Fall 2021

Approaches for Eye-Tracking While Reading

Xiaohao Sun
University of Windsor

APPROACHES FOR EYE-TRACKING WHILE READING

by

Xiaohao Sun

A Thesis
Submitted to the Faculty of Graduate Studies
through the Department of Electrical and Computer Engineering
in Partial Fulfilment of the Requirements for
the Degree of Master of Applied Science at the
University of Windsor

Windsor, Ontario, Canada

Approaches for Eye-tracking While Reading

by

Xiaohao Sun

APPROVED BY:

E. Kim

Department of Mechanical, Automotive and Materials Engineering

---

X. Chen

Department of Electrical and Computer Engineering

B. Balasingam, Advisor

Department of Electrical and Computer Engineering

August 18, 2021

Declaration of Co-Authorship / Previous Publication

Co-Authorship

I hereby declare that this thesis incorporates material that is result of joint research, as follows: Chapters 2 of this thesis were co-authored with professor Balasingam, who provided supervision and guidance during the research and writing process. In all cases, the key ideas, primary contributions, experimental designs, data analysis, interpretation, and writing were performed by the author.

I am aware of the University of Windsor Senate Policy on Authorship and I certify that I have properly acknowledged the contribution of other researchers to my thesis, and have obtained written permission from each of the co-author(s) to include the above material(s) in my thesis.

Previous Publication

Thesis chapter	Publication title/full citation	Publication status
NA	X. Sun and B. Balasingam, "Algorithms for Reading Line Classification," *IEEE International Conference on Systems, Man, and Cybernetics*, July 2021.	Accepted
2	X. Sun and B. Balasingam, "Reading Line Classification Using Eye-Trackers," *IEEE Transactions on Instrumentation and Measurement*, July 2021.	Published

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my thesis. I certify that the above material describes work completed during my registration as a graduate student at the University of Windsor.

General

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis. I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

In this thesis, we developed an algorithm to detect the correct line being read by participants. The comparisons of the reading line classification algorithms are demonstrated using eye-tracking data collected from a realistic reading experiment in front of a low-cost desktop-mounted eye-tracker. With the development of eye-tracking techniques, research begins to aim at trying to understand information from the eyes. However, state of the art in eye-tracking applications is affected by a large amount of measurement noise. Even the expensive eye-trackers still suffer significant noise. In addition, the inherent characteristics of gaze movement increase the difficulty of obtaining valuable information from gaze measurements. We first discussed an improved Kalman smoother called slip-Kalman smoother, designed to separate eye-gaze data corresponding to correct text lines and reduce measurement noise. Next, two different classifiers are applied to be trained; one is Gaussian discriminant based while the other is support vector machine based. As a result, our algorithm improved the performance of eye gaze classification in the reading scenario and beat the previous method.

Acknowledgement

Throughout my Master's duration and study, I have received so many assistance and support. First of all, I would like to especially thank my supervisor, Dr. Balakumar Balasingam, whose expertise in human-computer interaction and signal processing gave me significant support. Your instruction and patience have enabled me to make continuous progress in my research and reach a higher level. Secondly, I would like to acknowledge my committee member, Dr. Xiang Chen and Dr. Eunsik Kim, whose suggestions after my 1st seminar have inspired me greatly and made my research more rigorous and excellent. I would also like to thank my family, who selflessly supported me through my study. They were my spiritual support when I was confused and in difficulty, especially during the epidemic. Finally, I'm genuinely grateful to my girlfriend, Xinyi Yang, for her love and support. She encouraged me, never gave up on me during my Master's study, and helped me through the difficult period.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis shows the procedure and advancement in the research area of eye-gaze tracking presented by the author's already submitted and published works. Eye-tracking is essential for a wide range of technologies, and it is a hot research topic in human-computer interaction. An increasingly important application in eye-tracking is to track people's eye movements as they read, which is the focus of our research.

With the development of eye-tracking technology, more and more accurate eye trackers have been developed. For example, Tobii Pro eye tracker [1], and Pupil Core eye tracker [2] are two kinds of expensive wearable high accuracy eye tracker. However, these kinds of wearable eye trackers are expensive to be used in daily life. Moreover, wearable devices, such as glasses, are not convenient in general use. It is better to use the webcam itself to build the application. However, the shortage of a webcam is that the accuracy of the webcam to detect the eye gaze movement is poor. Instead, we choose a desktop-mounted eye tracker since this kind of eye tracker is similar to the webcam in that they both are placed in a fixed position. In our experiment, GazePoint GP3 [3] desktop-mounted eye tracker is chosen and develop more applications. Even using a desktop-mounted eye tracker, low precision eye-gaze detection is still a challenge for research because of measurement noises and the erratic nature of the human eye. In order to go further in the application related to eye tracking, it is vital to increase the performance of the low-cost eye tracker. Consequently, during my research, I spare no effort to overcome the challenge of inaccurate eye gaze detection during reading behavior and predicting which line is being read reliably.

1.1 Eye Tracking and Applications

Research into eye tracking has a long history. As early as 1737, Porterfield came up with the description of eye movements [4], which opened the door to the study of eye-tracking. Moreover, later in 1792, Wells described fast movements of eyes [5]. At present, two centuries later, more and more eye-tracking technology is appearing. Commercial eye trackers with new technologies, such as Tobii Pro eye tracker [1] and GazePoint GP3 [3], can detect eye gaze movement, pupil diameters, and blinks, etc.

With those advanced commercial eye trackers, applications in all kinds of fields are beginning to develop rapidly. The eye-tracking technique can be applied to various areas, such as Psychology, Marketing, Medical, Gaming, Automotive research, etc. In the following subsection, different research utilized eye-tracking techniques will be introduced to provide an intuition of the application of this technique.

Diefendorf and Dodge's work in 1908 [6] is the first and most well-documented in the relationship between the eye-tracking and the diagnosis of the psychology disease – schizophrenia. Not only that, in 2010, Levy et al. [7] also stated that eye-tracking dysfunction was the robust finding associated with schizophrenia and provided analytical evidence. In 2014, Garcia-Blanco [8] showed that attentional biases, which can be detected by eye-tracking technology, was a key feature of bipolar disorder. Dmitry Lagun et al. [9] utilized eye-tracking data with the machine learning algorithm to detect Alzheimer's disease in 2011. Antony and Ramaswamy [10] present the potential of using eye-tracking information to measure cognitive load in a human-computer interaction context. Eye tracking technology also related to other mental disease such as autism [11], mental disorder [12], Parkinson [13], etc.

As shown above, eye-tracking has been successfully and broadly applied to the research area of psychology these years. Furthermore, eye-tracking technology has been wildly used in marketing research [14]. For instance, Qu et al. integrated the questionnaires and eye-tracking information to evaluate the user experience on an instant message APP [15]. In 2006, Chandon et al. provided a solution about analyzing the commercial behavior of point-of-purchase by eye-tracking data [16]. The eye-tracking also played an important role in attention-based marketing [17], which is part of the attention economy.

In recent years, research in education using eye-tracking also shows a rapid development trend. Rayner et al. [18] tested the eye movements with two different contexts, one in which the full text was difficult to read and the other in which the text was clearly inconsistent. They found that the eye movements reflect the reading comprehension sensitively. Maria et al. [19] found that eye-tracking and biometric methods can be used to help teachers ascertain students' attention and then take appropriate actions to improve the teaching quality. In 2012, Meng-Jung Tsa et al. applied eye-tracking to examine the students' visual attention in a multiple-choice science problem [20]. In a recent paper, Kaakinen and Johanna [21] proposed that eye tracking can be used to study students' engagement, teachers' expertise, and student-teacher interaction. And they pointed out that eye tracking is valuable for future research in classroom context. The studies with eye-tracking in education raise the interest in eye gaze analysis in reading behavior, which will be briefly introduced in the next subsection.

1.2 Eye-tracking while reading and benchmarks

Inspired by the research about eye-tracking technology, some researchers began to dig into a more specific application of eye-tracking, which is eye-tracking while reading.

Tracking eye movement during reading has excellent potential in various fields, such as education, marketing, psychology, etc. Though the eye-trackers significantly improve accuracy, it is still not accurate enough to precisely track the reading progression, especially for the low-cost eye trackers. Some of the previous research tried to increase the performance by using a different algorithm to process the eye gaze data. One effective way to do this is to increase the accuracy of reading progression prediction by classifying the eye gaze into the text it belongs to.

For instance, an algorithm was proposed by Sanches et al. to align the gazes to the correct line [22]. They utilized the Tobii EyeX eye tracker to collect the eye fixation data while reading. After that, the vertical error between the measurements and the true line position was calculated to align the eye fixation to the correct line. The algorithm proposed in this paper beats the previous algorithm with 69% accuracy.

Another research related to reading line classification is made from our Human System Lab (HSL) by Steve Bottos and Dr. Balakumar Balasingam [23]. In this paper, they proposed a Kalman-filter based technique called slip-Kalman filter, which can detect the line return and eliminate the measurement noise. They assumed the amount of line is fixed and is already known. Then an HMM algorithm is used to classify the eye gaze data from a whole page reading procedure into corresponding lines. Their algorithm reached an accuracy of 87.43%, which is a significant improvement on this research topic.

Though the previous study has made a great contribution in eye gaze classification while reading, the accuracy still has much space for improvement. Furthermore, Steve's study has assumed the prior knowledge of the text, such as the position and amount of lines, is know to the line detection system, which made the algorithm not general enough. Thus, I would like to dedicate myself to this research area and contribute to improving the performance and making the algorithm more general.

1.3 Organization of the Thesis

In this thesis, we choose to use the manuscript format instead of the traditional format. Chapter 1, 3 provides the context that has not been published or submitted for publication. This chapter introduces the background of eye-tracking technology, motivation of our research, and state of art. Chapter 2 is a written paper published in the journal "IEEE Transactions on Instrumentation and Measurement." This paper well reflects the work and contributions I have made in my Master's study. Then Chapter 3 will conclude our contribution and discuss future research in eye tracking while reading.

1.4 Bibliography

[1] "Tobii pro eye tracker website." https://www.tobiipro.com. Accessed: 2021-21-07. pages 1, 2

[2] "Pupil core eye tracker website." https://pupil-labs.com/products/core/. Accessed: 2021-21-07. pages 1

[3] "Gazepoint eye tracker website." https://www.gazept.com/. Accessed: 2020-31-05. pages 1, 2, 10, 26

[4] W. Porterfield, "An essay concerning the motions of our eyes. part i. of their external motions," *Edinburgh medical essays and observations*, vol. 3, no. 1737, pp. 160–263, 1737. pages 2

[5] W. C. Wells, *An Essay Upon Single Vision with Two Eyes: Together with Experiments and Observations on Several Other Subjects in Optics. By William Charles Wells, MD.* T. Cadell, in the Strand, 1792. pages 2

[6] A. R. Diefendorf and R. Dodge, "An experimental study of the ocular reactions of the insane from photographic records.," *Brain*, vol. 31, no. 3, pp. 451–489, 1908. pages 2

[7] D. L. Levy, A. B. Sereno, D. C. Gooding, and G. A. O'Driscoll, "Eye tracking dysfunction in schizophrenia: characterization and pathophysiology," *Behavioral Neurobiology of Schizophrenia and Its Treatment*, pp. 311–347, 2010. pages 2

[8] A. García-Blanco, L. Salmerón, M. Perea, and L. Livianos, "Attentional biases toward emotional images in the different episodes of bipolar disorder: an eye-tracking study," *Psychiatry research*, vol. 215, no. 3, pp. 628–633, 2014. pages 2

[9] D. Lagun, C. Manzanares, S. M. Zola, E. A. Buffalo, and E. Agichtein, "Detecting cognitive impairment by eye movement analysis using automatic classification algorithms," *Journal of neuroscience methods*, vol. 201, no. 1, pp. 196–203, 2011. pages 2

[10] A. W. Joseph and R. Murugesh, "Potential eye tracking metrics and indicators to measure cognitive load in human-computer interaction research," *Journal of Scientific Research*, vol. 64, no. 1, 2020. pages 2

[11] S. Wang, M. Jiang, X. M. Duchesne, E. A. Laugeson, D. P. Kennedy, R. Adolphs, and Q. Zhao, "Atypical visual saliency in autism spectrum disorder quantified through model-based eye tracking," *Neuron*, vol. 88, no. 3, pp. 604–616, 2015. pages 3

[12] M. Vidal, J. Turner, A. Bulling, and H. Gellersen, "Wearable eye tracking for mental health monitoring," *Computer Communications*, vol. 35, no. 11, pp. 1306–1311, 2012. pages 3

[13] C. A. Kuechenmeister, P. H. Linton, T. V. Mueller, and H. B. White, "Eye tracking in relation to age, sex, and illness," *Archives of General Psychiatry*, vol. 34, no. 5, pp. 578–579, 1977. pages 3

[14] M. Wedel and R. Pieters, "A review of eye-tracking research in marketing," *Review of marketing research*, pp. 123–147, 2017. pages 3

[15] Q.-X. Qu, L. Zhang, W.-Y. Chao, and V. Duffy, "User experience design based on eye-tracking technology: a case study on smartphone apps," in *Advances in applied digital human modeling and simulation*, pp. 303–315, Springer, 2017. pages 3

[16] P. Chandon, J. Hutchinson, E. Bradlow, and S. H. Young, "Measuring the value of point-of-purchase marketing with commercial eye-tracking data," *INSEAD Business School Research Paper*, no. 2007/22, 2006. pages 3

[17] J. L. Orquin and M. Wedel, "Contributions to attention based marketing: Foundations, insights, and challenges," 2020. pages 3

[18] K. Rayner, K. H. Chace, T. J. Slattery, and J. Ashby, "Eye movements as reflections of comprehension processes in reading," *Scientific studies of reading*, vol. 10, no. 3, pp. 241–255, 2006. pages 3

[19] M. Villa, M. Gofman, S. Mitra, A. Almadan, A. Krishnan, and A. Rattani, "A survey of biometric and machine learning methods for tracking students' attention and engagement," in *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 948–955, IEEE, 2020. pages 3

[20] M.-J. Tsai, H.-T. Hou, M.-L. Lai, W.-Y. Liu, and F.-Y. Yang, "Visual attention for solving multiple-choice science problem: An eye-tracking analysis," *Computers & Education*, vol. 58, no. 1, pp. 375–385, 2012. pages 3

[21] J. K. Kaakinen, “What can eye movements tell us about visual perception processes in classroom contexts? commentary on a special issue,” *Educational Psychology Review*, vol. 33, no. 1, pp. 169–179, 2021. pages 3

[22] C. L. 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, pp. 1–7, 2016. pages 4, 14

[23] S. Bottos and B. Balasingam, “Tracking the progression of reading using eyegaze point measurements and hidden markov models,” *IEEE Transactions on Instrumentation and Measurement*, 2020. pages xii, 4, 11, 12, 14, 26, 29, 30, 31, 32

Chapter 2

Reading Line Classification Using Eye-Trackers

2.1 Introduction

Eye-trackers are emerging as a new form of measuring systems that estimate several physiological features of a person, such as eye-gaze position and pupil diameter [1]. Eye-trackers used to be costly and bulky with limited applications in laboratories. With recent advances in video sensors and embedded computing systems, low-cost, desktop-mounted eye-trackers are being experimented for a wide variety of applications. However, improving the accuracy and performance of desktop-mounted eye-trackers is an active research domain. In [2], a video-based eye gaze tracking method was proposed to precisely detect the corneal reflection light and pupil. In order to get more reliable measurements, [3–5] provide several solutions under the scope of calibration. In addition, a head movement detection system is proposed in [6] in order to eliminate the effects of head movement and improve eye-tracking accuracy.

With the development of low cost eye-tracking technology [7], research into eye-tracking has become commonplace in wide range of applications, such as, advanced driver assistance systems [8], human computer automation [9], robotics [10], market research [11], eye gestures detection [12], psychology research [13], etc. With the advances in sensing technology, computing, connectivity, and storage, it is now possible to collect eye-tracking data at negligible cost in a wide variety of situations. Research into eye-tracking has seen exponential growth in the past two decades. In the same period, the accuracy of low-cost eye-trackers has also seen significant improvement [7]. Reading line classification is one of the futuristic research areas of eye-tracking that has applications in such fields as reading comprehension analysis [14], reading interest analysis [15], and online education [16].

Typical eye-tracker measures several parameters of the eye and its movement, such as gaze locations, gaze fixations, pupil parameters, and eye-blink information. Particularly, eye-gaze measurements [17] are of interest in this chapter. Eye-gaze points have information about the mental state of the subject [18] as well as it contains information about the visual interaction of the subject with their surrounding environment [19]. Using the eye-tracker data, it is possible to know more about the interests and experience of the subject, e.g., interests while reading a book. However, there are significant challenges faced by today's eye-trackers — particularly when it comes to tracking the progression of reading — from the fact that the low-cost eye-trackers have meager resolution compared to what is required. Consequently, there is significant research interest related to analyzing eye-tracking while reading. In Section 2.2, we will briefly go through these recent studies.

The purpose of this chapter is to track the progression of reading, especially the line being read, utilizing the data collected by the Gazepoint GP3 desktop mounted eye-tracker [20] when participants read text on a computer screen without any prior knowledge about the content. The eventual goal of developing this technology is to help understand the level of interaction a person had with a text passage. Such technology will be useful to classify the reading pattern into known categories [21], e.g., reading, skimming, and scanning. Further, an accurate reading line classification algorithm can have more potential applications, such as online training and education, retail research, marketing, psychological studies, cognitive research, and testing of reading comprehension [22]. It is also important to note that some languages are read from left to right while others are read from right to left. The line classification algorithm presented in this chapter is applicable to work under both scenarios.

The contributions of this chapter are as follows:

a) *A slip-Kalman smoother (slip-KS) approach* to pre-process the eye-gaze data so as to reduce the measurement noise and to decouple the gaze data corresponding to different lines. The proposed approach is found to improve the line detection performance compared to a recent prior work reported in [23, 24].

b) *Application of Gaussian discriminant classifier for line detection while reading.* this chapter introduces the Gaussian discriminant approach, for the first time, to line detection problem. Gaussian discriminant is a robust approach to classification. However, eye-gaze data, while reading, are scattered across the page, partly due to the nature of the eye-gaze and partly due to the measurement noise of the eye-tracker. The preprocessing approach in this chapter, using the Kalman smoother, enabled effective application of the Gaussian discriminant classifier and resulted in robust and computationally efficient approach to line detection system.

c) *A novel feature selection strategy for line detection algorithms.* The new feature selection strategy was tested with support vector machine (SVM) as the classifier. The new strategy used information from both x and y coordinates, and selected feature evenly from the entire line. The resulting performance is found to be superior to existing approaches.

d) *Validation using realistic data.* The performance of the proposed approaches was evaluated using data collected from seven participants.

The proposed approaches in this chapter are significant for the following reasons:

- The commercial (particularly low-cost) eye-trackers come with significant measurement noise [25] both due to limitations of the sensors as well as the limitations of the state-of-the-art eye-gaze processing algorithms. The proposed approaches in this chapter could be used by future eye-tracker developers to improve their outputs. Also, the proposed approaches can be useful in webcam-based eye-gaze tracking applications [26] where the measurement noise is significant.

- When a person reads, the eye-gaze might wander into the previous or subsequent lines — this does not mean the person switched lines; rather, it is a feature of the human gaze; this feature needs to be filtered out to get the correct picture of the reading pattern. The proposed approaches in this chapter are designed to address this.

- Previous research [23] on the same problem assumed that the number of lines on a page and the coordinate of each line are known as prior knowledge. These two assumptions are relaxed in the present paper, i.e., the numbers of lines and their locations are learned during the training phase. Which means that, the lines of text does not need to be equally spaced.

The remainder of this chapter is organized as follows: Section 2.2 briefly introduces recent studies about eye-gaze tracking analysis while reading. Section 2.3 describes the proposed approach in this chapter. Section 2.4 presents the results of the proposed approach using real world data. Finally, this chapter is concluded in Section 2.5.

2.2 Related Work

In this section, a review of recent studies on reading procedure analysis via eye-tracking technology is presented.

2.2.1 Applications of Eye-tracking while Reading

The main research area about eye-tracking analysis while reading is reading comprehension. In [27], L. Copeland et. al. proposed a new performance function for training feed-forward neural networks to predict reading comprehension based on eye gaze data. Also, Sanches et al. [28] extracted some specific eye-gaze features and used support vector regression model to predict participants' percentage of correct answers about the contents in the text. Similarly, another approach measures participants' comprehension ability by calculating reader ranking based on correct answer lines recorded by eye gaze tracker and the number of correct answers given by participants [29]. In addition to reading comprehension, some other research domains also received significant attention to eye-tracking while reading. In [30], supervised machine learning techniques were used to detect mind wandering probes from eye gaze and context features to analyze which features had the most significant impact. Moreover, Lara-Alvarez et al. [31] introduced an approach called *counting words from lines (COWL)*, which deals with the imprecision problem by associating the eye-tracking data with points obtained from character recognition.

Various research applications of eye-tracking while reading are still at the early stage due to the fact that accurate enough eye-gaze tracking — that is sensitive enough to detect the line being read — still requires a separate hardware. This might rapidly change when the eye-gaze tracking technology based on built in webcams matures [32]. Most present studies directly use the raw eye-gaze data from the eye-tracker, which has noise and disturbance caused by the device and participants [33]. As opposed to many existing approaches that use raw eye-gaze data to extract statistical information of the reading behavior, the present paper focuses on improving the eye-tracking technology itself while reading.

2.2.2 Tracking Accuracy Improvement

Only a few prior works proposed preprocessing techniques to eye-gaze data. In [34], an approach was proposed to correct vertical error of the gaze fixations and alignment of estimated eye-gazes with text resulting in improved accuracy of line classification. However, this approach only considered fixations and vertical error — in the eye gaze context, the gaze position's distribution has more information. In a recent work, Bottos et al. proposed a hidden Markov model (HMM) based line detection system (LDS) [23]. The proposed LDS was demonstrated using data collected from one participant where the lines of text spanned the entire page (page width), and the reading always occurred line-by-line without any repetitive reading of lines. The proposed HMM-based LDS also assumed the number of lines as well as their locations in a page. In order to be widely applicable, an LDS needs to be able to track without requiring to know the number of lines on the screen and their exact locations, because a user can easily move the location of a text (e.g., an e-book) on a computer screen.

2.3 Proposed Approach

In this section, a new approach for real-time line classification is presented. Figure 2.1 shows the workflow diagram of the proposed approach. First, the raw eye-gaze data is sent through the slip-Kalman filter (slip-KF) [23] which separates data corresponding to different lines. The decouple data, i.e., the data belonging to each line is then sent through a Kalman Smoother (KS), which is employed to reduce the variance in the filtered data. Hereafter, the above two steps, the slip-KF followed by the KS, is referred to as the slip-KS. The slip-KS output is then rescaled to span the entire text region. After the above preprocessing, data from few pages are used to train the classifiers. The parameters from the classifier output are used for line classification.

The remainder of this section will detail each block in Figure 2.1.

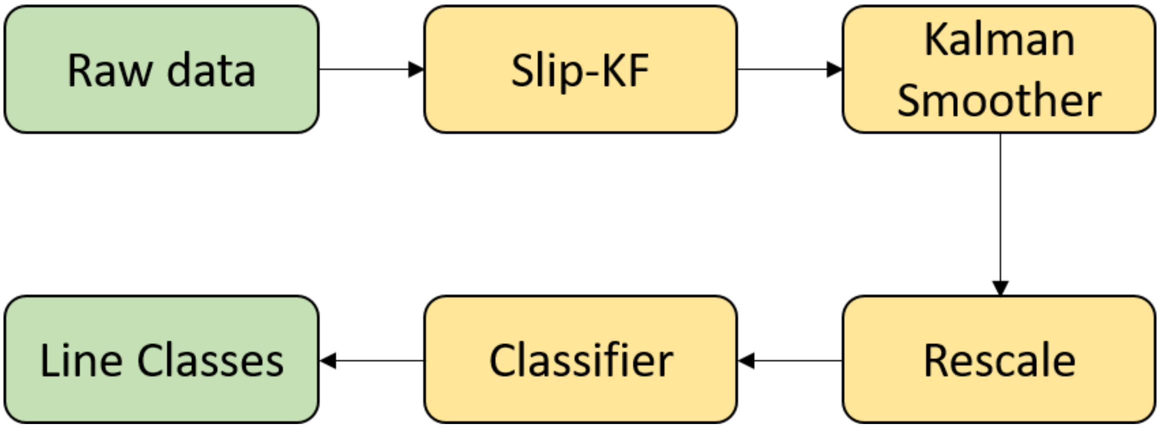


Figure 2.1: **Procedure of whole experiment** This is the diagram of the whole Classification approach.

2.3.1 Regular Kalman Filter

Let us consider the following 4×1 state vector

$$\mathbf{x}(k) = [x(k) \quad \dot{x}(k) \quad y(k) \quad \dot{y}(k)]^T \tag{2.1}$$

where $x(k)$ is the x-coordinate of the eye-gaze at time step k, $y(k)$ is the y-coordinate of the eye-gaze at time step k, $\dot{x}(k)$ is the velocity of the eye-gaze movement in the x direction at time step k , and $\dot{y}(k)$ is the velocity of the eye-gaze movement in the x direction at time step k. The *process model* [35], based on the state vector introduced in (2.1) is written as

$$\mathbf{x}(k+1) = \mathbf{F}\mathbf{x}(k) + \mathbf{\Gamma}\mathbf{v}(k) \tag{2.2}$$

where vector $\mathbf{v}(k)$ is a 4×1 *zero-mean Gaussian noise with unity standard deviation*,

$$\mathbf{F} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{\Gamma} = \begin{bmatrix} \frac{1}{2}\Delta T^2 & 0 \\ \Delta T & 0 \\ 0 & \frac{1}{2}\Delta T^2 \\ 0 & \Delta T \end{bmatrix} \begin{bmatrix} q_x & 0 \\ 0 & q_y \end{bmatrix} \tag{2.3}$$

q_x and q_y are the parameters of the process noise covariance $\mathbf{Q}$, which can be shown as [35]

$$\mathbf{Q} = E\left\{\mathbf{\Gamma}\mathbf{v}(k)\mathbf{v}(k)^T\mathbf{\Gamma}^T\right\} \tag{2.4}$$

$$= \begin{bmatrix} \frac{1}{4}\Delta T^4 q_x & \frac{1}{2}\Delta T^3 q_x & 0 & 0 \\ \frac{1}{2}\Delta T^3 q_x & \Delta T^2 q_x & 0 & 0 \\ 0 & 0 & \frac{1}{4}\Delta T^4 q_y & \frac{1}{2}\Delta T^3 q_y \\ 0 & 0 & \frac{1}{2}\Delta T^3 q_y & \Delta T^2 q_y \end{bmatrix} \tag{2.5}$$

After the *process model*, the *measurement model* of reading can be built as follows

$$\mathbf{z}(k) = \mathbf{H}\mathbf{x}(k) + \mathbf{w}(k) \tag{2.6}$$

where

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.7}$$

and

$$\mathbf{z}(k) = [z_x(k) \quad z_y(k)]^T \tag{2.8}$$

is the eye-gaze measurements, which are the noisy observations of $x(k)$ and $y(k)$. The 2×1 vector $\mathbf{w}(k)$ indicates measurement noise.

After the definition of the state-space model comprising of (2.2) and (2.6), the KF can be summarized through the diagram shown in Figure 2.2, where $\hat{\mathbf{x}}(k|k)$ is the estimated state vector at time k and $\mathbf{P}(k|k)$ is the state covariance matrix ($k = 1, \ldots, N$, where N is the total number of gaze points). For more details of the parameters and initializations please refer to [24].

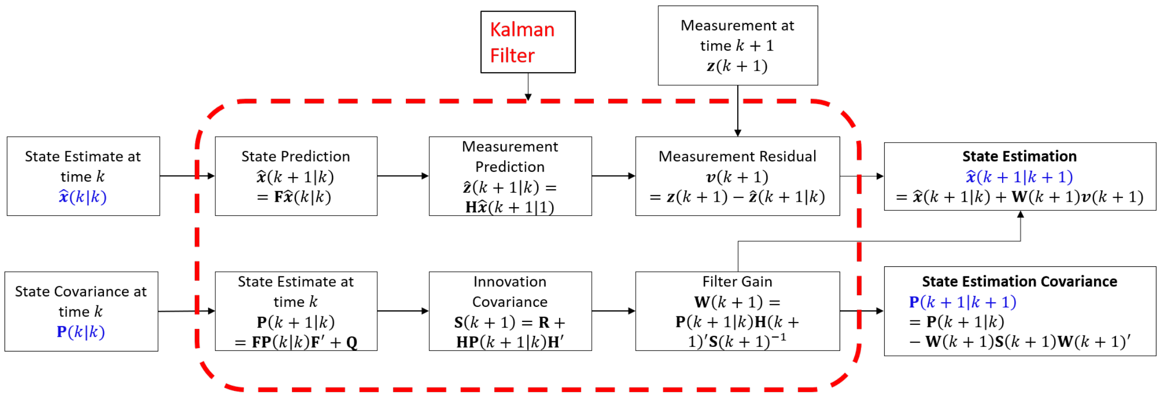


Figure 2.2: **Regular Kalman filter.** Regular KF can be used to filter eye-gaze data by selecting appropriate values for the process and measurement noise as it is described in Section 2.3.1. However, provided that the eye-gaze is engaged in reading, a slip-KF (see Figure 2.5) will improve the eye-gaze tracking performance while reading.

Figure 2.3 shows the performance of the regular KF, where the red dots indicates the measurements $\mathbf{z}(k)$ and the blue lines are the estimate from KF. From Figure 2.3, it can be seen that the KF was able to *follow* the reading progress. However, the KF estimates overshoot when the reading progress returns to a new line — this can be used to detect line returns [24].

2.3.2 Slip Kalman Smoother

It was demonstrated in [24], that the slip-KF can be used to decouple the data belonging to each line and remove the overshoot. The slip-KF is inspired by slip gear [36], detecting the reading line return when a specific threshold is exceeded. So the problem transfers into finding a proper indicator to detect the line return while

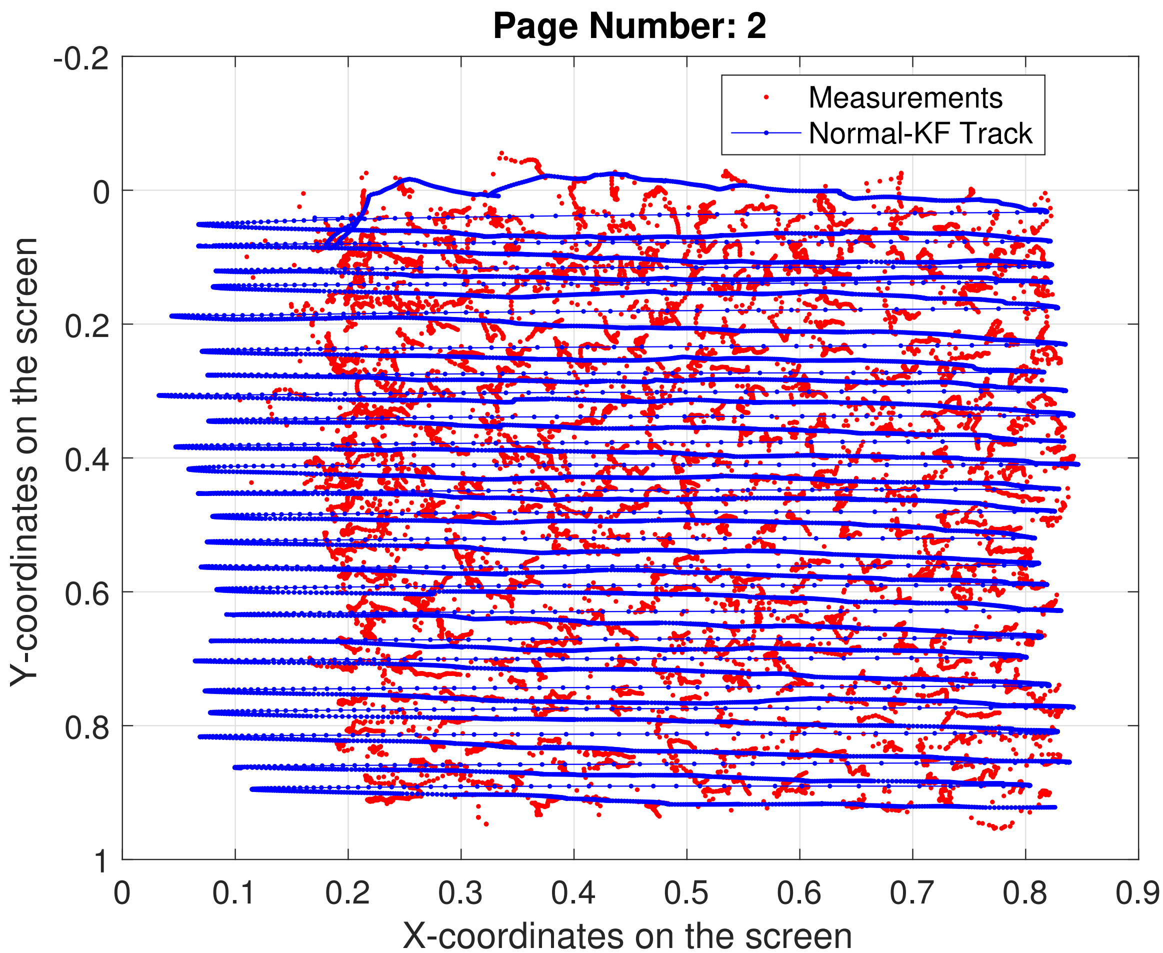


Figure 2.3: **Tracking eye-gaze while reading using regular Kalman filter.** The problem with the regular KF is illustrated as the backtrack over-estimation on the left side.

reading.

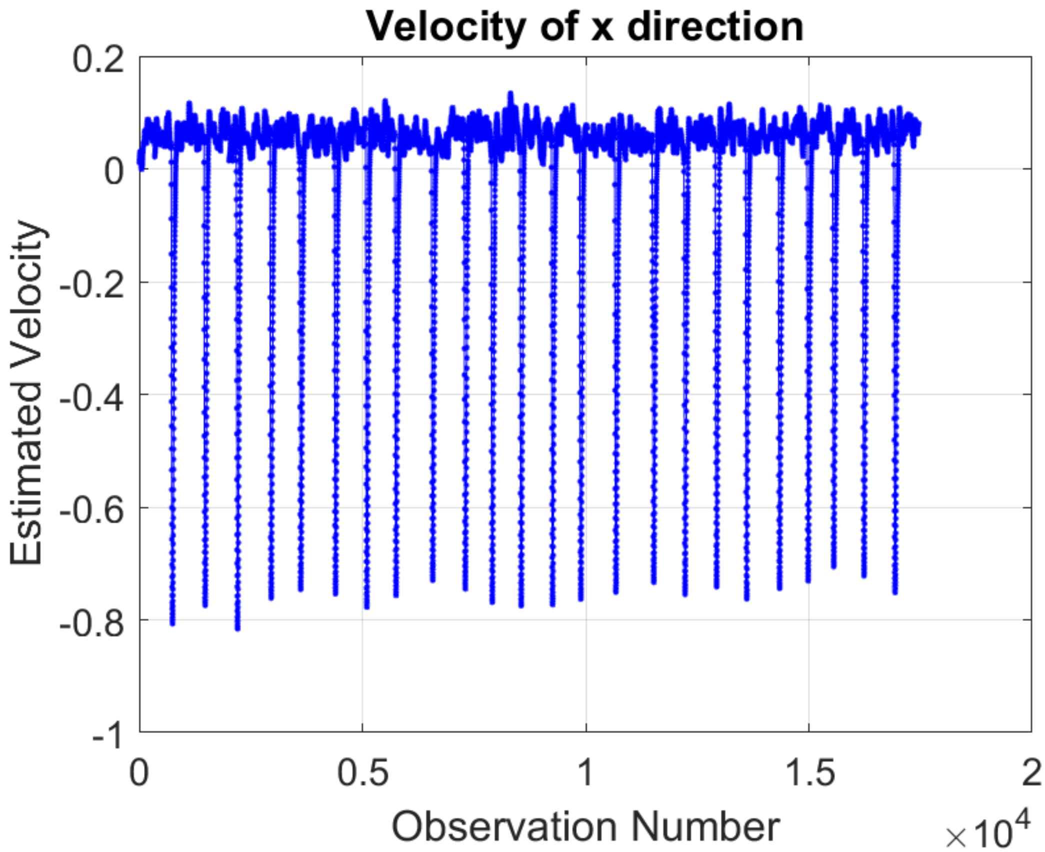


Figure 2.4: **Velocity $\dot{x}(k)$ (text-width)/sec of regular KF.** Velocity of the x direction is a good indicator since the line return can be distinguished very clearly when the velocity goes to a relatively large negative value.

From Figure 2.4, the dramatic change in velocity of the x direction $\dot{x}(k)$ can be clearly seen. As a consequence, the slip-KF chooses $\dot{x}(k)$ to be the indicator of line return. The final slip-KF can be summarized as follows: When running the regular KF, detect the unusual changes in estimated velocity $\hat{\dot{x}}(k)$ in time step k. If $\hat{\dot{x}}(k) < v_{th}$, re-initialize the state vector $\hat{\mathbf{x}}(k+1|k+1)$ of next time step $k+1$,

$$\hat{\mathbf{x}}(k+1|k+1) = [z_x(k+1), v_0, z_y(k+1), \hat{\dot{y}}(k|k)]^T$$

where v_{th} is a threshold on the x-velocity, and v_0 is the initial x-velocity that usually is a predetermined value for different types of reading. The state covariance matrix $\mathbf{P}(k+1|k+1)$ re-initializes to the initial state $\mathbf{P}(0|0)$. The remaining operations of the slip-KF are the same as the regular KF. Figure 2.5 shows a block diagram of one recursive step of the slip-KF.

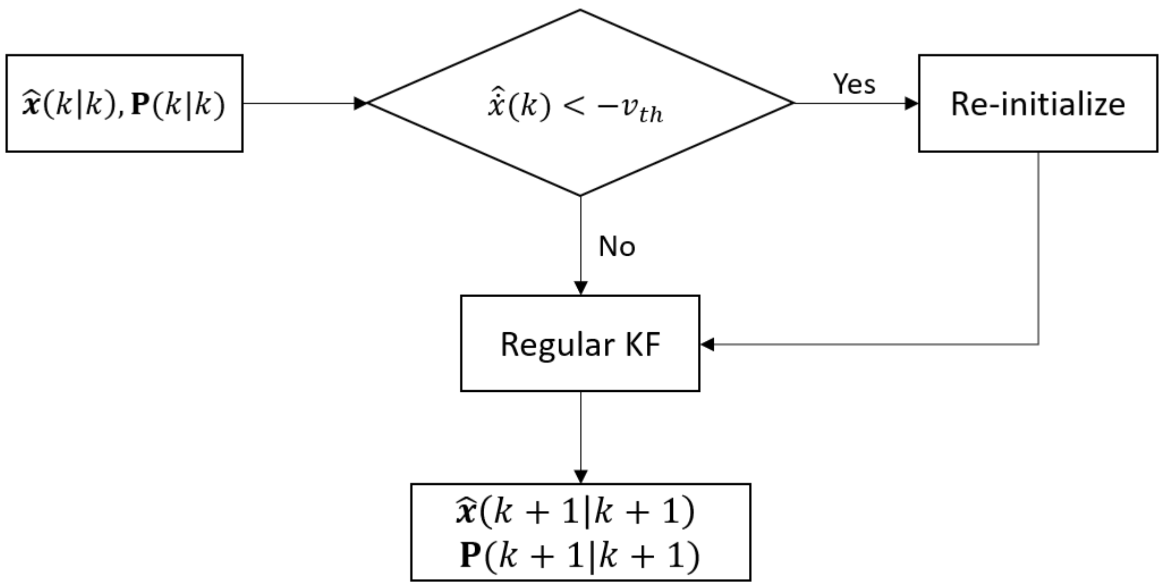
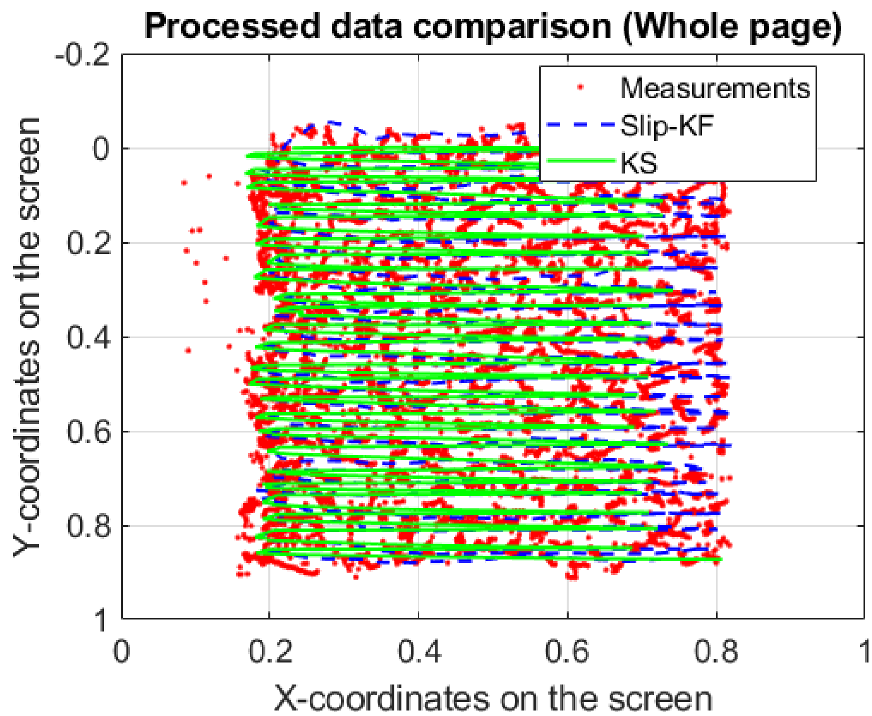


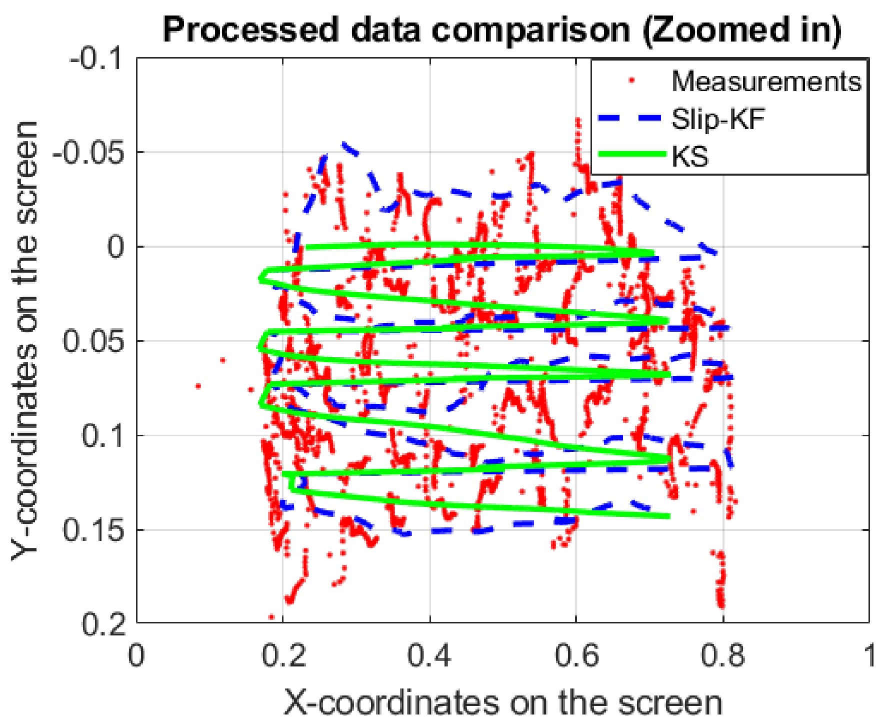
Figure 2.5: **Slip Kalman filter.** The slip-KF is suitable for eye-gaze tracking while reading. v_{th} is the general expression of the velocity threshold.

The slip-KF approach described above is re-initialized at the start of every new line to preset values. This would eliminate the over-shot estimates shown in Figure 2.3. However, the output of the slip-KF is still observed to contain high variance. Thus, the slip-KS is introduced to further reduce the variance in the estimates. After slip Kalman filtering, the entire eye-gaze data is decoupled into different lines. The slip-KF estimates (i.e., $\hat{\mathbf{x}}(k|k)$ and the covariance matrix $\mathbf{P}(k|k)$) corresponding to one line are then sent through a KS. This process comprises the slip-KS step. For more details about KS, the reader is referred to [35].

Figure 2.6 shows the output of KS compared to that of the KF for few lines; the smoothing results of the KS can be clearly observed in this figure. Figure 2.7 shows the performance comparison between the two filters more clearly; the KS provides



(a) Entire page



(b) Zoomed version

Figure 2.6: **Eye-gaze tracking while reading smoothed by KS.** The previous slip-KF is able to follow the eye-gaze measurements along the line. However, the y-coordinate still have some disturbance which will have influence on the proposed classification algorithm. In this figure, the KS can elimate more noise on y-coordinate than the previous slip-KF.

smoother gaze position estimates compared to the KF. Also, as shown in Figure 2.8, the smoothed data from KS distinguishes the lines much more clearly.

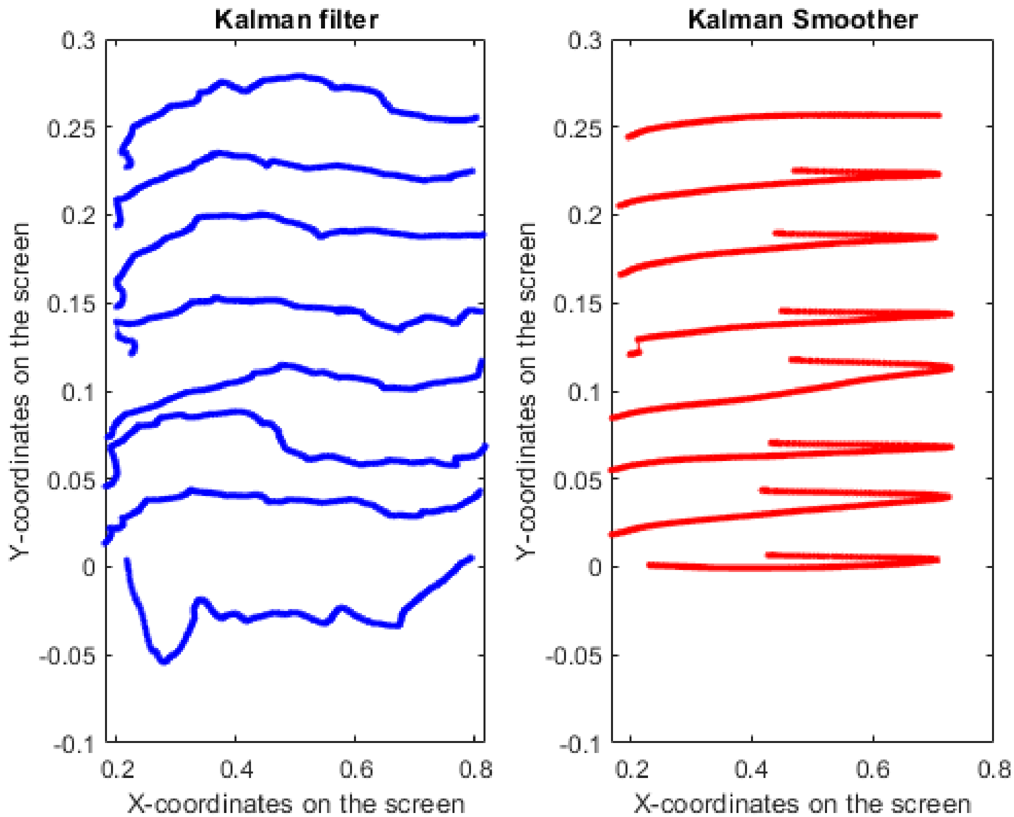


Figure 2.7: **Comparison between KF and KS results.** In contrast, KS can more effectively reduce the noise than KF. After smoothing, the process of reading becomes clearer.

2.3.3 Classification Approach

In this subsection, we present the line classification algorithm. We tested two different classifiers: one is the proposed Gaussian discriminant classifier, and the other is SVM-based classifier. We will discuss each of these approaches in the remainder part of this subsection.

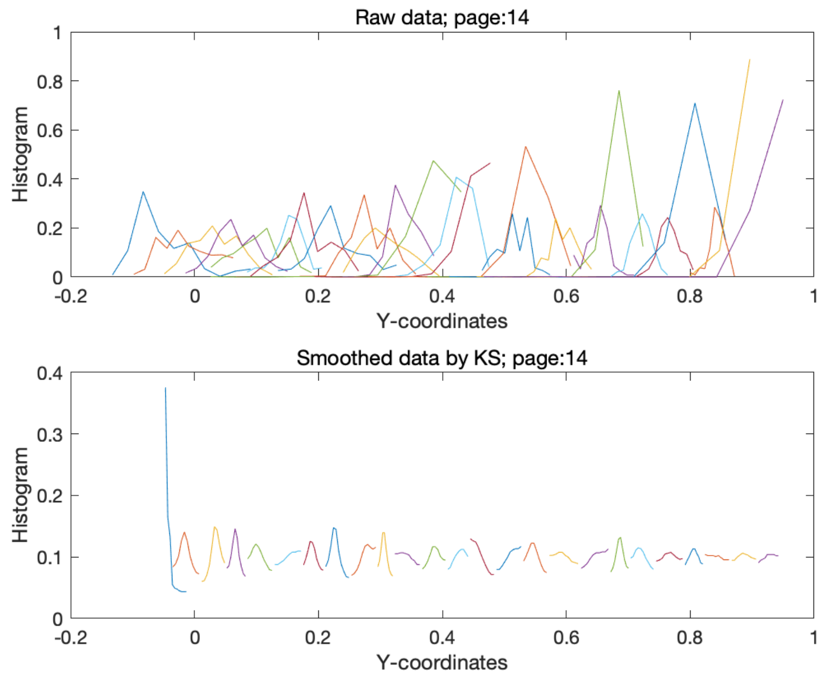
2.3.3.1 Gaussian Discriminant Classification

Figure 2.8(a) shows the y-coordinates of the gaze measurements as well as the estimated y-coordinates using the KS. The data corresponding to each line is shown in different colors. It can be observed that the KS outputs are much more concentrated in each line than the raw observations. In other words, the KS outputs are separated well according to each line.

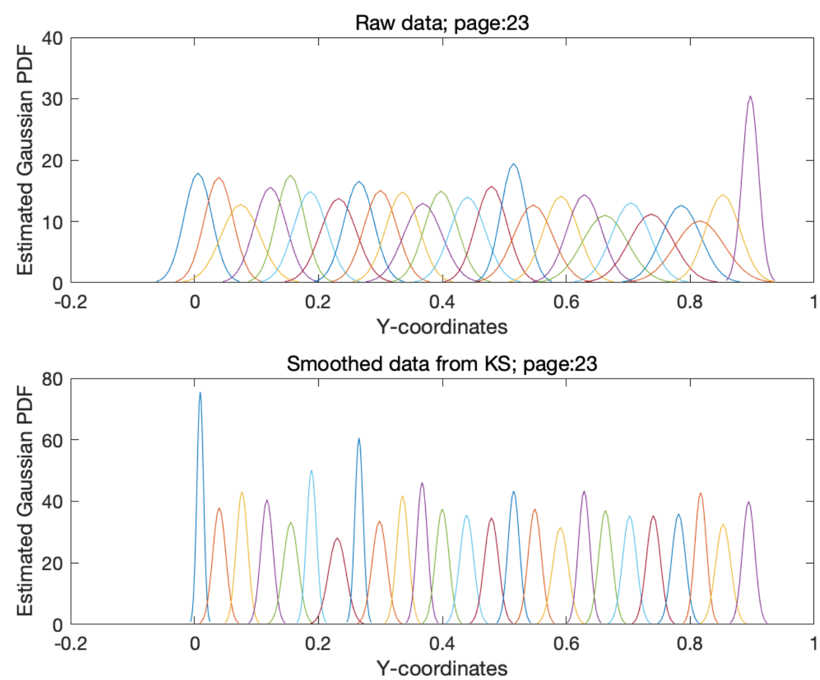
Another observation from Figure 2.8(a) is that the y-coordinates are distributed according to a Gaussian-like distribution. This observation is then used to approximate data corresponding to each line as a Gaussian distribution. Figure 2.8(b) shows the Gaussians computed for each line from page 23 of the collected data — the top-plot shows the Gaussian distribution with parameters computed from the raw data, and the second plot shows the same with parameters computed from the Kalman smoothed data. The parameters of this Gaussian distribution will be used in the classification algorithm.

In order to evaluate our proposed approach without loss of generality, the leave-one-out cross-validation (LOOCV) [37] approach was used. Assume we had N pages of eye-gaze data while reading, then extract page j where $j \in \{1, 2, ..., N\}$ to be the test set and pages $1, 2, ..., j-1, j+1, ..., N$ were used as the training set. With this assumption, the parameter of line i can be written as (μ_i, σ_i) where μ_i is the mean of the line i y-coordinates from pages $1, 2, \ldots j-1,\ j+1, \ldots, N$ and σ_i is the standard deviation of line i's y-coordinates from pages $1, 2, \ldots j-1,\ j+1, \ldots, N$.

Now, once all N sets of parameters $(\mu_i, \sigma_i,\ i = 1, 2, \ldots, 25)$ are computed it is ready to classify the lines in page j. The objective of the classification is as follows: given the y-coordinate of a gaze point $\hat{y}(k|n)$ the log-likelihood that the observation



(a) Histogram of one page



(b) Gaussian distribution of one page

Figure 2.8: **Distribution of y-coordinate.** Both figure (a) and (b) show the distribution of different lines of one page. It can be seen from the histogram plot that the distribution of y-coordinate is close to the Gaussian distribution. So the guassian distirbution of classification is considered.

might have been generated form line i is computed as follows

$$
\begin{aligned}
d_i &= \log\left(\frac{1}{\sqrt{2\pi}\sigma_i}\mathrm{e}^{-\frac{(\hat{y}(k|n)-\mu_i)^2}{2\sigma_i^2}}\right) \\
&= \log\left(\frac{1}{\sqrt{2\pi}\sigma_i}\right) - \frac{(\hat{y}(k|n)-\mu_i)^2}{2\sigma_i^2}
\end{aligned}
\tag{2.9}
$$

where d_i denotes the likelihood that the gaze data $\hat{y}(k|n)$ comes from line i. The estimated line corresponding to the k^{th} observation is then given as

$$
\text{class}(k) = \arg\max_i d_i \quad i = 1, 2, \ldots, 25 \tag{2.10}
$$

In other words, this chapter transfers this problem into a multi-classe Gaussian discriminant analysis [38]. Algorithm 1 is how the classification approach, log-likelyhood (LLH) classifier, works. However, due to the differences in the head position of participants and other external factors of the eye tracker, each page's distribution is not the same. For instance, the mean of different pages' first line is different, and the ranges of different pages are also different. In order to overcome this distribution, the rescale approach plays a vital role. Before the classification, the y-coordinate of every page has to be rescaled into the same y-coordinate region, which is $[0, 1]$, for data preprocessing.

Algorithm 1 (Gaussian-based classifier)

Input: N pages training data
Output: estimated line **class** of test data

1: Select page/participant j to be the test set and others be the training set. ($j = 1, \ldots, N$)
2: Use the training data to calculate the mean y-coordiante $\boldsymbol{\mu}(i)$ and standard deviation $\boldsymbol{\sigma}(i)$ of every line
3: **for** k = 1:length(test data) **do**
$\mathbf{LLH}(k) = -\frac{(\hat{y}(k)-\boldsymbol{\mu})^2}{2\boldsymbol{\sigma}^2}$
$[\sim, \mathbf{class}(k)] = \max(\mathbf{LLH}(k))$
4: **end for**

The classification approach, explained so far for each point, can be simply extended to a *line-by-line classification* approach as follows: Since the slip-KF records the line return while reading, it is easy to split the measurements into different lines. After splitting the measurements, all the measurements are labeled with a pre-estimated line number. Then the Gaussian-based classifier is used to predict the line classes for every measurement and a majority voting is applied to assign a line number for each separated line.

2.3.3.2 SVM Classification

The second type of classifier is implemented using SVM. A built-in multiclass-SVM classifier available in Matlab is used for this. The SVM classifier required an equal number of data points from each line, whereas the number of measurements can differ based on a person's reading behavior. Hence, it is important to choose a proper feature selection approach. In this chapter, we utilize the x-coordinate to implement a feature selection approach to pick the most representative data from each estimated line. We divide each estimated line data into 40 equal length bins by x-coordinate and randomly extract ten measurements from each bin. Then assemble the measurements selected from different bins in order from left to right. Each line will have 400 measurements' y-coordinate as its feature. Finally, the features are fed to train the SVM model. The SVM-based approach also employs LOOCV to guarantee generality.

2.4 Evaluation and Results

2.4.1 Data Collection Procedures

In this section, two datasets are used to evaluate our proposed algorithm. Firstly, we use the data from [23] to demonstrate the proposed method, which has 25 pages of data from one participant (a male in his twenties). Eye gaze data were collected utilizing Gazepoint GP3 [20] a desktop mounted eye-tracker at a sampling rate of 60

Hz. Text used to be read is a passage taken from a publicly available copy of Moby Dick by Herman Melville. As shown in the left part of Figure 2.9, the eye-tracker is placed at the bottom of the computer screen (a 1920×1080 PC screen) and the recorded position of eye-gaze data while a person reads one page of text consisting of 25 lines is shown on the right. After the eye-tracker was set up, the participant sat in front of the screen and went through a calibration process for the eye-tracker. The eye-tracker software offers a 9-point calibration and 5-point one; the participant was asked to select the former. During the 9-point calibration, the eye-tracker interface displays bright circle markers at nine random positions on the screen and the participant was asked to look at them. If the calibration fails (this happens when the participant does't track the displayed markers), the participant will be prompted to redo the calibration step. Once the calibration is done, the participant remained seated at the same position and proceed to the reading step. During the reading, the participant was required to press the space key to reveal a single text line. Each keystroke will only show one line, starting from line 1 at the top of the screen and ending at line 25, and will hide the previous line. In this way, every time the participant finished reading one line, he pressed the space key to display the next line — this allowed to record the ground truth. The x and y coordinate of the eye-gaze measurements will be used as the input raw data to do the anlaysis.

In order to further validate the algorithms, we collected more data from six additional participants in the same experimental setup as above. These six participants (5 males and 1 female all in their twenties) are Chinese international students at University of Windsor. This time, the participants read only ten pages. Then, we combined these six data with the first ten pages of the prior data. Eventually, we had two datasets, the first dataset had one participant reading 25 pages and the second one had seven participants reading 10 pages each, respectively.

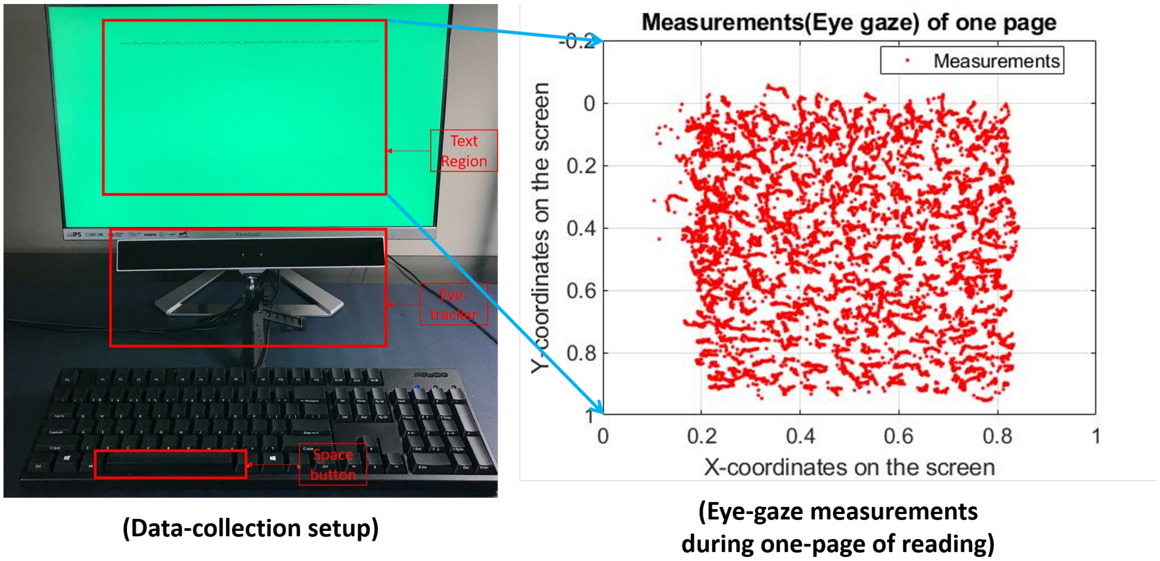


Figure 2.9: **Experiment Setup.** Participants will sit in front of the monitor where the text for reading will be displayed in full-screen mode (shown on the left). This geometry causes a vertical measurement bias that is discussed in Section 2.4. The distribution of one page collected eye-gaze measurements is shown on the right.

2.4.2 Evaluation Metrics

The classification error for page p is defined as

$$e_p \quad = \quad \left[\frac{\text{count}_{t=1}^{t=N_p}\left(\hat{S}_p(t) \neq S_p(t)\right)}{N_p} \right] \times 100 \tag{2.11}$$

where $\hat{S}_p(t)$ denotes the estimated line number, $S_p(t)$ denotes the true line number (ground truth), and N_p denotes the number of data points from page p. The average error across all pages, e_{avg} was then computed as,

$$e_{avg} \quad = \quad \frac{\sum_{p=1}^{N} e_p}{N} \tag{2.12}$$

where N denotes the number of pages. The classification accuracy is then defined as

$$\text{Line Detection Accuracy (in \%)} = 100 - e_p \tag{2.13}$$

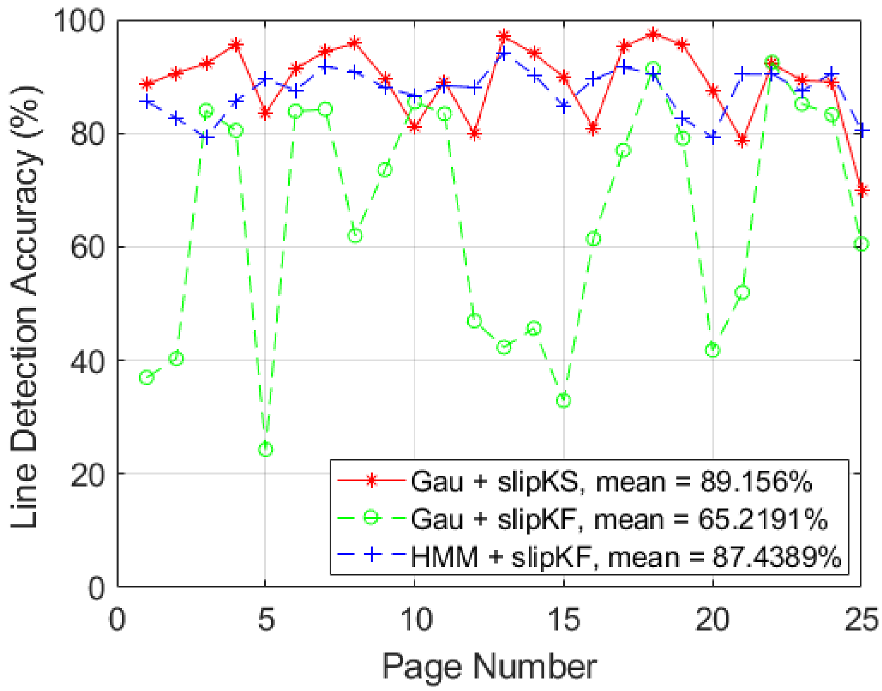
In the remainder of this section, the classification accuracy of various approaches are presented.

2.4.3 Classification Results

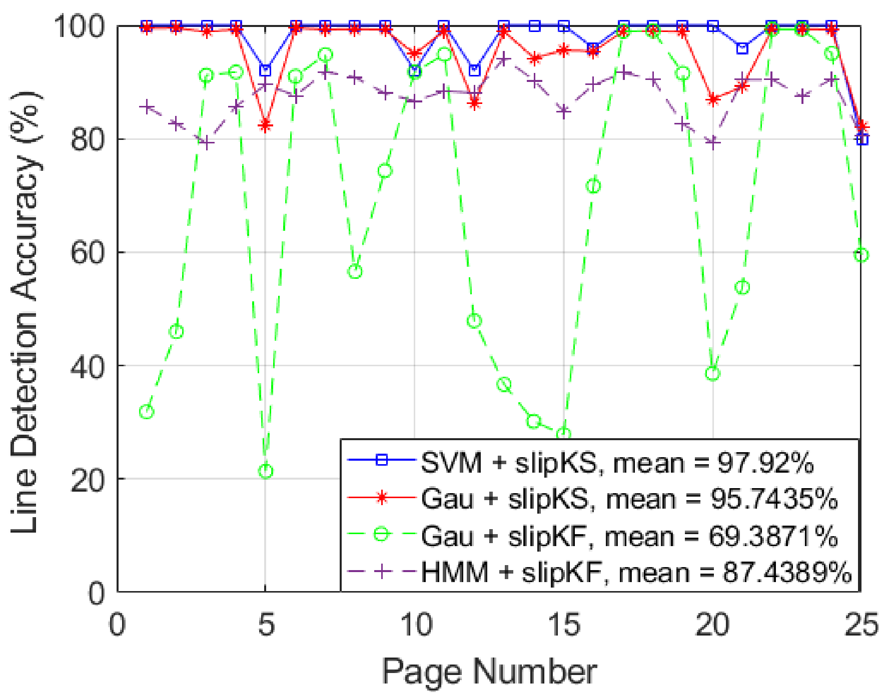
Figure 2.10(a) shows the prediction accuracy of the proposed classification approach. The line detection accuracy is computed by averaging the percentage of correct classification from all 25 pages. For comparison, the line detection from Slip-KF based estimates and Slip-KS based estimates are shown in the same plot. Further, the performance of the HMM-based line classifier [23] is also shown on the same plot. It can be noticed that the proposed Gaussian classification approach combined with slip-KS is better in performance than the previous HMM approach. Further, the proposed approach has the advantage that it is able to instantly classify a given gaze point measurement after training the classifier, whereas the HMM approach, as it is implemented in [23], requires data from the entire page before the (Viterbi based) classification algorithm can start.

Figure 2.10(b) shows the prediction accuracy of the *line-by-line classification* approach. From this figure, the Gaussian-based classifier and the SVM-based classifier on the slip-KS approaches both outperform the HMM approach.

It can be observed that the performance of the Gaussian classifier, just with the KF outputs, is very poor. It is due to the fact that the performance of the Kalman filter is in such a way that the variance (particularly, in the y-direction) is not that reduced after filtering. This can be observed from Figure 2.7. When the smoothing is applied, the variance significantly reduced, resulting in reduced overlap of the line distribution and much-improved classification performance.



(a) Point-by-point Accuracy



(b) Line-by-line Accuracy

Figure 2.10: **Classification Accuracy.** In (a), the proposed point-by-point (or measurement-by-measurement) line classification is compared with that of the HMM approach in [23]. In (b), different line-by-line classification accuracies are compared.

Figure 2.11 shows the averaged accuracy of the proposed classification scheme on a line-by-line basis. It can be noticed that, except for the first line, the line classification accuracy increases from top lines towards the bottom ones. The placement of the eye-tracker (at the bottom of the screen) influences the above pattern in accuracy. The outlier in the first line is probably due to classification bias for initial line observation.

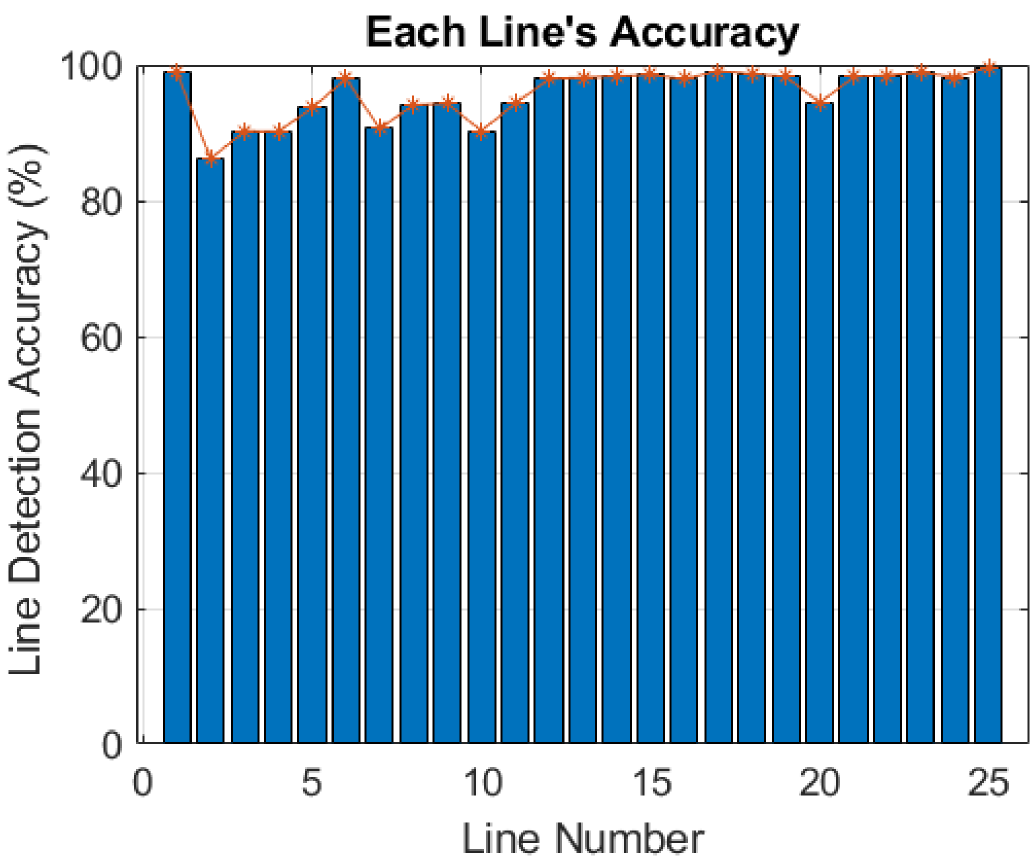


Figure 2.11: **Demonstration of eye-tracker bias.** The eye-tracker is placed at the bottom of the screen (see Figure 2.9) as such the classification accuracy can be seen to increase with the line number (top line on the screen is denoted 1 and the line number increases from top to bottom).

Table 2.1 shows the performance comparison of different algorithms analyzed in this chapter. In order to compare the performance against the previous approach in [23], the same dataset as in [23] was used for Table 2.1. Compared to the classification based on slip-KF, the slip-KS based classifier has a much higher mean accuracy and

Table 2.1: **Accuracy comparison**

	Mean	Standard Deviation
SVM + slip-KS (line-by-line)	97.92%	4.63%
Gaussian + slip-KS (line-by-line)	95.74%	5.68%
Gaussian + slip-KS (point-by-pont)	89.15%	6.79%
Gaussian + slip-KF (line-by-line)	69.38%	27.73%
Gaussian + slip-KF (point-by-pont)	65.21%	21.15%
HMM (page-by-page)	87.43%	4.03%

a smaller standard deviation which shows the benefit of data preprocessing using the slip-KS compared to that of slip-KF. Another advantage of the proposed approach is that the data does not need to be in a specific order, which means the accuracy will not be affected if the observation is measured while reading an arbitrary line of text on the screen. The average accuracy of the point-by-point classification based on KS is up to 89.15%. Moreover, for the ling-by-line classification approach, the average accuracy is much higher than the point-by-point, which is up to 95.74%, much higher than the HMM approach proposed in [23]. Last but not least, the SVM based approach reach the highest accuracy 97.92% with the standard deviation 4.63%.

The data collection procedure described in Section 2.4.1 is repeated for six additional participants with the exception that, this time, each participant read 10 pages. Figure 2.12 shows the comparison of classification accuracy for all seven participants. Initial experimentation with data from all seven subjects showed great variation in performance. As a result, it was decided to combine two adjacent lines together before the classification analysis. It is suspected that the observed variation might have been based on whether a participant is a native speaker who is fluent in English or not. This needs to be analyzed in detail using a separate study and is left as future work. As for the eye-tracking while reading, the Gaussian discriminant classifier shows a higher accuracy of 79.3% compared to the SVM accuracy at 72%. The SlipKS approach proved to be an important component for both classifiers.

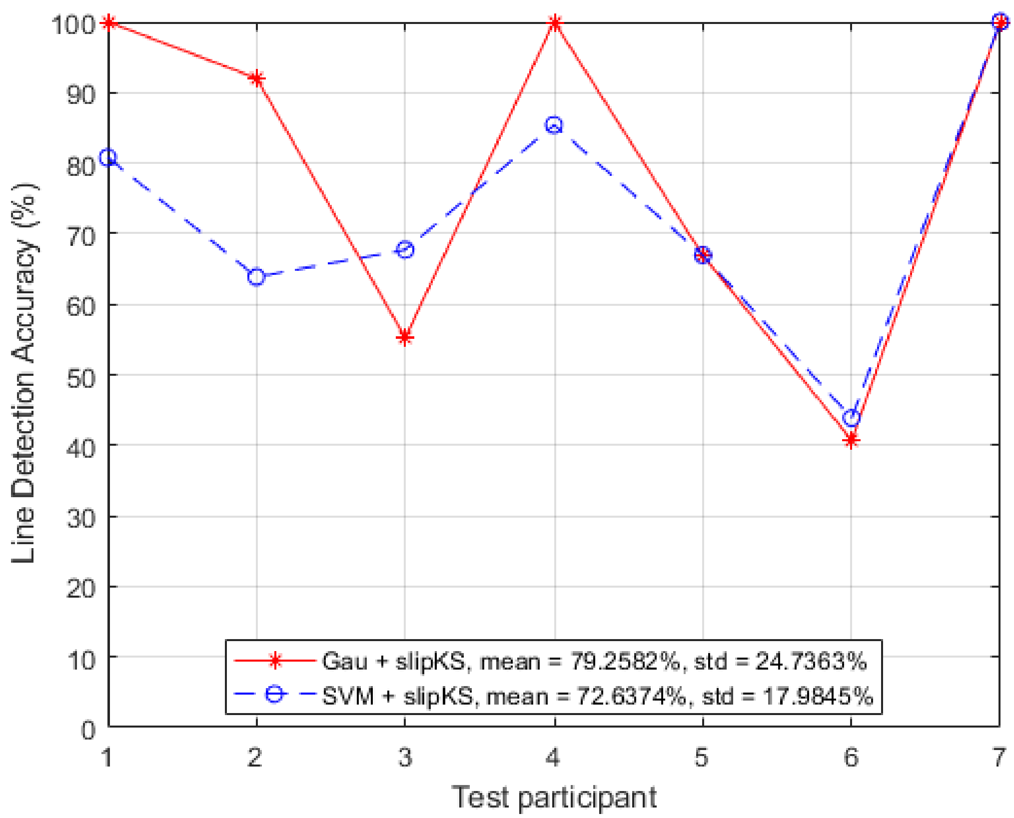


Figure 2.12: **Classification performance for all seven participants.**

2.4.4 Real-time analysis

In this subsection, some details are provided to understand the different algorithms analyzed in this chapter for their performance for real-time implementation.

It is reported that the average reading speed of a native English speaker is 4.06 words per second; for fast readers, the reading rate is 7.66 words per second [39]. In order to be used in real life applications, the response time of the reading line classification algorithm should be less than the time it takes to finish reading a line. In the experimental setup of this chapter, there were 150 characters per line. According to statistics provided in [40], the average length of words in English is 4.7 characters. Based on these, the baseline latency of the experiment is defined as

$$\begin{aligned} T(\text{baseline}) &= \frac{\text{characters per line}}{\text{length of words} \times \text{reading rate}} \\ &= \frac{150}{4.7 \times 4.06} = 7.86 \text{ sec.} \end{aligned}$$

For each algorithm, the latency is computed as

$$T(\text{algorithm}) = T_p + T_f + T_c \tag{2.14}$$

where T_p is the pre-processing time (SlipKS), T_f is the time taken for feature extraction (SVM only), and T_c is the time taken for classification. Table 2.2 shows the baseline latencies as well as the latencies of each algorithm considered in this chapter on a personal computer (i5-8500 CPU). It can be noticed that, latencies of all the approaches considered in this chapter are less than that of the two baselines. Further, the Gaussian based classifiers show significantly low latency compared to other classification approaches.

Table 2.2: **Latency analysis (Line-based)**

Classification approach	**Average Latency (seconds)**
T(baseline) - normal reading	7.86
T(baseline) - fast reading	4.166
T(SVM+ slip-KS (line-by-line))	1.844
T(Gaussian + slip-KS (line-by-line))	0.034
T(Gaussian + slip-KF (line-by-line))	0.033

2.5 Conclusions and Discussions

In this chapter, a novel slip Kalman smoother (slip-KS) based classification approach was developed to predict line being read, on a real-time, point-by-point or line-by-line basis, while reading. The proposed approach is presented as a novel slip-KS, which can detect the line returns and obtain less-variant measurements corresponding to each line being read. With the Kalman smoothed estimates, two different classification approaches, one based on the support vector machines (SVM) and the other bases on Gaussian discriminants, are employed to classify smoothed gaze observations into possible text lines. Both of the proposed classification approaches can be used to predict the line being read in real-time. Both the SVM classifier and the Gaussian discriminant classifier show significant improvement in prediction accuracy compared to the previous approach, and the SVM classifier performs best on the line-by-line classification. However, the disadvantage of the SVM classifier is that it can not work in a point-by-point fashion. Both of the proposed approaches are less reliant on the continuity of lines while reading, making them more attractive in practical applications.

The present work opens several avenues for future research. Testing the proposed algorithm using data from multiple and diverse participants (e.g., different ages, different educational backgrounds, gender, different native languages) is one of them. Further, the proposed algorithm needs to be tested on more realistic reading settings (different head/body positions of the participants, different number of lines per page, multiple columns, etc.). Finally, the proposed approach needs to be demonstrated using built in webcam of a computer for suitability in a wide ranging applications.

2.6 Bibliography

[1] A. T. Duchowski, "A breadth-first survey of eye-tracking applications," *Behavior Research Methods, Instruments, & Computers*, vol. 34, no. 4, pp. 455–470, 2002. pages 9

[2] Y. Ebisawa, "Improved video-based eye-gaze detection method," *IEEE Transactions on instrumentation and measurement*, vol. 47, no. 4, pp. 948–955, 1998. pages 9

[3] Z. Ramdane-Cherif and A. NaÏt-AliNait-Ali, "An adaptive algorithm for eye-gaze-tracking-device calibration," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 4, pp. 716–723, 2008. pages 9

[4] M. B. Ahmad, M. A. Raja, M. W. Asif, K. Khurshid, *et al.*, "i-Riter: Machine learning based novel eye tracking and calibration," in *2018 IEEE International Instrumentation and Measurement Technology Conference*, pp. 1–5, IEEE, 2018. pages 9

[5] J. Chi, Z. Yang, G. Zhang, T. Liu, and Z. Wang, "A Novel Multi-Camera Global Calibration Method for Gaze Tracking System," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 5, pp. 2093–2104, 2019. pages 9

[6] Y. Ebisawa, "A pilot study on ultrasonic sensor-based measurement of head movement," *IEEE transactions on Instrumentation and Measurement*, vol. 51, no. 5, pp. 1109–1115, 2002. pages 9

[7] D. Li, J. Babcock, and D. J. Parkhurst, "openEyes: a low-cost head-mounted eye-tracking solution," in *Proceedings of the 2006 symposium on Eye tracking research & applications*, pp. 95–100, 2006. pages 9, 10

[8] A. Shaout, D. Colella, and S. Awad, "Advanced driver assistance systems-past, present and future," in *2011 Seventh International Computer Engineering Conference*, pp. 72–82, IEEE, 2011. pages 9

[9] R. J. Jacob and K. S. Karn, "Eye tracking in human-computer interaction and usability research: Ready to deliver the promises," in *The mind's eye*, pp. 573–605, Elsevier, 2003. pages 9

[10] M. Bhuiyan, V. Ampornaramveth, S.-y. Muto, and H. Ueno, "On tracking of eye for human-robot interface," *International Journal of Robotics and Automation*, vol. 19, no. 1, pp. 42–54, 2004. pages 9

[11] M. Wedel and R. Pieters, *Eye tracking for visual marketing.* Now Publishers Inc, 2008. pages 10

[12] B. O'Bard and K. George, "Classification of eye gestures using machine learning for use in embedded switch controller," in *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, pp. 1–6, IEEE, 2018. pages 10

[13] P. J. Rosa, F. Esteves, and P. Arriaga, "Beyond traditional clinical measurements for screening fears and phobias," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 12, pp. 3396–3404, 2015. pages 10

[14] P. A. Carpenter and M. A. Just, "Reading comprehension as eyes see it," *Cognitive processes in comprehension*, pp. 109–139, 1977. pages 10

[15] R. Biedert, G. Buscher, and A. Dengel, "The eyebook–using eye tracking to enhance the reading experience," *Informatik-Spektrum*, vol. 33, no. 3, pp. 272–281, 2010. pages 10

[16] J. Holsanova, H. Rahm, and K. Holmqvist, "Entry points and reading paths on newspaper spreads: comparing a semiotic analysis with eye-tracking measurements," *Visual communication*, vol. 5, no. 1, pp. 65–93, 2006. pages 10

[17] C. H. Morimoto and M. R. Mimica, "Eye gaze tracking techniques for interactive applications," *Computer vision and image understanding*, vol. 98, no. 1, pp. 4–24, 2005. pages 10

[18] C. I. Hooker, K. A. Paller, D. R. Gitelman, T. B. Parrish, M.-M. Mesulam, and P. J. Reber, "Brain networks for analyzing eye gaze," *Cognitive Brain Research*, vol. 17, no. 2, pp. 406–418, 2003. pages 10

[19] P. C. Ellsworth and L. M. Ludwig, "Visual behavior in social interaction," *Journal of Communication*, vol. 22, no. 4, pp. 375–403, 1972. pages 10

[20] "Gazepoint eye tracker website." https://www.gazept.com/. Accessed: 2020-31-05. pages 1, 2, 10, 26

[21] S. S. Mozaffari, F. Raue, S. D. Hassanzadeh, S. Agne, S. S. Bukhari, and A. Dengel, "Reading Type Classification based on Generative Models and Bidirectional Long Short-Term Memory," 2018. pages 10

[22] M. Horsley, M. Eliot, B. A. Knight, and R. Reilly, *Current trends in eye tracking research*. Springer, 2013. pages 11

[23] S. Bottos and B. Balasingam, "Tracking the progression of reading using eye-gaze point measurements and hidden markov models," *IEEE Transactions on Instrumentation and Measurement*, 2020. pages xii, 4, 11, 12, 14, 26, 29, 30, 31, 32

[24] S. Bottos and B. Balasingam, "A Novel Slip-Kalman Filter to Track the Progression of Reading Through Eye-Gaze Measurements," *arXiv preprint arXiv:1907.07232*, 2019. pages 11, 17

[25] P. Mannaru, B. Balasingam, K. Pattipati, C. Sibley, and J. T. Coyne, "Performance evaluation of the gazepoint GP3 eye tracking device based on pupil dilation," in *International Conference on Augmented Cognition*, pp. 166–175, Springer, 2017. pages 12

[26] K. Semmelmann and S. Weigelt, "Online webcam-based eye tracking in cognitive science: A first look," *Behavior Research Methods*, vol. 50, no. 2, pp. 451–465, 2018. pages 12

[27] L. Copeland, T. Gedeon, and S. Mendis, "Fuzzy output error as the performance function for training artificial neural networks to predict reading comprehension from eye gaze," in *International Conference on Neural Information Processing*, pp. 586–593, Springer, 2014. pages 13

[28] C. L. 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*, pp. 121–124, 2017. pages 13

[29] C. Singh, D. Yadav, and J. Lee, "Reader comprehension ranking by monitoring eye gaze using eye tracker," *International Journal of Intelligent Systems Technologies and Applications*, vol. 13, no. 4, pp. 294–307, 2014. pages 13

[30] R. Bixler and S. D'Mello, "Automatic gaze-based user-independent detection of mind wandering during computerized reading," *User Modeling and User-Adapted Interaction*, vol. 26, no. 1, pp. 33–68, 2016. pages 13

[31] C. Lara-Alvarez, T. Reyes, and H. Rodríguez-Rangel, "Counting the number of words and lines read by fusing eye tracking and character recognition data: A bayes factor approach," *Journal of Intelligent & Fuzzy Systems*, vol. 34, no. 5, pp. 3147–3154, 2018. pages 13

[32] W. Sewell and O. Komogortsev, "Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network," in *CHI'10 Extended Abstracts on Human Factors in Computing Systems*, pp. 3739–3744, 2010. pages 13

[33] A. Poole and L. J. Ball, "Eye tracking in HCI and usability research," in *Encyclopedia of human computer interaction*, pp. 211–219, IGI Global, 2006. pages 13

[34] C. L. 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, pp. 1–7, 2016. pages 4, 14

[35] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004. pages 15, 16, 20

[36] J. W. Hughes, "Slip-gear.," Feb. 22 1910. US Patent 950,290. pages 17

[37] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012. pages 23

[38] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006. pages 25

[39] M. Dyson and M. Haselgrove, "The effects of reading speed and reading patterns on the understanding of text read from screen," *Journal of research in reading*, vol. 23, no. 2, pp. 210–223, 2000. pages 34

[40] "English letter frequency counts: Mayzner revisited or etaoin srhldcu." http://norvig.com/mayzner.html. Accessed: 2021-19-05. pages 34

Chapter 3

Thesis Conclusion

To sum up, I have presented two different real-time classification approach to classify the eye gaze point to the correct line being read. These two classification approaches are based on the novel preprocessing algorithm – slip-KS, one of the most valuable contributions to this thesis. Firstly, the participants' eye gaze measurements while reading the given text were collected by Gazepoint GP3 eye tracker. Secondly, the slip-KS is proposed to track the line return and smooth the eye gaze data. After that, the smoothed data is fed to the classifier to classify the data to the corresponding line. Finally, the newly proposed SVM-based algorithm can reach 97.92% accuracy on the old data and 72.63% on the new data from seven different participants. The Gaussian-based algorithm has 95.74% on the old data and 79.25% on the new data. Both of these two kinds of classifier beats the previous approach on classification performance. It is also worth mentioning that both algorithms are satisfied with real-time requirements, which is significant for daily application development.

There are several limitations of my master's research. First of all, the amount of participants in the experiment is not large enough. And not all participants were native English speakers. Future research needs to examine the differences in reading pattern between native and non-native speakers. It is also worth to mention, different people have different reading pattern, which will lead the distribution of the gaze data to be different. More participants needs to be involved in our experiment to evaluate the robustness of our algorithm. What's more, the reading line classification algorithm is only suitable for the line-by-line reading scenario, but it provides good ideas for future research. The algorithm needs to be improved to a more general reading pattern – including reading, skimming, scanning, and reading back.

For further research, the algorithm should be developed for the web-cam-based situation instead of using professional eye-trackers. Once the web-cam-based device can detect the line being read, many applications can be used for general use. In addition, with the development of the neural network, the research for reading line prediction can try to build a specific neural network for this kind of task, which can change the pipeline proposed in this paper into an end-to-end algorithm and make it better to develop and reproduce.

Vita Auctoris

NAME:	Xiaohao Sun
PLACE OF BIRTH:	Zibo, Shandong, China
YEAR OF BIRTH:	1994
EDUCATION:	University of Electronic Science and Technology of China, Chengdu, China 2013-2017, Bachelor of Science Mathematical and Physics Basic Science University of Windsor, Windsor, Ontario 2019-2021, Master of Applied Science Electrical Engineering