading condition," *IPSJ Transactions on Computer Vision and Applications*, vol. 8, no. 1, p. 7, 2016.

[19] O. Augereau, H. Fujiyoshi, and K. Kise, "Towards an automated estimation of english skill via toeic score based on reading analysis," in *Pattern Recognition (ICPR), 2016 23rd International Conference on.* IEEE, 2016, pp. 1285–1290.

[20] C. Lima Sanches, K. Kise, and O. Augereau, "Japanese reading objective understanding estimation by eye gaze analysis," in *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers.* ACM, 2017, pp. 121–124.

[21] P. Wozniak and E. J. Gorzelanczyk, "Optimization of repetition spacing in the practice of learning," *Acta neurobiologiae experimentalis*, vol. 54, pp. 59–59, 1994.

[22] O. Le Meur and Z. Liu, "Saccadic model of eye movements for free-viewing condition," *Vision research*, vol. 116, pp. 152–164, 2015.

[23] R. P. Rao, G. J. Zelinsky, M. M. Hayhoe, and D. H. Ballard, "Modeling saccadic targeting in visual search," in *Advances in neural information processing systems*, 1996, pp. 830–836.

[24] C. Meng and X. Zhao, "Webcam-based eye movement analysis using CNN," *IEEE Access*, vol. 5, pp. 19 581–19 587, 2017.