Tracking of Eye Movement Features for Individualized Assessment of Neurocognitive State Using Mobile Devices

by

Hsin-Yu Lai

B.S., National Taiwan University (2014)

S.M., Massachusetts Institute of Technology (2016)

Submitted to

the Department of Electrical Engineering and Computer Science

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2021

Author .....

Department of Electrical Engineering and Computer Science

June 30, 2021

Certified by .....

Thomas Heldt

Associate Professor of Electrical and Biomedical Engineering

Thesis Supervisor

Certified by .....

Vivienne Sze

Associate Professor of Electrical Engineering and Computer Science

Thesis Supervisor

Accepted by .....

Professor Leslie A. Kolodziejski

Professor of Electrical Engineering and Computer Science

Chair, Department Committee on Graduate Students

Tracking of Eye Movement Features for Individualized Assessment of Neurocognitive State Using Mobile Devices

by

Hsin-Yu Lai

Submitted to the Department of Electrical Engineering and Computer Science
on June 30, 2021, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

The ability to objectively track neurocognitive state is very important in a wide variety of settings and conditions. For example, with current clinical techniques, it is difficult to assess a patient's neurodegenerative disease (e.g., Alzheimer's) state accurately and frequently. The most widely used tests are qualitative, variable and only performed intermittently, exposing the need for quantitative, accurate, and non-obtrusive metrics to track disease progression. Clinical studies have shown that saccade latency (an eye movement measure of reaction time) and error rate (the proportion of eye movements towards the wrong direction) are significantly affected by neurocognitive states. We propose a novel system that measures and tracks these features outside of the clinical environment using videos recorded with a mobile device. It is challenging to attain this goal, given variable environments and the absence of infrared illumination, high-speed cameras, and chinrests.

Several steps are taken to overcome these challenges and therefore enable tracking of eye movement features in large cohorts of subjects. We designed an app to guide subjects to record their eye movements at a proper distance in a well-lit environment. By enabling large-scale data collection, we have collected over 6,800 videos from 80 subjects across the adult age spectrum, which are about two orders of magnitude more videos than in most previous literature. To measure eye-movement features from these video recordings, we used a deep convolutional neural network for gaze estimation and model-based methods to measure saccade latency and error rates. With the frequent measurements of these features, we then designed an individualized longitudinal model using a Gaussian process that learns individual characteristics and the correlations across these eye-movement features. With a system that can measure eye-movement features on a much finer timescale in a broader population than previously available, our research opens up the possibility to understand whether eye-movement features can be used to help track neurocognitive states more frequently and accurately.

Thesis Supervisor: Thomas Heldt
Title: Associate Professor of Electrical and Biomedical Engineering

Thesis Supervisor: Vivienne Sze
Title: Associate Professor of Electrical Engineering and Computer Science

Acknowledgments

"She fights with semi-folded arms,

Her strong bag, and the stiff

Frost of her face (that challenges "When" and "If".)

And altogether she does Rather Well."

— Weaponed Woman by Gwendolyn Brooks

The journey of Ph.D. was filled with battlefields. I fought against my imposter syndrome. I fought against discrimination. I fought against my frustration about getting stuck in research. But luckily, I never needed to fight alone.

This thesis was completed with love and support.

I want to thank my advisors. I remembered I was trying to find a new advisor after getting my master degree. Several professors had already turned me down. And there I was, sitting in Vivienne's office and listening to Vivienne and Thomas talking about this very cool project. I said yes, because the project was impactful and interdisciplinary. I love learning, I love challenges, and I love thinking that I can be making a difference.

Three months later, they said they were willing to take me. They introduced me to Gladynel, my dear collaborator and dear friend, and Professor Charlie Sodini, who was interested in this work and ended up serving as my thesis committee. Together, we started on this journey of exploring the unknown. I remembered I told myself, "nobody had believed in me. I am not going to take this for granted."

Nothing should be taken for granted, definitely not Thomas' compassion. He cared about me and was willing to provide guidance not only on research, but also on presentation, career advice, and personal growth. I remembered after my first group meeting presentation, where I failed to keep anyone engaged, Thomas spent an hour going through my presentation slides by slides. Instead of focusing on all my mistakes, he focused on how I could improve. When I was in the last stretch of my Ph.D. completion, we had several exchanges about career advice. He has spent

so many hours sharing his experience, guiding me through the writing of research statement, revising my cover letters, listening to my practice job talk. I was worried that I became a burden, but he kept reassuring me that he enjoyed mentoring his students. I cannot say how thankful I felt and how lucky I was to have Thomas as my advisor.

Vivienne taught me all kinds of communication skills. To begin with, Vivienne helped me in seeing the big picture in my research. She asked me a lot of "so what?" questions. I often gave an answer that was too narrow of a focus. Not surprisingly, it would then be followed by another "so what?" It was like Socratic questioning, instructing us to see the real impact of our work. She also trained me to explain things to people outside of my field. My work is interdisciplinary. To make an impact, I need to explain my work to signal-processing people, statisticians, but more importantly, neurophysiologists. Vivienne's group worked on interdisciplinary fields. By practicing my presentation in her group meetings, I gradually learned how to explain an idea without an equation and how to explain an equation with intuitions. In addition, I also tended to be overly careful in what I said. Therefore, for statements that I cannot guarantee to be 100% true, I tended to add in "I believe" or "I hope" or "I think." Vivienne would always point those out, asking me to provide more authority.

Charlie was like my third advisor. He came to our weekly meetings. He was great at pointing out confusing statements and missing details. It often happened that when Gladynel, Vivienne, Thomas, and I were in a debate, Charlie would be sitting there saying, "this is what I think can be helpful to me." And the heated debate would be quenched by his wisdom.

I was deeply blessed to have worked with Gladynel. She was extremely caring and responsible. We had a lot in common. We both came from a different culture, are non-native speakers, experienced similar cultural expectations, faced similar self-doubt, and tried to learn new fields for this project. As a result, she was extremely understanding of my struggles. She also carried me a lot throughout this Ph.D. When I felt down or was too busy with my classes, she covered me for those weeks of presentations. She gave me a notebook inscribed with "Nevertheless She Persisted"

as a birthday gift. I have since written several incidents where I persisted thanks to the inspiration. Last but not least, we had so much fun together. We went to gyms together. We had coffee together. We went kayaking together. We made dessert for our labmates together. My Ph.D. experience was so much happier with her understanding and companion.

I was glad to have worked with so many kind people throughout this project. I want to thank Peter Kamm, our app developer. He is super knowledgeable and extremely patient. I was worried that I would be too demanding, but he was very understanding in fulfilling our frequent requests. Jingpeng Zhou was the first student that I mentored. He helped us with implementing iTracker-face on the device. He was so competent that I did not think I had offered much help. He also was very kind. Thus, my first mentoring experience was super enjoyable. We also collaborated with Dr. James Kozloski from IBM shortly. James was also very considerate. While the work we collaborated on did not end up in this thesis, that work might one day become the key to linking this project directly with neurophysiology.

I cannot say how thankful I felt to all the volunteers. We did not pay any volunteers. Therefore, it took Gladynel and me so much time to find volunteers and we were thankful that so many people were willing to help us. It has helped us so much in learning how we may improve our app and algorithms. This experience made me gain deep respect to anyone working on human subjects research. I also want to thank James Bales from the Edgerton center to have lent us the research-grade cameras. Setting up the research-grade cameras made me feel like a "real" engineer. Thankfully we did not break the cameras, or else I may not have been writing this acknowledgment.

I also have a lot of other guidance throughout this Ph.D. journey. My master thesis advisor, Professor Alan Oppenheim, has always been there. He still invited me to his group meetings. He still guided me if I ever felt a little bit lost. He still encouraged me to keep improving myself and inspired me to live up to his vision of me. I also want to thank Professor Tamara Broderick. My longitudinal analysis came around thanks to her class and her literature suggestion. In addition, I want to

thank people from the Impact program. Dr. Burstein and Dr. Caffarel-Salvador, in particular, have helped put me into context with other neurophysiologists, including Dr. Fine-Edelstein and Dr. Galaburda. Talking with them has helped me define how my work may make an impact.

While all the previous acknowledgment was directly related to this project, I received tremendous indirect support. I enjoyed hanging out with my labmates (in Al's, Vivienne's, and Thomas' labs) — James Noraky, Tien-Ju Yang, Zhengdong Zhang, Yu-Hsin Chen, Amr Suleiman, Guolong Su, Pablo Martinez, Tarek Lahlou, Catherine Medlock, Nellie Wu, Peter Li, Soumya Sudhaker, Yi-Lun Liao, Diana Wofk, Jamie Koerner, Syed Muhammad Imaduddin, James Lynch, Frederick Vonberg. I would always remember my ICIP trip with James to Athens and the Christmas celebrations with Tien-Ju, Nellie, Yi-Lun, and Zhengdong.

Additionally, there are four groups of people I want to thank. First, I want to thank my friends from Sidney Pacific. As an international student, the community felt like my family. I want to thank my Wonderwomen roommates — Zelda Mariet, Geeticka Chauhan, and Tu-Lan Vu Han. Zelda taught me confidence. Geeticka taught me courage. And Tu-Lan taught me love. I would always cherish the time we have spent in those very special apartments. I want to thank people who have served in the SP government with me. We had so much fun interpreting SP constitution and interpreting life. I also want to shout out to Yen-Ling, Chia-Jung, and Aashka, who are my pandemic partners. We worked together, chatted together, struggled together. Eventually, we thrived together. The pandemic year has been stressful. With these amazing people, I had pushed everything through.

Secondly, I want to thank my first-year squad — Zied, Tadayuki, Eren, Igor, Nicha, and Govind. It was a blessing to have known all of these wonderful and supportive people. Thirdly, I want to thank my Taiwanese friends. In particular, I want to thank Chih-Yun for all her company and encouragement. We have known each other since high school and I want to cherish this long-lasting friendship.

Finally, I want to thank my family. I want to thank my brother for visiting me so many times in my Ph.D. I want to thank my dad for his support. I want to thank

my mom deeply. My studying abroad was never easy for her. Despite the loneliness, she has continued to support my dream, to remind me that I am loved, and to let me know that there is always a fallback plan. I sometimes missed home. However, I had many things around me that reminded me of home such as the bunny stuffed toy my mom managed to pack in my luggage, the beautiful cookie containers my mom sent me, and sometimes that weekly phone call from Mom. Therefore, I know I have a home. I have someone who loves me. And I love her back. And I know who I am, and can keep exploring who I will be.

To my mom

Contents

List of Figures

Chapter 1

Introduction

The ability to objectively, accurately, and frequently track neurocognitive states is important. For example, in transportation, drowsy driving contributes to 9.5% of all crashes [6]. Alcohol consumption can also temporarily lower cognitive ability, and an objective assessment of neurocognitive state might help reduce the rate of accidents in transportation or other settings.

Neurocognitive states also degrade over the progression of neurodegnerative diseases. The increase in life expectancy in the developed world as well as the failure in development of effective medications, particularly for Alzheimer's disease, makes the detection and tracking of changes in neurocognitive ability a pressing clinical need. Current assessments of neurodegenerative diseases are subjective and sparse, and standard neurocognitive and neuropsychological test batteries require a trained specialist to administer and score [7, 8]. Additionally, these tests demand significant patient time and cooperation, and can therefore be influenced by a patient's level of attention and comfort with the clinical setting [9]. The lack of objective and accurate assessment tools to quantify disease state hinders the development and validation of novel treatment strategies. Since the quest for disease-modifying therapies in neurodegenerative diseases is increasingly focusing on the early or even prodromal stages of the disease process, the need for accurate and frequent measures of disease progression and response to treatment has become urgent [10, 11]. Frequent assessments can also mitigate the effects of normal variations when determining neurodegenerative

disease progression.

Assessment of eye movement is a promising candidate for such a quantitative, objective, and frequent test. First, eye movements are readily observable. Second, their neural pathways involve several brain regions controlling cognitive functions, and they might hence be affected by degenerative processes affecting various brain centers [4]. Some diseases also directly affect oculormotor pathways, such as Huntington's disease and progressive supranuclear palsy. As a result, clinical eye-movement assessments are key to diagnosing and tracking these diseases. Among the clinical eye-movement assessments, pro- and anti-saccade visual reaction tasks are often used challenge tests [12, 3]. In the pro-/anti-saccade tests, a subject is asked to look towards/away from a visual stimulus. An anti-saccade task, in particular, requires a person to inhibit a natural reflexive eye movement towards the stimulus and initiate an eye movement in the opposite direction of the stimulus. Thus, it requires more cognitive processing than a pro-saccade task [13, 14]. Because these tasks demand cognitive abilities which can be affected by neurodegenerative diseases, two saccadic eye movement features were observed to be significantly different between healthy subjects and patients: saccade latency (visual reaction time) and directional error rate (the proportion of eye movements towards the wrong direction) [15, 16, 17, 18]. However, these features are commonly measured with dedicated infrared cameras and chinrests, which limits the measurements to the doctor's office or the neurophysiological laboratory. As a result, few longitudinal studies were conducted to analyze how saccade latency and error rate change over the disease progression [19, 20], and the measurements of these studies were usually too sparse (less than or equal to twice per year) to detect disease onset or efficiently evaluate treatment effects. An alternative to this approach could be afforded by performing eye movement tracking and analysis at the convenience of the patient on mobile devices such as cell phones and tablets with user-facing cameras. In fact, the use of such "digital biomarkers" has recently attracted significant attention in neurology [21, 22, 9] (where biomarkers refer to biological signs for a disease or a condition).

The goal of this thesis is to enable frequent and accurate tracking of saccade

latency and error rate using mobile devices. To achieve this goal, there are three building blocks. a) We aim to design an instructive and easy-to-use data-collection system that allows a subject to record themselves in their own homes and offices using a mobile device. With such a system, frequent recordings of eye movements become possible. b) We aim to design a robust and automated measurement pipeline that can measure saccade latency and error rate from mobile-device recordings. With such a measurement pipeline, frequent measurements of eye-movement features become available. c) We aim to analyze how these features change over time in healthy subjects. By characterizing these longitudinal eye-movement features from healthy subjects, we can put into context how neurocognitive impairment may affect these eye-movement features and evaluate the possibility to use these features to track neurocognitive states objectively, accurately, and frequently.

1.1 Related Work

Several directions of work are related to our research. First, some studies were conducted to understand how clinical biomarkers may be affected by disease progression. However, since these biomarkers rely on cognitive tests, neuroimaging techniques, and cerebrospinal fluid analysis, the assessments are not sufficiently quantitative, objective, and frequent to identify early or even prodromal stages. Digital biomarkers such as gaits, finger tapping, and saccadic movements are promising unobtrusive measurements to help detect disease onsets. While several mobile-device-based monitors have been proposed to measure gaits and finger tapping, most of these studies were not conducted longitudinally and mobile-device-based saccade measurements were still underdeveloped. Therefore, to the best of our knowledge, our work is the first to enable saccade measurements using mobile devices and is the first to have characterized how these measurements change over days in healthy subjects. Finally, since our ultimate goal is to potentially track the disease progression using eye-movement features, we gave a concise review on existing disease progression models (including models developed for other diseases).

1.1.1 Clinical Biomarkers of Neurodegenerative Diseases

Several studies and datasets have been proposed to understand how neurodegnerative diseases affect clinical measurements through the course of progression. One of the most studied open dataset for the clinical biomarkers of Alzheimer's disease (AD) is the Alzheimer's Disease Neuroimaging Initiative (ADNI) [23]. The study has started in 2004 and has collected a relatively comprehensive set of biomarkers. The Coalition Against Major Diseases (CAMD) provided another online dataset for AD [24]. Large-scale cohort studies are also being developed for other diseases including Parkinson's diseases (PD) [25] and Huntington's diseases (HD) [26, 27]. Most of these studies focused on measurements instrumented by physicians. As a result, the measurements were usually sparse in time and may not be used to track disease progression sufficiently frequently identify early stages. However, studying these datasets can help understand how disease progression is currently assessed.

1.1.2 Oculor Biomarkers of Neurodegenerative Diseases

Since eye movements are affected by disease progression [4, 28], it has been suggested that eye movement patterns are useful and informative adjuncts to the standard neurocognitive assessment tools in routine clinical care and clinical trials [3, 29, 30, 31]. Eye movements are also accessible. Therefore, they may be used as objective and frequent assessments of disease progression. In this work, we focus on two of the most studied saccade features – saccade latency (visual reaction time) and error rate (the proportion of eye movements towards the wrong direction) – which can be measured from pro/anti-saccade tasks (tasks to look toward/away from a stimulus). Such visual reaction tasks require cognitive attention as well as appropriate execution of oculomotor responses once a stimulus is registered. This stimulus-response paradigm therefore probes a subject's cognitive and oculomotor function, either or both of which can be impaired in neurocognitive diseases [3, 15, 16, 17, 18]. Due to the restricted recording setup in the literature, there are few studies that track the longitudinal changes in saccade latency among patients [19, 20] and the measurements are sparse (fewer than

two recording sessions a year). Therefore, it is unknown whether saccade latency and error rates can be frequent, objective, and accurate measurements for monitoring disease progression. One of our major objectives in this work is to understand how these features change over time in healthy subjects to put into context how these features may be affected by disease progression.

1.1.3 Digital Biomarkers of Neurodegenerative Diseases

To enable frequent and quantitative measurements of neurodegenerative diseases, several mobile-device monitors have been proposed to meausre digital biomarkers. Since most of these studies are in early stages, the main focus is to detect a disease rather than to track the progression of a disease. To get an overview of what measurement platforms have been developed, we mention several work including two review papers for mobile-device monitors for AD [32, 33] and a review paper for gait-based monitors for PD[34]. Recently, multiple methods have been proposed to estimate eye gaze using data collected from mobile devices [35]. Among them, convolutional-neural-network-based algorithms [2, 36] have become the state of the art. While these algorithms are tuned to optimize gaze estimation accuracy, this metric does not translate into accuracy of saccade onset detection. To the best of our knowledge, our work in [37] is the first to measure saccade latency and error rate using mobile devices, which enables the potential to consider saccade latency and error rate as digital biomarkers for neurodegenerative diseases.

1.1.4 Disease Progression Modeling

After we enabled frequent measurements of saccade latency and error rate using mobile devices, we developed longitudinal models for healthy subjects in Chapter 4. The models were developed based on literature in disease progression modeling, since we envisioned extending our model for such a purpose. There are several approaches to disease progression modeling – a graphical model [38], a Gaussian process model [39, 40, 41, 42], and an recurrent neural network [43]. A graphical model is

helpful in learning the relationship between biomarkers and discrete disease states. A Gaussian process (GP) model is useful in modeling the disease progression as a continuous process. A recurrent neural network model tends to rely on a large amount of training data and a large number of features. Due to limited longitudinal studies on non-intrusive measurements, most disease progression models developed for neurodegenerative diseases focus on clinical measurements [44, 45, 46, 43, 42]. Our goal is to develop individualized longitudinal models for the eye-movement features we collected from healthy subjects. We used a GP model for three reasons. First, one can capture how the correlation over time and the correlation across the features are characterized. Second, GP is a nonparametric model and its complexity can be adapted to the complexity of the training data. Compared to a linear model which can only characterize a linear function, GP can characterize an infinite dimensional function. Thus, it is more flexible than any model consisting of a finite number of basis functions. Third, because any finite samples from GP form a Gaussian distribution, the computation for learning and inference is relatively simple. Therefore, a GP model provides interpretability, flexibility, and computability. An in-depth overview of GP models can be found in [47]. Our models are special cases of a multi-task GP model[48], which is known as linear models of coregionalization (LMC) in the geostatistics literature [49]. With the amount of data we have collected, we developed a model similar to that in [50]. However, we carefully designed the hyperparameters and whether they should be individualized or shared across the subjects based on the characteristics of the eye-movement features. This design allowed us to enable individualized tracking of saccade latency and error rates from healthy subjects.

1.2 Summary of Contributions

We map the contributions of this work to the three building blocks – a self-guided mobile-device-based data-collection system, a measurement pipeline for measuring saccade latency and error rate from large-scale mobile-device recordings, and characterization of eye-movement features from healthy subjects.

A self-guided mobile-device-based data-collection system (Chapter 2)

In this thesis, we developed an **instructive** and **easy-to-use** app to display the pro-/anti-saccade tasks on an iPad while recording a subject's eye movements with the built-in camera. The app guides a subject to adjust their recording setup to record themselves properly in their own homes and offices without the assistance of an expert. In addition, to measure saccade latency on a tablet, we ensured that the absolute timing error between on-screen task presentation and the camera recording is within 5 ms, which is well within the standard deviation of a subject's saccade latency distribution. With this user-friendly recording platform, we collected over 6,800 videos and over 235,000 individual eye movements from 80 subjects across the adult age spectrum, which is around 100 times more eye movements than in previous literature.

A measurement pipeline for large-scale mobile-device recordings (Chapter 3)

With large-scale mobile-device recordings enabled by the app, we presented a **robust** and **automated** measurement pipeline to track eye movements and determine pro-/anti-saccade latency and error rate. Without infrared illumination, the lighting conditions are less controlled. We used a deep convolutional neural network for gaze estimation and showed that it is robust to different lighting conditions. Without a chinrest, head movements may affect saccade latency measurement. We used a model-based approach for saccade latency measurement that allows for automated flagging and rejection of eye-movement traces that might be of questionable quality. In addition, we developed an automated algorithm for measuring error rate that takes into account the possibility that it might not always be possible to determine the direction of an eye movement from app-based recordings due to insufficient lighting or eyelid drooping. Finally, we showed that simultaneous recordings with a smartphone and a high-speed camera resulted in negligible differences in saccade latency distributions. Additionally, our error detection algorithm achieved a sensitivity of 0.97 and a specificity of 0.97. We therefore can conclude that we have enabled saccade latency and error rate measurements from mobile-device recordings outside of clinical

environment.

Characterization of eye-movement features (Chapter 4)

Understanding the characteristics of eye-movement features from healthy subjects is key to understand how disease progression affects eye-movement features. With the data we obtained through the app and the measurement pipeline, we analyzed the individual distributions, the day-to-day variations, the correlations across the features, and the relationship between the features and age. In particular, we observed that the various strategies subjects used to perform the tasks could introduce significant intra- and inter-subject variability in the day-to-day variations. This observation highlights the importance of individualized tracking of eye-movement features.

We then built a Gaussian-process-based individualized longitudinal model based on the intuitions we obtained from analyzing these eye-movement features. We showed that if there are more than 25 days of recordings, our model can characterize the data better than assuming that eye-movement features do not change over time in healthy subjects. In addition, our model can learn the correlation across the eye-movement features and thus learn the subjects' task-performing strategies. With an **individualized** and **interpretable** model, we hope it can provide a foundation to put into context the effects of different neurocognitive states on eye-movement features.

Conclusion and significance (Chapter 5)

Our system and algorithms allow ubiquitous tracking of saccade latency and directional error rate, which opens up the possibility of quantifying neurocognitive states on a finer timescale in a broader population than previously possible.

1.3 Publications based on work done in this thesis

Chapter 2 and Chapter 3 are modified based on the following publications. My contributions are mostly on the design of the app and the development of the measurement pipeline.

- H.-Y. Lai, G. Saavedra-Peña, C.G. Sodini, T. Heldt, and V. Sze, "Enabling

saccadelatency measurements with consumer-grade cameras," *in Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 3169–3173.

- G. Saavedra-Peña, H.-Y. Lai, V. Sze, and T. Heldt, "Determination of Saccade Latency Distributions using Video Recordings from Consumer-Grade Devices," *in Proceedings of the IEEE International Engineering in Medicine and Biology Conference (EMBC)*, 2018.

- H.-Y. Lai, G. Saavedra-Peña, C.G. Sodini, V. Sze, and T. Heldt, "Measuring Saccade Latency using Smartphone Cameras," *IEEE Journal of Biomedical and Health Informatics (JBHI)*, vol. 24, no. 3, pp. 885-897, 2020.

- H.-Y. Lai, G. Saavedra-Peña, C.G. Sodini, T. Heldt, and V. Sze, "App-Based Saccade Latency and Error Determination across the Adult Age Spectrum," in Review for a journal, `arXiv:2012.09723 [q-bio.NC]`.

Most of Chapter 4 is in preparation for a journal publication.

Chapter 2

Data-collection System

In this chapter, we discuss the design of our measurement system, including the task design, the recording setup, and the recruitment efforts. There were two stages of this design. The goal of the first stage was to enable measurements of eye-movement features using mobile-device cameras. Thus, we controlled the recording setup carefully to evaluate the performance of our algorithms in different lighting conditions. Once we can validate that it is possible to measure saccade latency using mobile-device cameras, in the second stage, we aimed to enable large-scale eye-movement measurements using mobile devices. Therefore, we developed an app to guide a subject to record themselves in the comfort of their homes and offices. To better present our results taken in these two different stages, throughout this thesis, we would refer to the first stage as "Validation Stage" and the second stage as "Deployment Stage".

2.1 Task Design

2.1.1 Validation Stage

In the Validation Stage, we used the Psychophysics Toolbox 3 for Matlab [51] to implement the visual fixation/stimulus task presented to participating subjects on the laptop screen. A single saccade task started with a fixation period in which three squares were presented on the screen, arranged horizontally, against a black

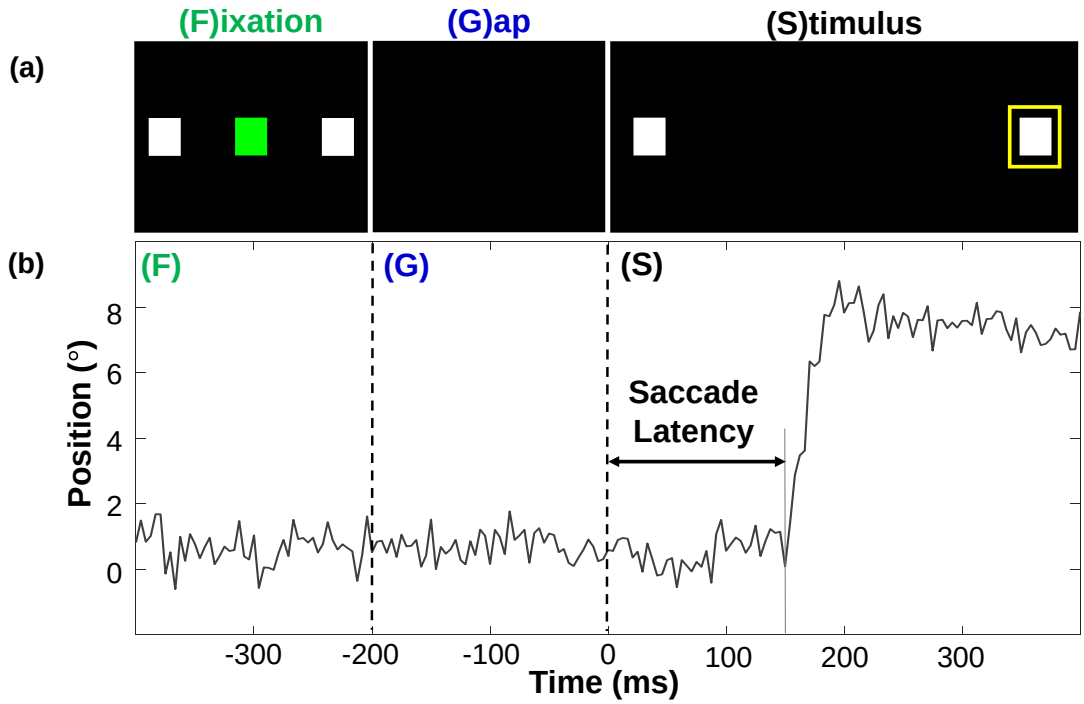

Figure 2-1: Saccade task in the Validation Stage: (a) Example of the visual tracking task during a saccade-latency measurement. The tasks consisting of a fixation period (F), a gap (G), and the appearance of the stimulus (S). Only the final 200 ms of the fixation period are shown. (b) The corresponding horizontal eye movement trace.

background, a green square at the center of the laptop screen and two white squares arranged at a horizontal distance on either side (Figure 2-1a). Subjects were asked to fix their gaze on the green square. After 1000 ms of fixation, all three squares disappeared. Following a 200 ms gap, the two lateral squares reappeared in their original position, with one of them bounded by a yellow square (the stimulus). Subjects were tasked with moving their eyes to – and subsequently keeping their gaze fixed on – the stimulus (Figure 2-1b). After the stimulus disappeared, subjects returned their gaze back to the centrally located green square. This task was repeated 40 times per trial, with a total of 20 stimuli appearing on the right and 20 on the left in randomized order. Each recording session consisted of three such trials conducted in close succession, resulting in 120 saccade tasks per session and taking about ten minutes to complete (including breaks between trials).

2.1.2 Deployment Stage

To better compare our results with the previous work, in the Deployment Stage, we implemented two commonly studied tasks in the literature, namely a gap-pro-saccade and a gap-anti-saccade task [52, 18, 16]. Both tasks start with a fixation period. During the fixation period (1 s), a fixation point (green square) is shown at the top center of the screen (as shown in Figure 2-2). Subjects were instructed to look at the fixation point during this period. The fixation is followed by a 200-ms gap period, where the fixation point disappears and the screen stays black. After the gap period, a stimulus (white square) is presented on either left or right side of the screen. If a subject is performing a pro-/anti-saccade task, the subject is instructed to move their eyes towards/away from the stimulus as quickly and accurately as possible. This stimulus period will last for 1.2 s and be followed by another 200-ms gap period. This sequence of "fixation-gap-stimulus-gap" will repeat for 20 or 40 times, with half of the stimuli presented to the right of the fixation point and half to the left in randomized order. Here, how many stimuli appear in a saccade task depends on whether a subject participates in one or multiple recording sessions. Subjects who chose to participate in a single session recorded three pro-saccade tasks and three anti-saccade tasks, where each task consists of a set of 40 stimuli. Subjects who chose to participate in multiple recording sessions were asked to take three pro-saccade and three anti-saccade tasks every day for at least two weeks and were given the choice of 20 or 40 stimuli per task.

It is beneficial to change the task design from the design in the Validation Stage to the design in the Deployment Stage. The reason is as follows. It requires more cognitive awareness to perform an anti-saccade task than a pro-saccade task, since a subject needs to inhibit an eye movement towards the stimulus and initiate an eye movement away from the stimulus. However, interestingly, we can see similar effects introduced by the task design in the Validation Stage. We notice that in the Validation Stage, during the stimulus presentation, both the white squares were presented. As a result, although we instructed a subject to look towards the white square surrounded

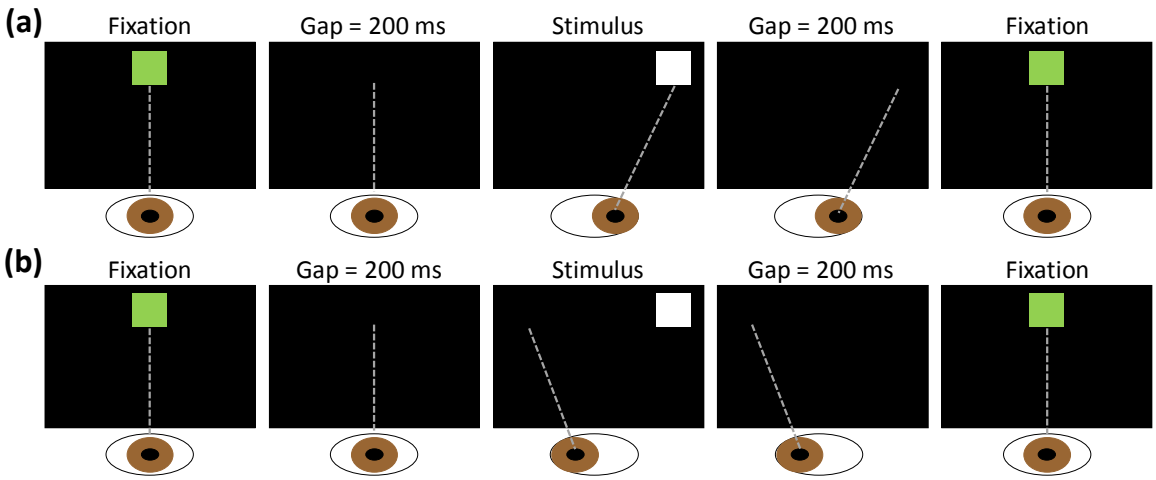

Figure 2-2: Saccade tasks in the Deployment Stage: (a) Pro-saccade task: Look toward the stimulus. (b) Anti-saccade task: Look away from the stimulus.

by a yellow square, the other white square may still hinder the subject from moving their eyes towards the stimulus. By presenting only one white square during the stimulus period in the design in the Deployment Stage, we can ensure that there is no such inhibition during a pro-saccade task. Therefore, the design in the Deployment Stage not only was similar to the tasks in the literature, it also allowed us to study the effect of inhibition on eye movement features by comparing the features from an anti-saccade task with those from a pro-saccade task. Since the inhibition of an eye movement towards the stimulus was studied in [13] to be related with the subcortical areas, such study can help us understand how disease progression affects those brain regions.

2.2 Recording Setup

2.2.1 Validation Stage

The video recording of volunteers in both stages was approved by MIT's Committee on the Use of Humans as Experimental Subjects, and informed consent was obtained from each participant prior to recording. In the Validation Stage, subjects were seated centrally in front of a laptop at a distance of about 1 m, with their chin

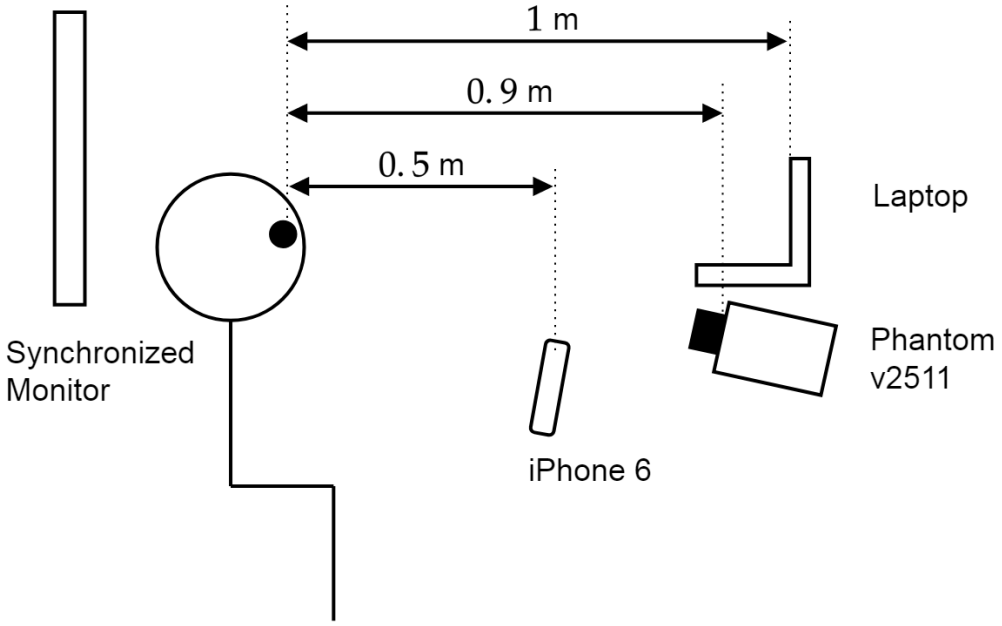

Figure 2-3: Diagram of the video recording set-up in the Validation Stage. A subject is seated facing an iPhone, (in some experiments) a high-speed camera, and a laptop displaying the visual stimulus task. A synchronized monitor behind the subject also displays the visual stimulus task so the cameras capture the eye movements and the visual task simultaneously.

Table 2.1: Validation Stage Camera & Recording Specifications

	Frame Rate	Resolution	ISO	Pixel Size	Shutter Type	Cost
iPhone 6	240 fps	1,280×720	32-160	1.5 μm	Rolling	$200-400
Phantom v2511	500 fps	1,280×720	6,400-32,000	28 μm	Global	∼$150,000

placed comfortably on a soft chinrest to minimize head movements (Figure 2-3). The sequence of visual stimuli were presented on the laptop screen. A second monitor was placed behind the subject's head, facing and mirroring the laptop screen. An iPhone 6 was placed centrally between the subject and the laptop screen at a distance of about 0.5 m from the subject and with the rear-facing (non-selfie) camera facing the subject. The laptop position was chosen to generate eye movements of 10° amplitude, and the camera position was chosen to capture the subject's face and the mirrored screen during the task, thus capturing the eye movement and the visual stimulus sequence in the same recording. Video recordings were made in slow-motion mode, resulting in recordings at 240 frames per second (fps) and a resolution of 1280×720 pixels. In a subset of recordings, we additionally and simultaneously collected reference videos with a high-speed camera (Phantom v2511) at 500 fps and a resolution of 1280×720 pixels (see Table 2.1). The distance from the high-speed camera to the subject was about 0.9 m; the camera lenses focused on the subject's eyes. Most recordings were acquired under fluorescent lighting. To understand the robustness of the recordings to realistic variations in ambient conditions, we collected a separate set of recordings while varying the lighting conditions with the help of LED panels, and subjects were recorded with and without glasses.

2.2.2 Deployment Stage

Recall that in the Validation Stage, we displayed the visual reaction task on a laptop and recorded the subjects with an iPhone. Synchronization of the recording and task display was achieved through a second screen that mirrored the laptop screen and was recorded alongside the subject's response. Given the elaborate set-up, the

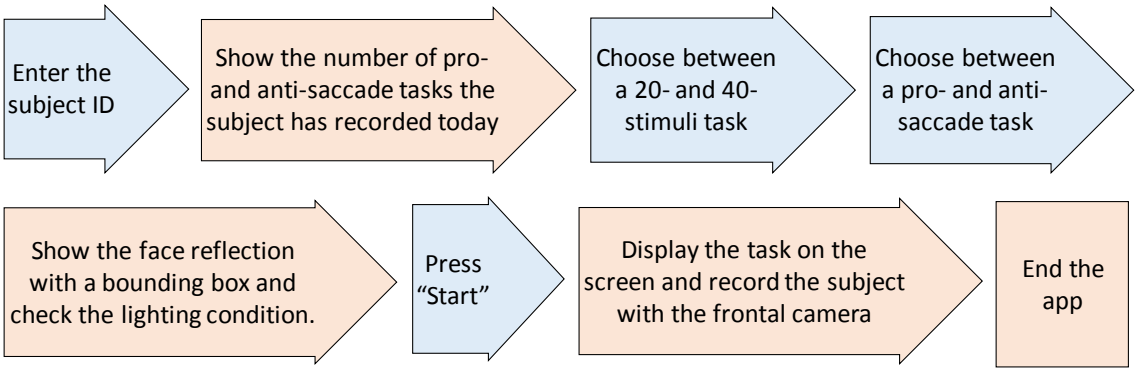

Figure 2-4: The flow of the app. Blue arrows request the input from the subject. Orange arrows denotes the response of the app.

recording was limited to our laboratory setting. In the Deployment Stage, our goal was to allow for ubiquitous recording and hence for subjects to record themselves in the comfort of their homes or offices. We therefore developed an iOS app so subjects could record themselves with the frontal (i.e., selfie) camera as the tasks were displayed on the screen. While the app can run on iPhones, our platform of choice was the iPad (Generation 2 and 3) for their larger dimensions and hence larger angular gaze amplitudes ($\sim$12.7 degrees at a distance of 40 cm to the camera).

The flow of the app is shown in Figure 2-4. The app first obtains the subject's ID and then reminds the subject of the number of pro- and anti-saccade tasks they have performed the same day. Subsequently, subjects are prompted to select the number of stimuli they wish to perform (20 or 40, depending on how they were instructed as discussed in Section 2.3) and whether they would like to perform a pro- or anti-saccade task.

To minimize the influence of environmental conditions on the quality of the recordings, we initially asked subjects to position the iPad at a distance of 30 to 50 cm. In subsequent releases for iPads with depth-sensing capability, the app senses the distance from the subject and provides visual feedback so the subject can position the iPad within the desired target distance (Figure 2-5). Besides distance, the app also guides the subject to position themselves in proper lighting conditions, and the subject will be asked to move to a brighter location if the automatically detected ISO

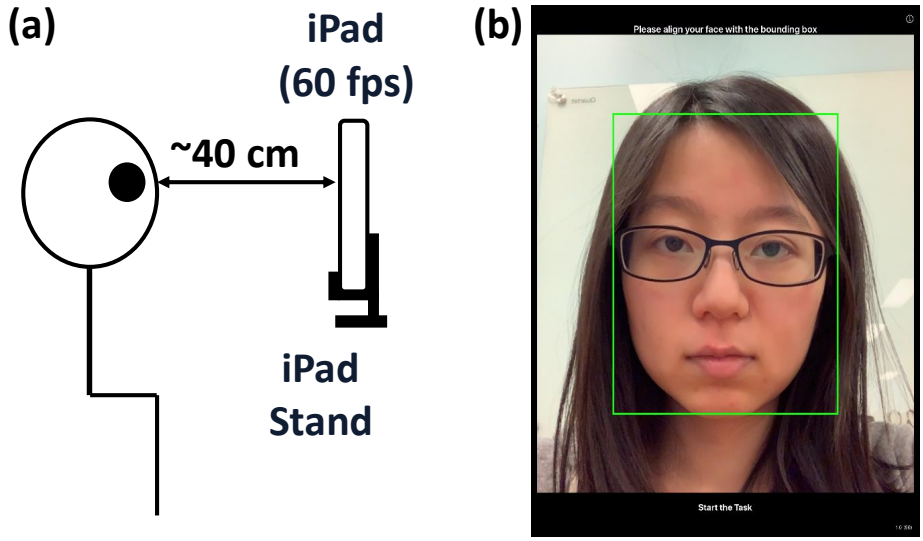

Figure 2-5: (a) Recording setup in the Deployment Stage; (b) before showing the task on the screen, the app displays the face of the subject with a bounding box. If the distance measurement from the camera to the subject's face is accessible (i.e., between 30 and 50 cm), the box will turn green. If the automatically detected ISO is greater than 1000, a warning will be shown to guide the subject to move to a better-illuminated place.

is greater than 1000.

When the subject is ready to perform the task, they start the recording, and a count-down will be displayed so the subject can begin to focus their attention on the pro-/anti-saccade task. The task will then be displayed while the frontal camera simultaneously records the subject's face. Once the subject completes the tasks, the app will ask for the subject's tiredness information. Then the task will ask if the subject wants to process the first 10 saccades to get immediate feedback about their saccade latency and error rate. After the task is completed, a detailed set of data files is saved for each recording, including: (a) the actual video recording, (b) the timestamps of each recorded frame, (c) the timestamps of each frame displayed on the screen, and (d) a text file containing information about the recording system (iOS version, iPad generation), the distance of the iPad to the subject (when available), and the recorded ISO value at the beginning of the recording.

We can make two remarks about this design. First, to acquire accurate saccade latency measurements, it is crucial to synchronize the task display on the iPad screen and the recording from the iPad camera. We detailed and evaluated the synchroniza-

tion in Appendix A. By requiring the ISO to be less than 1000, we showed that we can bound the absolute synchronization error to be within 5 ms, which is well within the standard deviation of a subject's saccade latency distribution.

The second remark is about the feedback we provided by processing the first ten saccades. While the results presented in Chapter 3 and Chapter 4 were based on data processed on server where we processed all the saccades instead of just the first ten saccades, this immediate feedback demonstrated the possibility to process the recordings on device. Such design is helpful for two reasons. First, with the feedback, a subject can evaluate their recording setup and adjust if necessary. If few saccadic features can be measured from the first ten eye movements, it is likely that the recording setup can be improved. Second, if data are processed on device, we may only need to upload the processed eye-movement features to our storage. In this case, not only can the required memory be reduced, without the subjects' original recordings, the privacy of these subjects can be more protected.

2.3 Recruitment Effort

In the Validation Stage, we recorded 19,200 saccadic eye movements from 29 healthy subjects where most of them are young adults (less than 30 years old). Among these subjects, eleven subjects have recorded five or more recording sessions. In the Deployment Stage, our recording setup was much more flexible, allowing us to record more subjects and more repeated sessions. Therefore, in the Deployment Stage, we recorded over 235,000 saccadic eye movements from 80 self-reported healthy adult subjects, ranging in age from 20 to 92 years. The number of single and multiple recording sessions, grouped by decades of age is shown in Figure 2-6.

2.4 Discussion and Summary

In this chapter, we presented the design of our measurement system to acquire eye movements without the need for specialized equipment (such as infrared illumination,

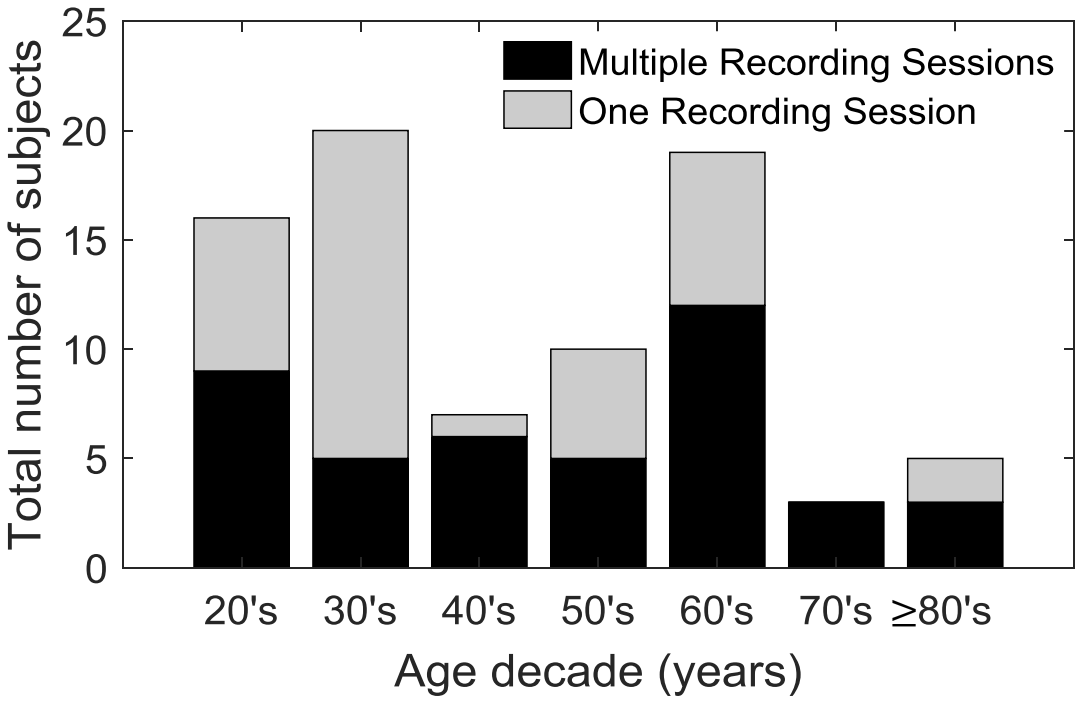

Figure 2-6: Age distribution of subjects with single or multiple recording sessions in the Deployment Stage.

chinrest and research-grade cameras). In the Validation Stage, we showed that instead of a special-purpose camera, we can measure saccade latency using a smartphone camera. The recording setup, nevertheless, required a laptop to display the task, a screen synchronized with the laptop to be placed behind the subject, and a researcher to record both the subject's eye movement and the synchronized screen using the back camera of an iPhone. Due to these requirements, the recording setup was not sufficiently portable for a subject to take recordings on their own in their homes or offices, which limits the possibility of using such a system to flexibly and ubiquitously monitor neurocognitive decline or disease progression. In the Deployment Stage, we designed an iOS app to record a subject with the frontal camera of an iPad while the subject is following a task shown on the screen. There are two challenges to achieve this goal.

First, unlike in the clinical setup and in our previous work where an expert researcher takes recordings of a subject, our app needs to guide the subject to record themselves at a proper distance to the camera and in a well-lit environment. To resolve this first challenge, before recording a subject, the app displays the subject on the screen and guides the subject to align their face with a bounding box shown on the screen. With such guidance, most subjects were recorded at an appropriate distance. To ensure the environment is well-lit, the app also asks the subject to move to a better-illuminated environment if the measured ISO is greater than 1000.

Second, the camera recording and the task displayed on the screen need to be well-synchronized to obtain accurate saccade latency. This can be challenging as most applications (e.g., video chatting) only require the synchronization error to be unnoticeable by a human (i.e., less than 80 ms). With careful app design and evaluation of the synchronization error, we show that we can restrict the absolute timing error to be within 5 ms, which is well within the standard deviation of a subject's saccade latency distribution.

Besides the recording setup, we also significantly increased the number of recordings we obtained. With the improvement in our measurement system, in the Deployment Stage, we took 6,823 recordings from 80 subjects ranging in age from 20

years to 92 years, a significantly larger number compared to the number obtained in the Validation Stage – around 500 recordings from 29 subjects mostly in their 20's and 30's, and most other work collected just one or two recordings from each subject [17, 53, 52]. Moreover, there were 43 subjects with multiple recording sessions in the Deployment Stage compared to 11 subjects in the Validation Stage.

In summary, we designed a novel app to guide a subject to record themselves in a well-lit environment at a proper distance, implemented both pro-saccade and anti-saccade tasks, and recruited a cohort of subjects spanning the adult age spectrum. With our portable measurement system, we collected over 235,000 eye movements in 80 subjects ranging in age from 20 to 92 years, around two orders of magnitude more than reported in most of the literature. With the data, we can analyze the relationship between eye-movement features and age and how they change over time in Chapter 4.

Chapter 3

Measurement Pipeline

In this chapter, we discussed the algorithms we designed to measure saccade latency and error rates from the mobile recordings. Our measurement pipeline is shown in Figure 3-1. The two principal steps are (1) *eye-tracking* to extract the eye position from each frame in a video sequence, and (2) *feature extraction* to measure saccade latency and error rate.

3.1 Eye Tracking Algorithm

Here, our goal is to estimate where a subject is looking at on the screen over time. However, instead of aiming for a higher accuracy in gaze position estimation, to acquire accurate saccade latency and error rate, we aim for a higher accuracy in saccade onset detection. Thus, three eye-tracking algorithms – Starburst [1], iTracker [2], and phase-based motion magnification [54], were modified to achieve this goal. These algorithms were chosen because they were developed based on three different concepts – feature/model-based, appearance-based, and phase-based algorithms and have achieved promising gaze tracking results.

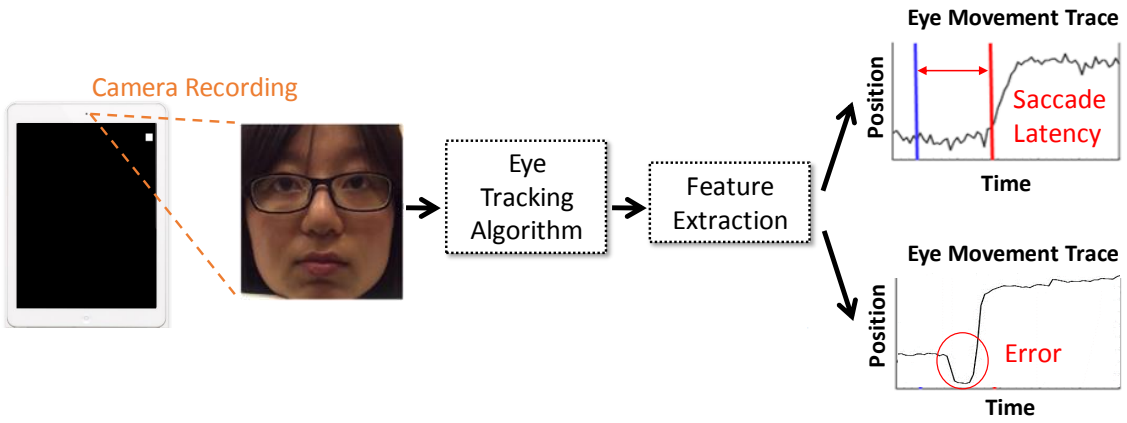

Figure 3-1: The measurement pipeline includes the mobile-based video recording, an eye tracking algorithm, a saccade-latency measurement algorithm, and an error detection algorithm.

3.1.1 Feature- and Model-based Algorithm

Starburst is a feature- and model-based algorithm developed for a head-mounted eye-tracking system [1]. It relies on infrared (IR) illumination to provide a sharp boundary between the pupil and iris (Figure 3-2). An initial estimate of the pupil center is used as a seed, and the pupil-iris boundary is detected using gradient-based features along rays that extend radially outward from the seed. RANSAC is used to iteratively fit an ellipse to the detected boundary and arrive at a final estimate of pupil center for each frame [1]. The fixed camera pose relative to the eyes ensures that the eye is always in the same region relative to the camera, which makes algorithm initialization easy across trials. However, the benefits of IR illumination and head-mounting no longer hold when the eye movement is captured with an iPhone camera with a varying pose under natural light.

To address these limitations, we develop **Starburst-phone**. First, we estimate the iris center instead of the pupil center, considering that in visible-spectrum imaging the boundary between the iris and the sclera is often more distinct than the pupil-iris boundary [1] (Figure 3-2). With an iPhone, the camera pose can vary, and thus the eye-crop position must be manually determined; by assuming minimal head movement during each test, which lasts under two minutes, the same eye-crop position can be

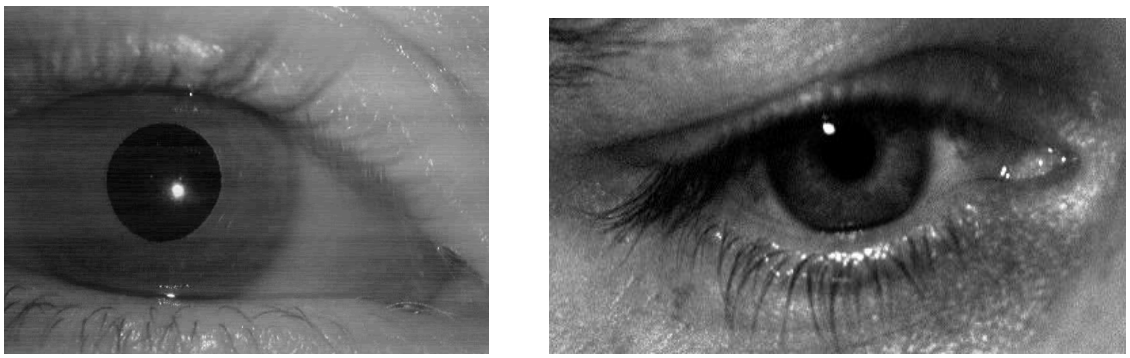

Figure 3-2: Eye images with *(left)* infrared (Figure from [1]) versus *(right)* natural light.

used for all frames. Similarly, the pupil center is also manually initialized in the first frame of each test; however, subsequent frames initialize the pupil center based on the previous frame, which allows for some minor head movement.

Figure 3-3 shows how the rays are generated from this initialization point and the gradient along each ray is calculated. We detect the iris contour by choosing the point with the maximum gradient along each ray rather than choosing the point that first exceeds a fixed gradient threshold. Since we are now measuring the boundary between the iris and sclera, the upper eyelid can cause occlusion and the directions of the rays are restricted accordingly. Due to the reduction in the number of rays, we fit a circle model to the iris contour rather than an ellipse. A circle has fewer parameters compared to an ellipse, giving a more stable estimate with fewer feature points. Finally, to adapt to the various lighting conditions, histogram equalization must be selectively applied. Figure 3-6 shows an example eye trace using the Starburst-phone algorithm.

3.1.2 Appearance-based Algorithm

iTracker uses a convolutional neural network (CNN) that is trained to determine where a user is looking on a screen (i.e., gaze estimation) based on images taken from a frontal camera of an iPhone or iPad [2]. These images were collected through an iOS application named GazeCapture, which includes built-in iOS face and eye detectors. The inputs of the iTracker include a cropped left eye, a cropped right eye, a cropped

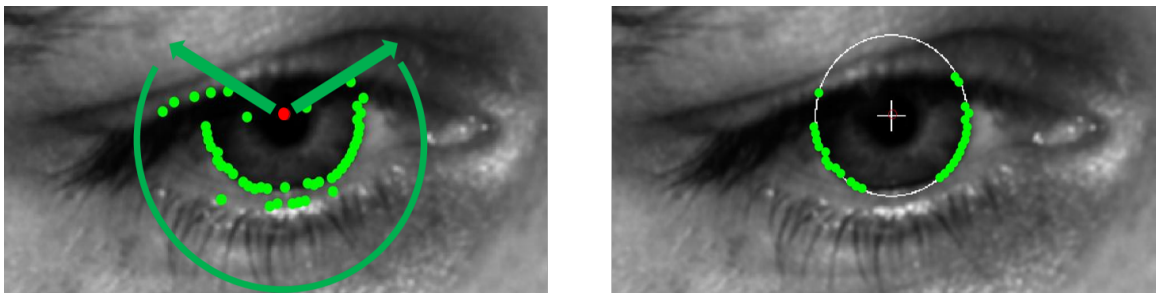

Figure 3-3: The Starburst-phone algorithm operating under natural light; (*left*) iris contour detection that avoids the upper eyelid; (*right*) iris model fitting.

face, and a face grid indicating the location of the face within the frame. All the input images have a resolution of 224×224 pixels. The architecture of the CNN is shown in Figure 3-4.

Before we evaluated the automation of eye crops and face crops as discussed in Appendix B, we manually annotated six anatomical landmarks on the first frame of each video clip: the two corners of each eye and the two corners of the mouth. To crop each eye region, in accordance with [36], we determined the midpoints of the inner and outer corners of each eye and surrounded these midpoints with squares of width 1.5 times the distance between the corners (Figure 3-5). We also computed the centroid of the six annotated landmarks and determined the face-crop region likewise as the square of width 1.5 times the largest distance of any two of the six landmarks, centered at the centroid location. Since all images are fed into iTracker at a resolution of 224×224 pixels, they undergo resizing from the original resolution. The eye crops are upsampled, while the face crop is downsampled with an anti-alias filter, using the `imresize` function in Matlab. We then apply iTracker to each frame in the video sequence, and the x-coordinate of the estimated gaze location over time is taken as the horizontal eye-movement trace. Nevertheless, we discovered that in some challenging scenarios (e.g., the illumination was low or the subject was wearing glasses), the variations in the output of iTracker can be so large that the saccade onset becomes ambiguous.

To further understand the source of the variations, we tested the output of iTracker

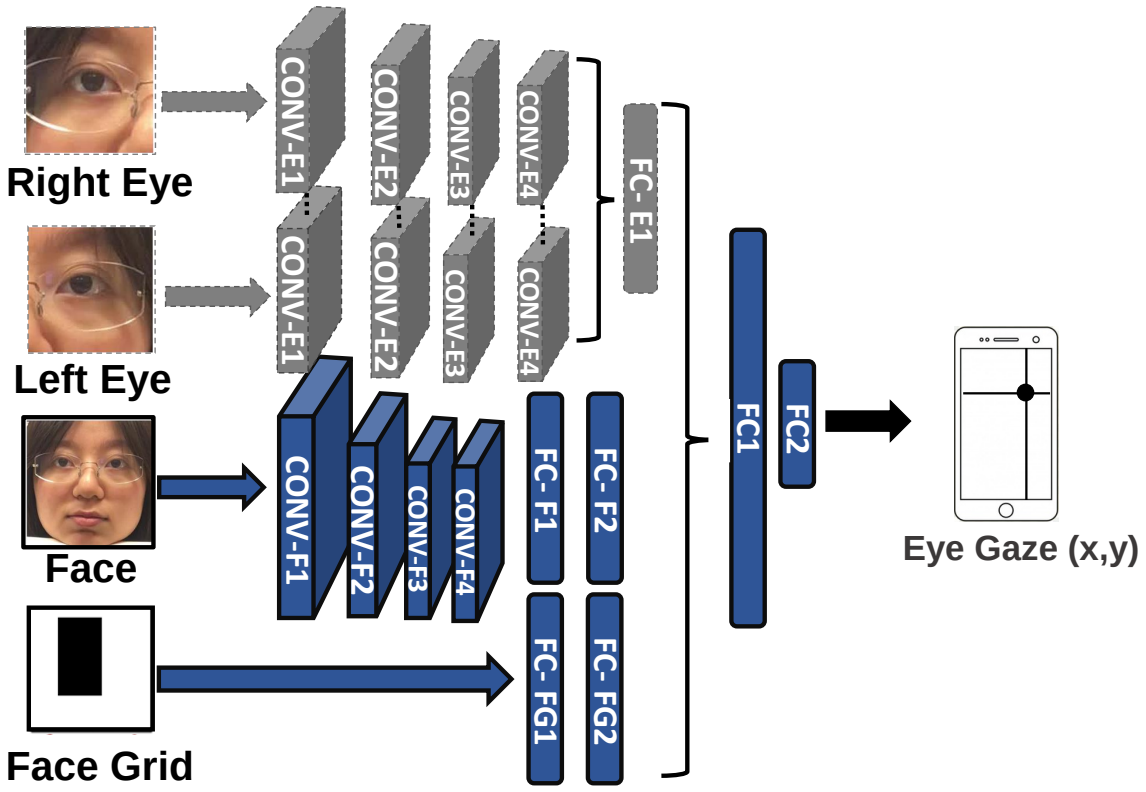

Figure 3-4: Convolutional neural network architecture used by iTracker and iTracker-face [2]. iTracker processes the face grid and the eye and face layers *(gray and blue)*, while iTracker-face only processes the face layers *(blue)*. See Krafka et al. [2] for details.

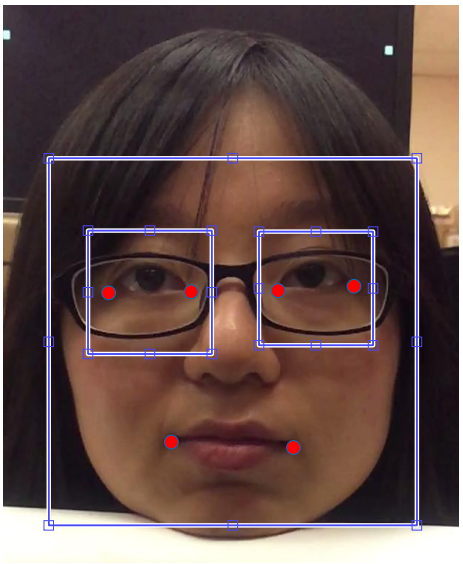

Figure 3-5: Manual eye crops and face crops for input to iTracker. The corners of the eyes and the mouth are manually determined on the first frame. The bounding boxes show the regions of eye and face crops derived from these fiducial markers.

when fixing the face grid input and two of the other three inputs (left eye crop, right eye crop, and face crop) to be the first frame of the video. We discovered that the variations in the output will be the smallest when we only changed the input to the face layers. Since the receptive field in the cropped eye only contains parts of the eye, one potential explanation for the observation could be that the eye layers may be trained to learn detailed features in the eyes to fine-tune the gaze estimation. On the contrary, the receptive field in the cropped face may contain a full eye. That is, the face layers may be trained to learn more global features in the eyes. When the image becomes blurrier, the detailed eye features will be replaced by noise, which causes the eye layers more sensitive to noise than the face layers.

Thus, we propose the **iTracker-face** algorithm, for which we only use the face-related convolutional layers of iTracker (Figure 3-4 blue layers). Although this choice does degrade the accuracy of the gaze estimation as discussed in [2], our objective is to determine the saccade onset (the time when the gaze changes). Figure 3-6 shows a sample eye-position trace using the iTracker and iTracker-face algorithms. In our application, iTracker-face generally has higher signal-to-noise ratio than iTracker.

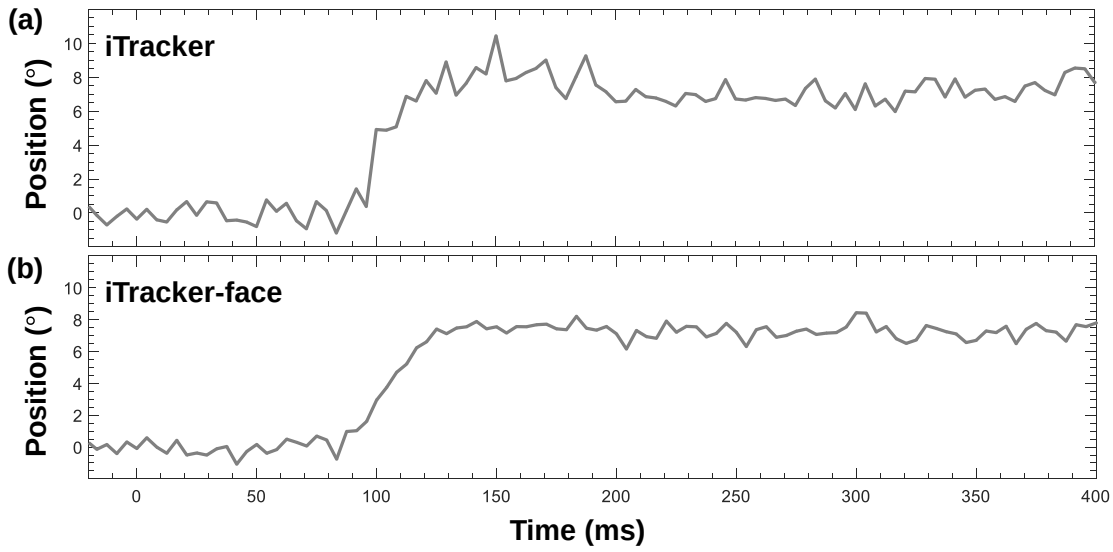

Figure 3-6: The same sample eye-movement trace from (a) iTracker and (b) iTracker-face.

3.1.3 Phase-based Algorithm

Motions in an image correspond to phase shifts in the frequency domain. To measure motions in a video, Davis et al. used a complex steerable pyramid wavelet to compute the phase changes across the frames [55, 54]. The complex steerable pyramid wavelet decomposes an image into complex-valued sub-bands in different scales and orientations. The phase change in each sub-band across the frames correspond to some local motions. The scale and the orientation of each sub-band determine the size and the direction of the motion respectively. If a video only contains one motion, one can estimate this motion by computing a weighted average of the local motions.

Most of our videos contained only one motion – the horizontal saccadic movement. Since we know the size and the direction of this eye movement, instead of averaging through filters of different scales and different orientations, we empirically determined the best scale and orientation. To begin with, we noticed that the eyes moved approximately by five pixels in our recordings. We therefore chose the fourth-order octave band filter to provide the best spatial support. Moreover, since we knew the movement was horizontal, we used a 2-orientation steerable pyramid filter and only measured phase shifts in the filter corresponding to the horizontal orientation.

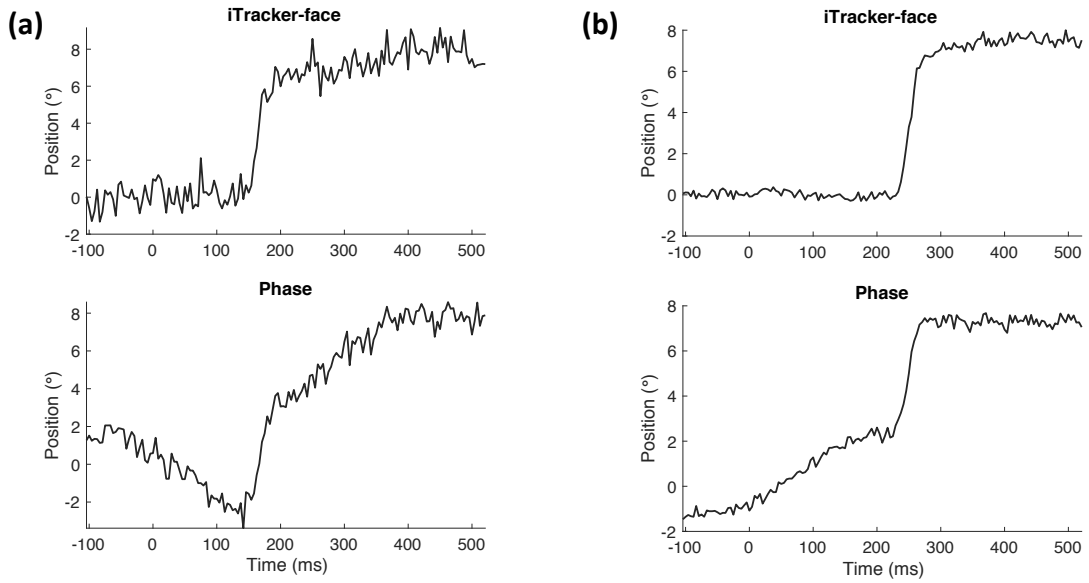

Figure 3-7: Two example eye-movement traces estimated from iTracker-face and the phase-based method.

We then unwrap the phase under the assumption that the phase change across two frames should be smaller than π. However, this assumption may fail when there is a blink.

3.1.4 Robustness of Eye-Tracking Algorithms

After experimenting these algorithms on our data, we noticed that the phase-based algorithm was very sensitive to movements. As shown in Figure 3-7, while the saccadic movements in all these traces were noticeable, the phase-based algorithm tended to pick up other movements in the video (e.g., a change in the reflection in the eyes) and introduced a positive or negative trend during the fixation period. It is hard to tell whether such a trend was caused by an eye movement or other movements. As a result, we chose to remove the phase-based algorithm from our consideration.

On the other hand, Starburst-phone is sensitive to the appearance of the eyes. In [37], we compared the performance of Starburst-phone, iTracker, and iTracker-face. We noticed that Starburst-phone can mistakenly detect the rim of glasses or hair as part of the iris contour. Under insufficient lighting, it also had difficulty detecting the iris-sclera boundary. Since our model selection in Starburst-phone was based on the maximum number of fitted points under RANSAC, with fewer points detected

on the iris contour, the iris fitting will frequently fail. As a result, we also removed Starburst-phone from our considerations. ([37] provides more detailed analysis among these three algorithms.)

After noticing that appearance-based algorithms may be the most robust algorithms, we carefully compared the robustness of iTracker and iTracker-face under a variety of environmental conditions that may be encountered outside the well-controlled clinical setting for eye-movement measurements. We compared the performance of the algorithms on video sequences of subjects with and without glasses and under various ambient lighting conditions under the Validation-Stage recording setup. Two illumination-adjustable LED panel lights were used to vary the illumination during the recording sessions. In total, four distinct lighting conditions were tested: (1) room light switched on in addition to the panel lights set to high (278 Lux); (2) room light switched on without additional lighting support from the LED panels (220 Lux); (3) room light switched off and the panel lights set to medium (54 Lux); and (4) room lights switched off and the panel lights set to low (26 Lux). Illuminance was measured at the participant's face using an LT40 LED Light Meter (Extech Instruments). Figure 3-8 shows how the lighting conditions affect image brightness. Five subjects contributed 120 saccade tasks under each of the four lighting conditions with and without glasses, for a total of eight test conditions per subject.

The video sequences were processed with both iTracker and iTracker-face, and the 9,600 resultant eye-movement traces were each reviewed by two annotators. Same as the previous, each annotator independently determined if a trace represented a horizontal saccade movement and had sufficiently high signal-to-noise to allow for credible saccade-onset determination. Traces that met these criteria were labeled 'good'; all other traces were labeled 'bad'. Traces labelled as 'bad' were typically interrupted by blinks, initially directed toward the opposite direction of stimulus presentation, or had a low signal-to-noise ratio. To assess the annotator agreement, we computed both the accuracy (fraction of annotations in which both annotators agreed) and Cohen's kappa coefficient (κ). The algorithm with the highest fraction of 'good' saccade traces, as judged by both annotators, across the different environmental conditions

Figure 3-8: A sample frame from each video taken under four distinct lighting conditions. From left to right, the pictures are arranged from the highest illuminance (278 Lux) to the lowest (26 Lux).

was deemed the more robust algorithm.

Figure 3-9 reports the inter-rater annotation accuracy, broken down by 'agreed good' and 'agreed bad', for both algorithms and each of the eight environmental conditions tested. The average annotation accuracy was 94.1% for eye-movement traces generated by iTracker-face and 86.8% for iTracker, with corresponding Cohen's κ values of 0.802 and 0.730, respectively. These results indicate excellent inter-rater agreement for the overall annotation task, which means that their judgment can be used as a benchmark. Their annotations also reveal that important trends exist between algorithms and across environmental conditions. The inter-rater agreement is lower when participants wear glasses and tends to decline with decreasing illuminance. For example, at the lowest illuminance level (26 Lux) and with participants wearing glasses, the annotators agreed in their label of 'good' in over 40% of the traces generated by iTracker-face. In contrast, their agreement of what constitutes a good saccade trace was less than 8% of the traces generated by iTracker. Obviously poor illumination conditions result in image sequences with lower contrast which makes it harder to detect eye features and subtle eye movements. A closer inspection of the video sequences also revealed that glasses, especially those with dark rims, tend to cast shadows that can obscure the eye regions. Additionally, some glasses have lenses with high reflectivity that make the eyes even less visible and therefore difficult to track.

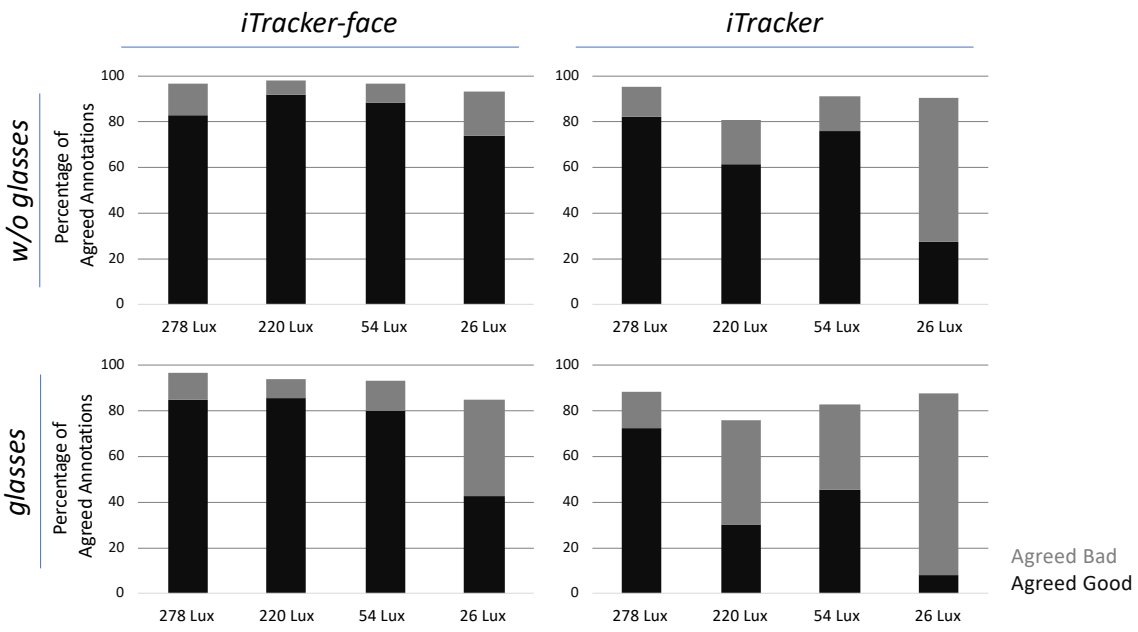

Figure 3-9: Annotation accuracy broken down for each of the eight environmental conditions tested per algorithm. The accuracy (or percentage of agreed annotations) is additionally broken down into the fraction of agreed-good and agreed-bad eye-movement traces between two annotators.

Across all eight conditions tested, the average fraction of traces judged as good by both annotators was consistently and significantly higher for traces generated by iTracker-face (78.9%) than for those generated by iTracker (50.7%). We conclude from this analysis that across all environmental conditions tested, iTracker-face is the more robust algorithm of the two and therefore formed the basis of all subsequent results reported here.

3.1.5 Automation of Eye-Tracking Algorithms

After iTracker-face was chosen as the eye-tracking algorithm, we replaced the manual face crop with a Viola-Jones detector. We show in Appendix B that the latencies we acquired from these two methods were almost identical. (Note that the evaluation was based on the latency measurement presented in the next section. One may need to read the next section before reading the appendix.) Hence, we completed the automation of the eye-tracking building block in the measurement pipeline.

3.2 Saccade Latency Measurement

In this section, we first discuss our model-based algorithm to detect saccade onset under the Validation-Stage setup. We show that this algorithm allows us to evaluate the quality of the latency measurement and identify outliers. With both iTracker-face and a saccade-onset-detection algorithm, we then compare our results with the measurements acquired from a research-grade camera. By showing that we can attain almost identical statistics from mobile devices as from a research grade camera, we have completed our main goal in the Validation Stage – enabling sacacde latency measurements using mobile devices.

As we collected more data from adults with a wider age spectrum in the Deployment Stage, we noticed that the model fit designed in the Validation Stage needs to be improved to account for various eye movements. In particular, we need to re-design how we normalize our eye-tracking results to degrees and how we choose an appropriate window of fit.

3.2.1 Validation-Stage Saccade Onset Detection

To calculate saccade latency, it is necessary to determine the onset of the eye movement toward the target. In prior work, the saccade onset has commonly been defined as an increase in eye velocity above a predefined threshold [15, 16], such as 30 °/s, where the velocity is commonly determined through numerical differentiation and subsequent filtering of the raw eye-position tracing [56]. Such saccade-onset determination requires accurate measurement of gaze and is prone to significant error at low sampling rates [57].

Here, we instead propose to model the eye-position trace during a saccade task as a hyperbolic tangent of the form

$$\tilde{x}(t) = A + B \cdot \tanh\left(\frac{t - C}{D}\right)$$

and fit the model to the the eye-position tracing from 100 ms before to 500 ms after

the stimulus presentation (Figure 3-10). The fitting was performed using the non-linear least-squares solver `lsqcurvefit` in Matlab to estimate the model parameters A, B, C, D. Using these optimal model parameters, we determine the saccade onset as the time when the best-fit solution exceeds 3% of the maximal saccade amplitude, which is independent of the velocity of the saccade.

In addition to generating well-behaved velocity tracings, this model-based approach has the benefit of providing a goodness-of-fit metric on the basis of which the reliability of saccade tracings can be evaluated in an automated manner, as the normalized root-mean-squared error (NRMSE) between the model fit and the eye-position trace quantifies the residual discrepancy between the two. Here, the normalization was done to the saccade amplitude (10° in our experiments). Measurements contaminated by excessive noise, artifact, or eye movements in the wrong direction typically result in a high NRMSE value while reliable measurements result in a low NRMSE. Thresholding the NRMSE allows for automated rejection of recordings in which the saccade onsets might have been erroneously detected or the measurements are subject to excessive variability, noise or artifact.

To evaluate the usefulness of the NRMSE as an automated metric to flag bad saccades, we used the expert-annotator labels in Section 3.1.4 as the ground truth for all iTracker-face derived traces described in the previous section and swept the NRMSE threshold to generate a receiver-operating characteristic (ROC) curve. By separately considering each annotator's judgment as the ground truth, we obtained two ROC curves (Figure 3-11), one for each annotator, and generated associated 95% confidence intervals (CI) by stratified bootstrapping over 2,000 replicates [58]. The two resultant ROCs tracked each other closely and achieved an area under the curve (AUC) of 0.923 (95% CI: 0.913 – 0.932) and 0.933 (95% CI: 0.923 – 0.943), respectively. If we consider all traces with a NRMSE<0.1 as 'good' saccades, we achieve average true positive rates of 0.87 and 0.86 and average false positive rates of 0.20 and 0.16 for the first and second annotator, respectively. In the following, we selected an NRMSE of 0.1 as the threshold.

The automation of saccade onset detection is crucial. We noticed that annotation

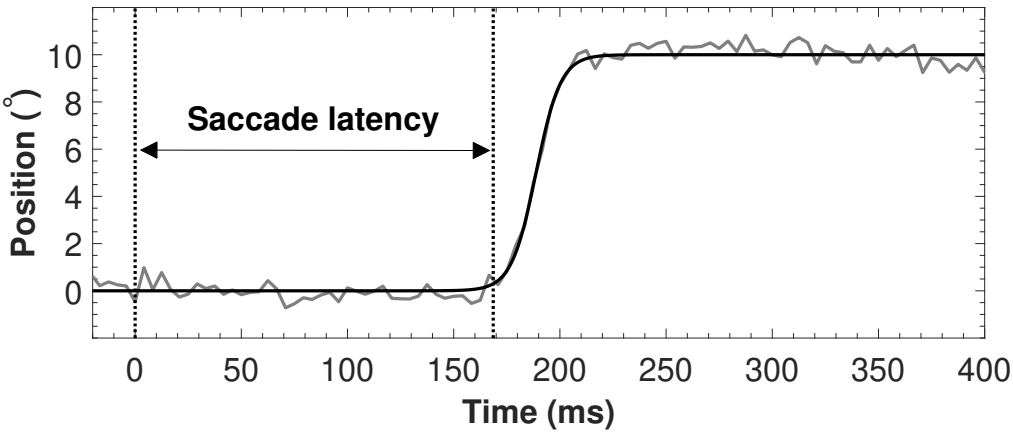

Figure 3-10: Eye position as estimated by the iTracker-face algorithm (gray) and hyperbolic tangent fit (black). The dashed line at 0 s indicates the moment of stimulus presentation. The saccade onset is determined by an increase in saccade amplitude above 3% of the target saccade amplitude.

of the 9,600 eye-movement traces took each annotator about 12 hours to complete. Since our goal is to leverage smartphones to make eye-movement recordings and analyses widely available and ubiquitous, visual inspection of individual tracings is not an option. As shown in [37], the main drawback of the differentiation methods used in the literature is that it cannot automatically identify bad saccades.

3.2.2 Comparison across Cameras

To verify that recordings from mobile devices can lead to similar saccade-latency statistics as those obtained from recordings of high-end, research-grade cameras, we took simultaneous recordings on four subjects using a low-cost, iPhone 6 camera and a research-grade camera (Phantom v2511, see Table 2.1 for their specifications).

Figure 3-12 shows the resulting saccade-latency distributions obtained using the iTracker-face algorithm and the model-based onset detection. The inclusion of the high-speed camera in the recording set-up resulted in increased distances between the subject and the cameras, as well as between the subject and the laptop's screen. The increased distances result in a smaller horizontal eye movement, which in turn produce slightly noisier, but acceptable, eye movement traces. Figure 3-12 demonstrates that

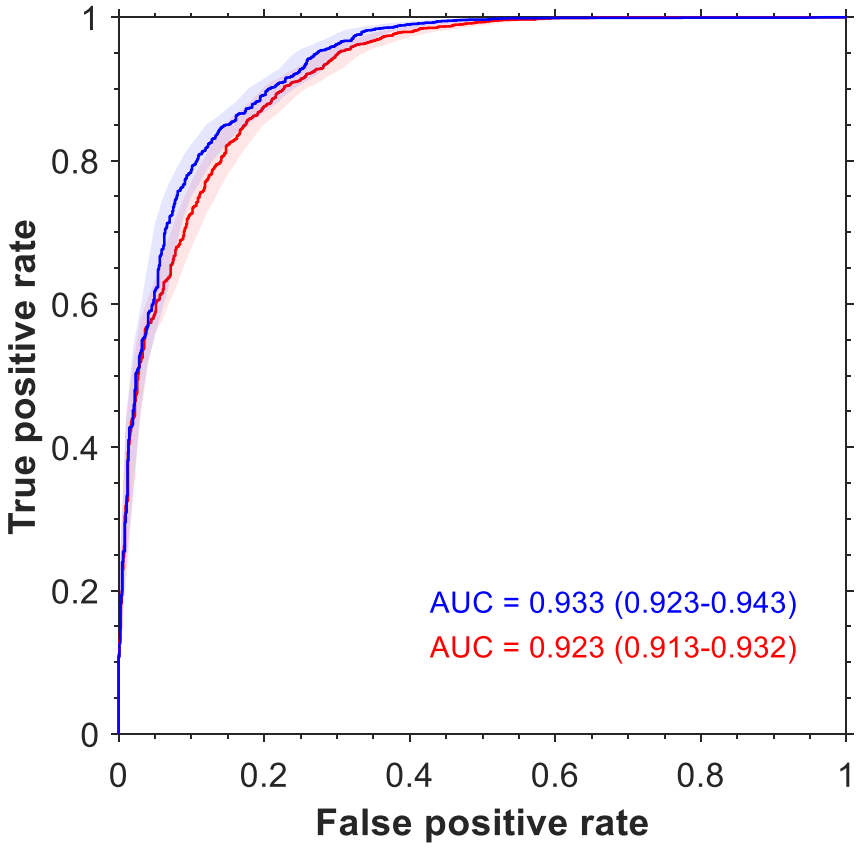

Figure 3-11: Performance of model-based fitting in classifying saccades. The adjudications of two annotators were taken as the ground truth, with the solid lines being the corresponding mean ROC curves. The shaded areas indicate the confidence intervals for the true positive rate. The parentheses mark the 95% confidence intervals for the areas under the curves.

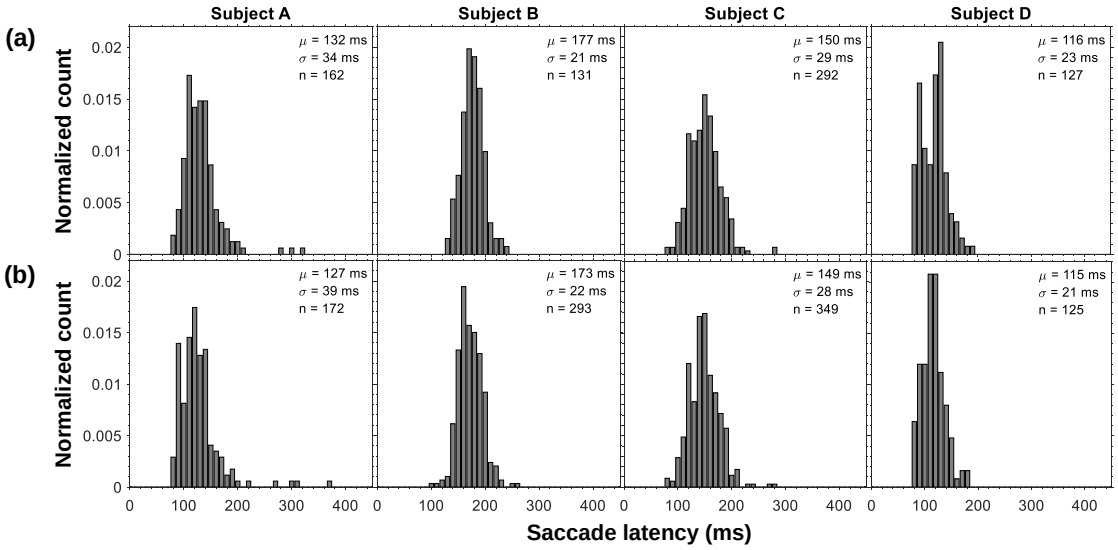

Figure 3-12: Saccade-latency distributions from four subjects obtained from video recordings using (a) the iPhone 6 and (b) a Phantom v2511 high-speed camera.

the distributions from both cameras are consistent, with negligible differences in the mean saccade-latency values and associated standard deviations between the two recording systems.

3.2.3 Deployment-Stage Saccade Onset Detection

In the Deployment Stage, we expanded upon our study cohort in the Validation Stage by specifically including self-reported healthy subjects across the adult age spectrum. Consequently, we observed a larger heterogeneity in saccadic eye-movement patterns that necessitated revisions to the saccade onset detection developed in the Validation Stage.

To allow for latency measurements from subjects with slower response times, we needed to increase the window of fit for the tanh model from 200 ms before to 800 ms after the stimulus presentation. However, we noticed that by expanding the window, it is more likely to capture a subject's eye movements back toward the center position (Figure 3-13a). Additionally, subjects may perform a series of hypometric saccades in which the initial saccadic movement does not reach the final position and a second saccade is made to correct for this undershoot (Figure 3-13b). Correct identification

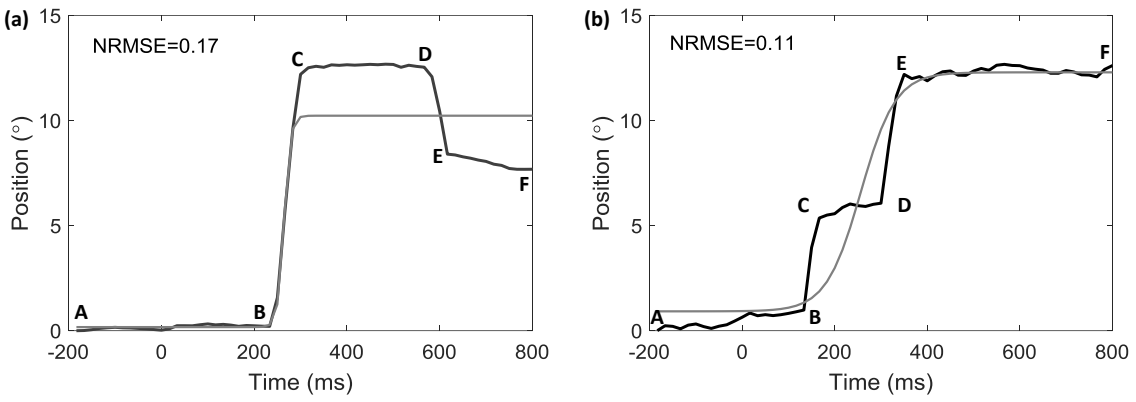

Figure 3-13: Examples where tanh cannot be fitted to the entire trace: (a) gaze returning (b) hypometric saccade [3, 4]. As one can see, to find the saccade latency, the window where we fit a tanh model should be from A to D.

of hypometric saccades is of relevance since an increased incidence of hypometric saccades is associated with certain neurodegenerative pathologies [17, 3]. The single tanh model cannot fit well to these traces if we use a fixed window to determine latency values. To determine saccade latency, we need to allow for an adaptive window of fit for the tanh model to identify the initial saccadic movement to be fitted. We also note that we cannot convert the unit of the eye-movement trace from centimeter to degree using the best fit tanh model. As a result, we needed to revise our method to normalize the trace.

Saccade Normalization

We make three assumptions to convert the unit of the eye-movement traces from the iTracker-face generated centimeter to degree. First, we assume that subjects were looking at the fixation point during the fixation period. Second, we assume that subjects did not overshoot their gaze. Finally, we assume that during the stimulus period, subjects either (a) did not move their eyes at all, (b) gazed at the stimulus, or (c) gazed at the opposite position of the stimulus.

With these assumptions, we normalize the trace as follows. First, to simplify the algorithms, we flip the trace if needed so that positive excursions correspond to eye movements in the correct direction. We then smooth the eye-movement trace with a

Savitzky-Golay filter [59, 60] (of order 3 and frame length 5) to make the final normalization more robust to noise. Subsequently, we determine two reference points to scale and shift the eye-movement trace. Our first reference point is set as the starting gaze position of a trace, that is 200 ms before the stimulus presentation. With the second assumption, our second reference point is either the maximum or the minimum value of the smoothed trace, depending on whether the subject makes a correct saccade, a corrected error, or an uncorrected error. Scaling and shifting coefficients can be found by shifting the first reference point to zero degree and scaling the second to either the final expected amplitude (12.7 degrees) or the negative amplitude (-12.7 degrees).

More precisely, we consider three scenarios. (a) Operating on the output of iTracker-face, if the difference between the maximum value and the starting gaze position is greater than 0.2 cm, we assume that the subjects have made a correct saccade or a corrected error, and we scale the second reference point to the positive expected amplitude value. (b) If the difference between the maximum value and the starting gaze position is smaller than 0.2 cm but the absolute difference between the minimum value and the starting gaze position is greater than 0.2 cm, we assume that the subjects have made an uncorrected error and we scale the second reference point to the negative expected amplitude value. (c) If neither of these criteria is met, we assume that the subjects have made only subtle eye movements or that the eyes were occluded.

In the first two scenarios, we find the scaling and shifting coefficients from the smoothed trace and normalize the original trace using these coefficients. One key observation of this normalization is that after normalization, traces with the same shape will become identical. This characteristic ensures that if the saccade-latency measurement algorithm and the error-detection algorithm are designed using this normalized trace, the algorithms will be scale-and-shift-invariant. That is, eye movement features are measured only based on the shape of a trace. In the third scenario, we noticed on visual inspection of the video recordings that the sizes of the eye movement were often comparable with noise and subtle head movement. To account for such observa-

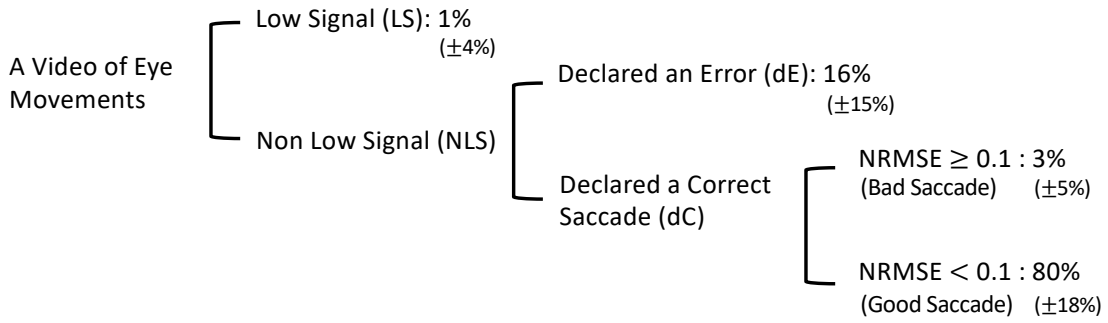

Figure 3-14: Breakdown of saccades collected in the Deployment Stage into error saccades, good saccades, bad saccades, and "LS" (low signal).

tions, we label such traces as "LS" (Low Signal) to acknowledge the fact that we are uncertain whether there is an actual eye movement even by visualizing the original videos. Traces labeled LS will be excluded from the saccade-latency measurement and the error-detection algorithm.

Adaptive Window of Fit

With the normalization, we next describe how we updated the window of fit for saccade latency measurement. Returning to the examples in Figure 3-13, during the period from A to B and C to D, the subject's eyes are fixated. During the period from B to C, the subject performed a correct saccade, in the sense that the eyes moved in the correct direction. As a result, the proper window of fit is the first sequence of fixation, directionally correct eye movement, and fixation. This period can be identified using the velocity of the gaze. We estimate the velocity of the gaze by computing the first-order derivative of the Savitzky-Golay filtered trace to avoid amplifying high-frequency noise.

We then classify a sequence of time instances as a correct saccade period if the velocity values cross 30 degrees/s, as an incorrect saccade period if the velocity values cross -30 degrees/s, and as a fixation period otherwise. When there are more than one correct saccade periods, we will fit our model to the one that first crosses a third of the amplitude. Figure 3-15 shows that by choosing the window of fit to be the period associated with the sequence of fixation, directionally correct eye movement,

and fixation, we can fit the tanh model to traces with multiple transitions and measure their saccade latencies. We compared the previously described fixed-window approach with the adaptive-window approach and observed that the proportion of saccades with a NRMSE > 0.1 dropped from 17% to 3% with the adaptive-window approach. Hence, by moving to the adaptive-window approach, we were able to compute significantly more latencies with this improved saccade-latency measurement algorithm.

3.3 Error Rate Measurement

In the Deployment Stage, we also extended our eye-movement features to include error rate. Here, we present how we designed a robust error-detection algorithm and appropriately defined error rate that takes into account the possibility that it might not always be possible to determine the direction of an eye movement from app-based recordings.

3.3.1 Error Detection

In the clinical literature, a directional error is defined as an initial eye movement in the wrong direction [52]. Manual annotation is often involved in the determination of these errors [17, 16]. Because such clinical studies have traditionally relied on specialized environments and eye-tracking equipment, including use of chinrests, infrared illumination, and research-grade cameras, there were usually comparatively few traces collected per subject and the traces tended to be clean. As a result, manual annotation of traces is possible in these cases. In contrast, to enable collection of large amounts of data, we use mobile-device cameras and do not use a chinrest. As a result, we obtained significantly many more traces, though some were affected by glares or head movements. Our goal is thus to reject poor recordings and develop an accurate and robust error detection algorithm.

As mentioned in Section 3.2.3, we exclude the traces labeled LS, since we cannot distinguish between saccadic eye movements and noise/head movements. Out of the remaining traces, we noticed that a typical error trace shows a period of fixa-

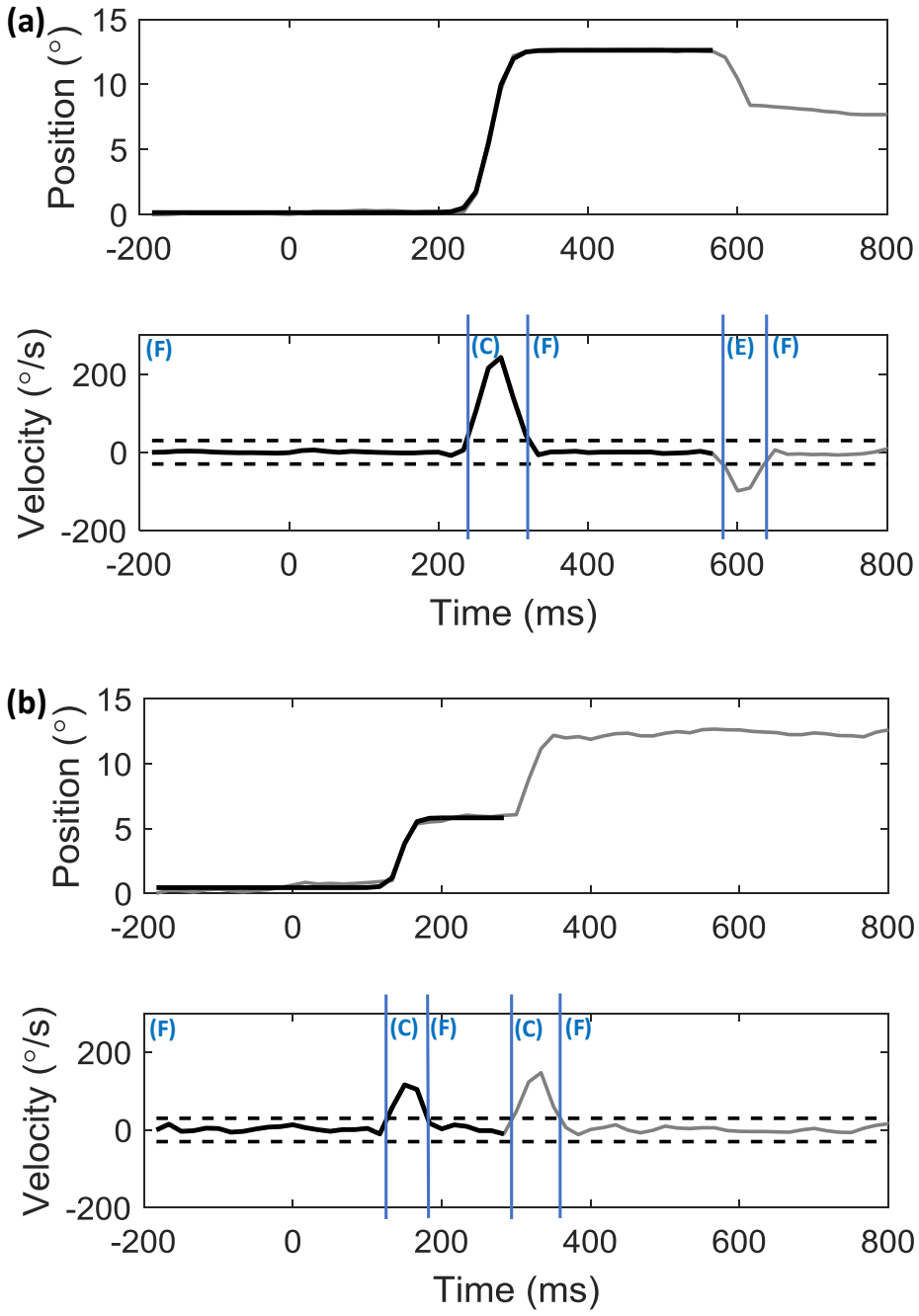

Figure 3-15: Tanh fitting example: (a) gaze returning (b) hypometric saccade. The top panels show the eye movement traces obtained from iTracker-face after normalization. The dark lines show the fitted hyperbolic tangent models. The bottom panels show the velocity of the eye movements and the velocity threshold (the dash lines). With such a threshold, we label different parts of the trace as fixation (F), correct saccade (C), or error saccade (E). The window of fit is chosen as the first "fixation(F)-correct saccade(C)-fixation(F)" period that crosses a third of the amplitude.

tion followed by a directionally incorrect eye movement (as shown in the top panel of Figure 3-16). Since our goal is to detect such a change, we developed our algorithm based on the change detection literature [61]. In particular, we extended the cumulative sum (CUSUM) algorithm [62] for our purposes.

We first assume that our measured eye movement trace x_t at time t is composed of an eye movement θ_t and an additive measurement noise ϵ_t. We then use a recursive least square filter to estimate the eye movement $\hat{\theta}_t$ according to

$$\hat{\theta}_t = \lambda\hat{\theta}_{t-1} + (1-\lambda)x_t, \tag{3.1}$$

where λ determines how much the estimation $\hat{\theta}_t$ relies on the current data point x_t versus the past data. The residual error then becomes $\hat{\epsilon}_t = x_t - \hat{\theta}_t$. If there is neither a positive trend nor a negative trend in x_t, $\hat{\epsilon}$ will be centered around zero. As a result, when we consider the cumulative sum of the residual error $s_t = s_{t-1} + \hat{\epsilon}_t$, s_t will be centered around zero as well. However, if there is a negative trend in x_t as shown in Figure 3-16, s_t will become progressively more negative. We can then use a threshold to determine whether s_t is sufficiently negative such that ϵ_t is unlikely to just represent additive measurement noise.

To distinguish between correct saccades and incorrect saccades, we define two separate variables for s_t: $gn_t = \max\{gn_{t-1} - \hat{\epsilon}_t, 0\}$ and $gp_t = \max\{gp_{t-1} + \hat{\epsilon}_t, 0\}$. That is, gn_t accumulates negative trends and gp_t accumulates positive trends. As a result, when gn_t and gp_t cross the pre-determined threshold, we detect an incorrect and a correct saccade, respectively. To apply the definition of a directional error as an initial eye movement towards the wrong direction, we detect an error if gn_t crosses the pre-determined threshold after 0 ms and before gp_t crosses the pre-determined threshold.

Here, we chose to scale the threshold with respect to the estimated (corrected) saccade amplitude. We notice that if there is no error in a trace, gn_t will be around zero while gp_t will approximate the amplitude of the saccade (Figure 3-16). When there is an error, gp_t will approximate the amplitude of the corrected saccade. On

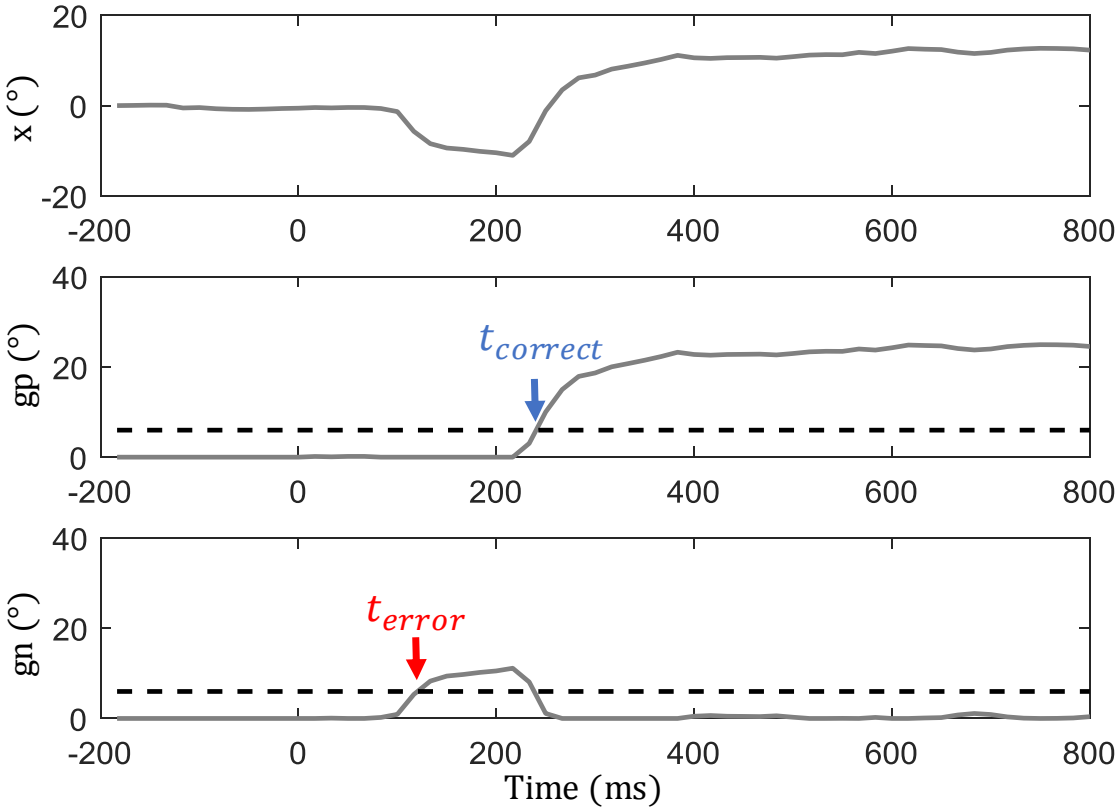

Figure 3-16: Error detection example. The top panel shows the x coordinates of the iTracker-face output over time (x_t). The middle and the bottom panel show gp_t and gn_t. The dashed line indicates the threshold T. When gp_t and gn_t cross the threshold T, $t_{correct}$ and t_{error} are detected, respectively. In this case, since $0 < t_{error} < t_{correct}$, an error is detected.

the other hand, when there is an uncorrected directional error, gn_t will approximate the amplitude of the saccade. As a result, we approximate the (corrected) saccade amplitude by $max_t\{gp_t, gn_t\}$. We further observe that if the saccade amplitude before the normalization is sufficiently large, the saccade will be less affected by head movement and noise. Thus, we can consider lowering the threshold to detect smaller errors. On the other hand, if the original saccade amplitude is closer to the size of the head movement and noise, the threshold needs to be sufficiently large to avoid artifacts from being detected. Recall that in Section 3.2.3, we scale the trace and shift it to normalize it from centimeters to degrees. We can use the scaling coefficient (denoted as B in **Algorithm**) as a metric to evaluate the size of the original saccade amplitude. If B is small (< 8), it means that the original amplitude is large and the threshold could be smaller. If B is large (≥ 8), we will use a fixed threshold. Here, the value 8 can be considered as a hyperparameter that we can tune. The final threshold is $\max_t\{gp_t, gn_t\} \cdot \min\{B, 8\} \cdot T$. The complete algorithm is shown in **Algorithm**.

To determine the threshold T, we asked four subjects to perform six anti-saccade tasks of 40 stimuli each. Two expert annotators reviewed the videos and annotated the directional errors. Out of the $4 \cdot 6 \cdot 40 = 960$ saccadic eye movements, there were only two disagreements between the annotators which were resolved after these two disagreements were reviewed together. With the annotated data set at hand, we swept the threshold T and determined the true positive and false positive rates for detecting a directional error (Figure 3-17). When the threshold is lower than the noise level, gp_t and gn_t may cross the threshold due to noise rather than a saccadic eye movement. That is, gp_t may be equally likely to cross the threshold as gn_t. Recall that we only detect a trace as an error if gn_t crosses the threshold before gp_t. As T goes to zero, the true positive rate and the false positive rate go to 0.5. On the other hand, if the threshold is too large, the amplitude of an incorrect saccade may be smaller than the threshold and the error may not be detected. When T is larger than the noise level but smaller than the amplitude of an error, we can get high sensitivity and specificity. By choosing $T = 0.03$, we can achieve a sensitivity of 0.97 and a specificity of 0.97 for detecting a directional error.

Algorithm: Error Detection

input : $x = [x_1, \ldots, x_N], B$, x_1 is chosen to be the first instance after the stimulus presentation, B is the scaling coefficient in the saccade normalization

output : $t_{error}, t_{correct}$ (An error is only detected if the first element in t_{error} is smaller than the first element in $t_{correct}$.)

parameter: λ, T

for *round=0:1* **do**

  $\hat{\theta} = x_1, t_{error} = [], gn = [0], gp = [0]$;

  for *t=2:N* **do**

    $\hat{\theta} = \lambda\hat{\theta} + (1-\lambda)x[t]$;

    $\hat{\epsilon} = x[t] - \hat{\theta}$;

    $gn.append(\max\{gn[t-1] - \hat{\epsilon}, 0\})$;

    $gp.append(\max\{gn[t-1] + \hat{\epsilon}, 0\})$;

    if *round==1* **then**

      if $gn[t] > A \cdot T$ **then**

        $t_{error}.append(t)$;

        $gn[t] = 0$;

        $\hat{\theta} = x[t]$;

      end

      if $gp[t] > A \cdot T$ **then**

        $t_{correct}.append(t)$;

        $gp[t] = 0$;

        $\hat{\theta} = x[t]$;

      end

    end

  end

  $A = \min\{8, B\} \cdot \max\{gp, gn\}$;

end

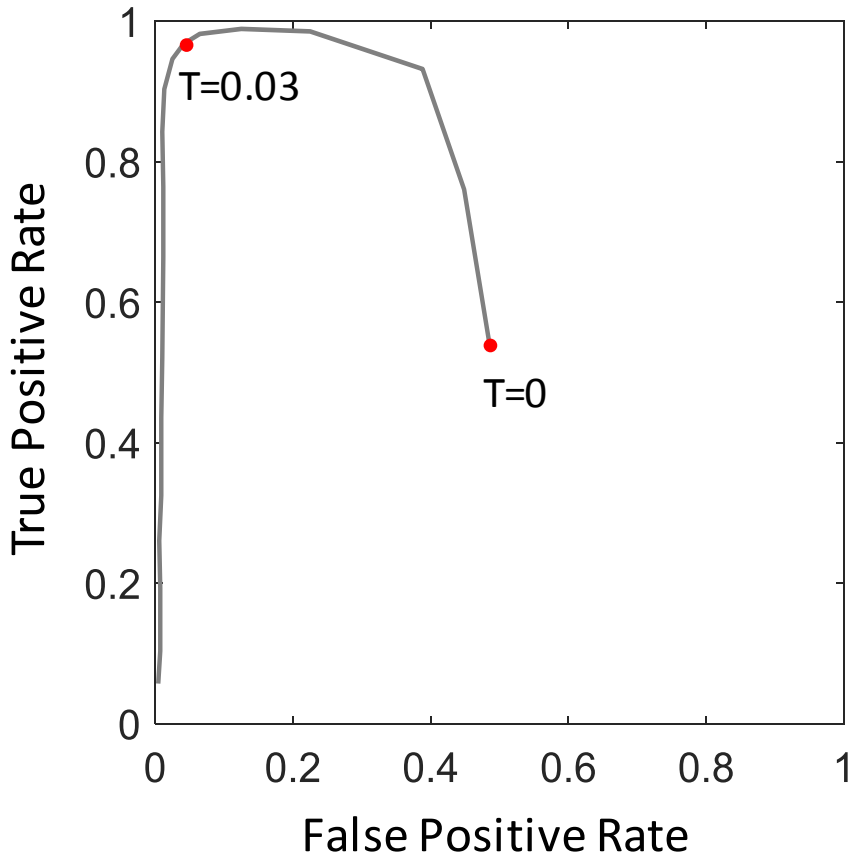

Figure 3-17: The true positive rate and the false positive rate as we increased the error detection threshold T from 0 to 0.1. We chose $T = 0.03$ as our final threshold to achieve a sensitivity of 0.97 and a specificity of 0.97.

3.3.2 Error Rate Definition

In the clinical literature, error rate is often defined as the proportion of errors, though it is not usually discussed whether noisy traces are excluded from such calculation. Given the use of special-purpose equipment and optimized environmental conditions in clinical research studies, such recordings may have very few noisy traces. Without a chinrest and a controlled laboratory setup, we obtained more noisy traces. We carefully identified the causes of these noisy traces: glares, head movements, eyelids drooping. Many of these causes could be reduced with more careful instruction. However, even with careful instruction, it is hard to eliminate all these causes, due to the nature of the much more relaxed and varying recording environment and the large number of recordings. As a result, it is important to define an error rate that takes these noisy traces into consideration.

An eye movement was either declared a correct saccade (dC), declared an error (dE), or labeled low signal (LS). If we define the error rate as the proportion of errors out of all the traces, we might significantly underestimate the error rate in records with a lot of eye movements in the LS category. A better approach might be to define the error rate in a recording as #dE/(#traces-#LS), as explained in Eq. (3.2). The question then arises under which conditions the error rate so defined approximates the (empirical) probability of an error.

Under the assumption that

- $P(dE|C) \approx 0, P(dC|E) \approx 0,$

- $P(LS|E) \approx P(LS|C),$

where E denotes errors and C denotes correct saccades, we can express the error rate

as

$$
\begin{aligned}
&\frac{P(dE)}{1-P(LS)} \\
=&\frac{P(dE|E)P(E)+P(dE|C)P(C)}{1-P(LS|E)P(E)-P(LS|C)P(C)} \\
\approx&\frac{P(dE|E)P(E)}{P(E)[1-P(LS|E)]+P(C)[1-P(LS|C)]} \\
\approx&\frac{[1-P(LS|E)]P(E)}{[1-P(LS|E)]P(E)+[1-P(LS|C)]P(C)} \\
\approx&\frac{P(E)}{P(E)+P(C)} \\
=&P(E)
\end{aligned}
\tag{3.2}
$$

where we made use of the fact that a trace is either an error or a correct saccade, i.e., $P(E) + P(C) = 1$. The first assumption states that the false positive and the false negative are essentially zero. As discussed in Section 3.3.1, our error detection algorithm achieved a sensitivity of 0.97 and a specificity of 0.97. Therefore, the first two assumptions are indeed met. The second assumption states that a correct saccade is equally likely to be declared LS as an error saccade. Since our determination of LS is simply based on the size of the trace, this condition is met as well. Therefore, it is reasonable to define the error rate as #dE/(#traces-#LS) as an estimate of the (empirical) probability of an error.

3.4 Discussion and Summary

In this chapter, we present our measurement pipeline – the eye-tracking algorithm, the algorithm to measure saccade latency, and the algorithm to measure error rate. Several technological challenges needed to be overcome to allow for these eye-movement measurements outside a specialized clinical environment. Among these technological challenges were the reliability on infrared (IR) light to estimate the position of the eye and the use of research-grade cameras that yield distinct images of the eyes.

To extract the position of the gaze from each frame in a video sequence, we proposed iTracker-face, a modified version of a deep convolutional neural network for gaze estimation on smartphones that does not rely on IR illumination. In our appli-

cation, iTracker-face is more robust to lower image quality than iTracker, providing eye-movement traces with a higher signal-to-noise ratio. Once the eye-movement traces are extracted with iTracker-face, our eye movement model is fitted to the individual traces to determine the onset of the eye movement toward the target. This model-based approach has the added benefit of providing a goodness-of-fit metric that allows for automated rejection of unreliable data, an instrumental contribution toward making saccade latency determination broadly available as large cohorts of patients and self-reported healthy subjects start recording saccadic eye movements on a continuous basis.

Because the environmental conditions outside of a typical clinical setting are variable, the evaluation of the robustness of our eye-tracking algorithm is paramount and strengthens our ability to measure saccade latency in complex real-world scenarios. Our robustness evaluation shows that iTracker-face was consistently and significantly more robust than iTracker across all testing conditions, as ascertained by two annotators that manually reviewed 9,600 eye-movement traces. Because the agreement between annotators was high (as given by the accuracy and Cohen's kappa coefficient), their annotations were used to determine an optimal threshold value for the NRMSE that automatically eliminates eye-movement traces that provide unreliable saccade latency estimates. Our evaluation of the sensitivity and specificity of this approach suggests very high sensitivity and specificity for automated signal quality determination compared against human annotators, and in a variety of environmental conditions that are expected to be encountered in everyday recordings.

After we demonstrated that we can measure pro-saccade latency outside a clinical environment, we further improved our algorithms to measure both saccade latency and error rate from app-based recordings from a much larger population. Our first observation is that in cases where eye movements are too small in amplitude or when the eyes are occluded, the eye movement signals can be smaller than noise. In these cases, we cannot tell the direction of the eye movement either from the trace or from the original video. As a result, we cannot classify these traces into a correct or an erroneous eye movement and cannot determine the saccade onset. We show that we

can identify these traces using the raw output of iTracker-face, label these traces as the "LS"s (low signal), and exclude them from the saccade latency measurement and error detection.

Our second observation is that, since we now implement both pro- and anti-saccade tasks and that anti-saccade latencies are usually larger than pro-saccade latencies, we need to increase the size of the window where we fit our tanh model. However, by doing so, we also increase the potential of including more than one saccade movement in the window. For example, subjects may make a hypometric saccade or return their gaze towards the center of the screen. Being able to measure saccade latency from these traces is crucial, especially when these eye movements indicate a certain phenotype. For instance, patients with Parkinson's disease may make more hypometric saccades [3, 17] than patients age-matched controls. Our previous saccade latency measurement algorithm cannot measure latencies from these traces since a tanh model with a fixed window cannot fit well on these traces. Here, we show how we can find the appropriate windows of fit for these traces and thus enable saccade latency measurement. By doing so, we keep 96% of the traces to be either a good saccade (the saccade with NRMSE ≤ 0.1) or an error saccade, which is much more than 82% of the traces to be either a good saccade or an error if we use a fixed window.

Our third observation is that, to detect directional error is the same as to detect a change in the negative direction in an eye-movement trace. We extend the CUSUM algorithm for this purpose and show that our error detection algorithm can achieve a sensitivity of 97% and a specificity of 97%. Our final observation is that, given the absence of infrared illumination, high-speed cameras, and chinrests, there may be more LSs and bad saccades (saccades with NRMSE > 0.1) in recordings where the subject did not record themselves properly or had several head movements. If we still define the error rate as the proportion of errors out of all the saccades as in the clinical literature, we may underestimate the error rate. As a result, after discarding undesirable recordings (recordings with more than half of the saccades being LSs or bad saccades), we define the error rate as the proportion of errors excluding LSs and

show that this definition is a reasonable approximation for the error rate used in the clinical literature.

All in all, we concluded that we have automated saccade-latency and error-rate measurements from app-based recordings. This achievement enables us to analyze these eye-movement features in Chapter 4.

Chapter 4

Characterization of Eye-movement Features

The motivation of our work is to track individual eye-movement features over time and analyze the correlation between these features and disease progression. To achieve this goal, we need to first understand how eye-movement features change over time in healthy subjects. With the recording system and algorithms demonstrated in Chapter 2 and 3, we have collected longitudinal eye-movement features from healthy subjects. We can then use these data to identify characteristics that are unrelated to disease progression and may become confounding factors. After that, we can design an individualized model that can incorporate these characteristics and may be extended to a disease-progression model so that our ultimate goal can be achieved.

4.1 Eye-movement Characteristics

In this section, we first summarize our data collection efforts. From these data, we notice significant intra- and inter-subject variability. With the belief that all our subjects are healthy, we can assume that the variability is not caused by different disease states. Thus, it is important to characterize the variability in healthy subjects so that we can identify the variability caused by disease progression once we have data from patients. To do so, we study subjects' eye-movement distributions, the

day-to-day variations of their eye-movement features, and the correlation across their eye-movement features. We also analyze how age may affect these eye-movement features. With a better understanding of the eye-movement features, we can then develop an individualized model that characterizes these features in Section 4.2.

4.1.1 Data Collection Summary

In this section, we summarized our data-collection efforts in the Validation Stage and the Deployment Stage.

Validation Stage

In the Validation Stage, we recorded 19,200 saccadic eye movements across 160 experimental sessions in 29 self-reported healthy subjects (20 males, 9 females; median age: 27 years; age range: 22–64 years), including five or more repeat recording sessions in a subset of eleven subjects. In two recording sessions, the Viola-Jones algorithm failed to detect the face of the subject, so the results presented here are based on 158 experimental sessions in 29 subjects.

When we aggregated the saccade latency measurements greater than 80 ms and NRMSE$<$0.1 for each subject, the mean latencies across the 29 subjects typically ranged from 120 ms to 200 ms (Figure 4-1), with one subject having a mean saccade latency of 290 ms. (Review of the latter subject's video sequences, eye-movement traces, and health questionnaire did not provide a credible reason to exclude this subject from our analysis.) While it is common practice in clinical studies to only report the population mean or median saccade latency, such aggregation results in loss of information encoded in each subject's full saccade latency distribution.

Deployment Stage

With a more flexible system in the Deployment Stage, we have collected 6,823 videos and 236,900 eye movements from 80 subjects across the adult age spectrum. We observe that in videos with a substantial number of LSs, subjects' eyes were often

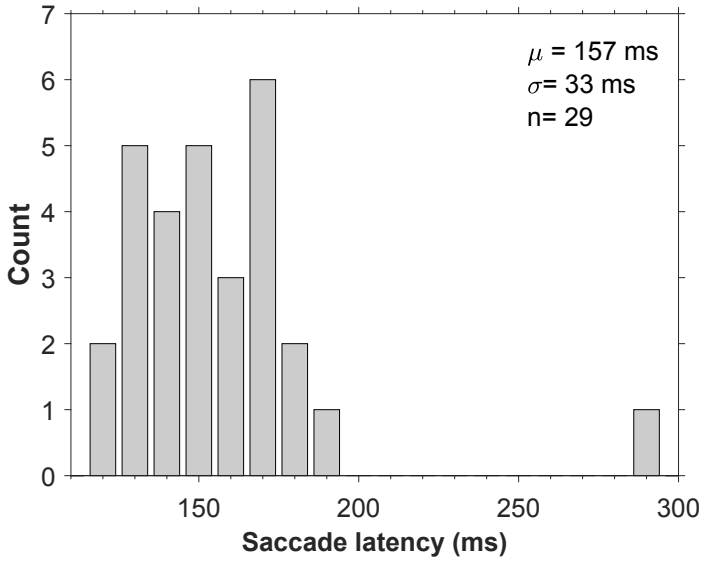

Figure 4-1: Distribution of the mean saccade latencies in the Validation Stage from 29 self-reported healthy individuals, including one subject whose mean saccade latency is 290 ms.

partially occluded due to eyelid droop. Videos with a large number of bad saccades tend to contain more head movements. As a result, the number of LSs and bad saccades indicates whether a subject recorded themselves properly. We therefore discard a video if more than half of the saccades are LSs or bad saccades. After discarding the videos with too many LSs and bad saccades, we retained 6,787 videos and 235,520 eye movements from 80 subjects. Out of the remaining videos, we calculated the mean (standard deviation) of the proportions of each label in a video. There are 1% (4%) of LSs and 3% (5%) of bad saccades. That is, on average, 96% of the saccades are good saccades or declared errors. We noticed that, since in the Deployment Stage we also implemented anti-saccade tasks, the proportion of errors is much larger than the proportion in the Validation Stage. Moreover, with the improvement of the saccade-latency measurement, the proportion of bad saccades is much lower.

As shown in Figure 2-6, there are also more subjects with multiple recording sessions in the Deployment Stage than in the Validation Stage where there were 11 subjects. In Figure 4-2, we show the distribution of the number of days of recordings per subject with multiple recording sessions.

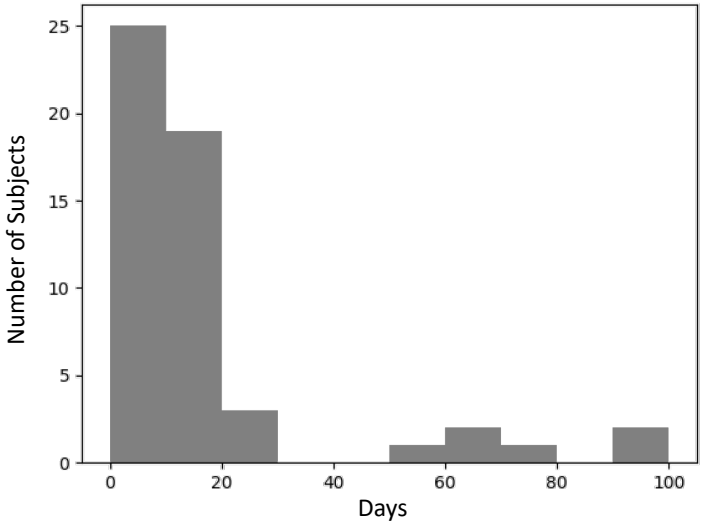

Figure 4-2: Distribution of the number of days of recordings from subjects with multiple recording sessions.

4.1.2 Individual Distribution Modeling

With the accessibility to sizable data, we can study individual distributions, instead of only reporting the population mean as in most clinical literature. Figure 4-3 shows normalized saccade-latency distributions for five subjects measured in the Validation Stage. These subjects were selected to illustrate the range of intra- and inter-subject variation among our study cohort. The distributions show variable degrees dispersion and skewness, with some subjects having a significant fraction of latencies above 200 ms.

It has been suggested that reaction times follow log-normal distributions [63]. We tested this hypothesis on our recordings by fitting a log-normal distribution to the saccade latency distributions of the individual recording sessions, and also to the saccade latency distribution of each subject for which we aggregated each subject's measurements across recording sessions. The log-normal distributions were truncated at 80 ms to reflect the censoring we imposed on the minimum saccade latency. The details of the log-normal fitting was described in Appendix C. The Kolmogorov-Smirnov test was used with the significance level set to 0.05 to test the null hypothesis that the saccade-latency distributions can be described by a truncated log-normal distribution. Of the 158 individual saccade-latency distributions (one for each recording session)

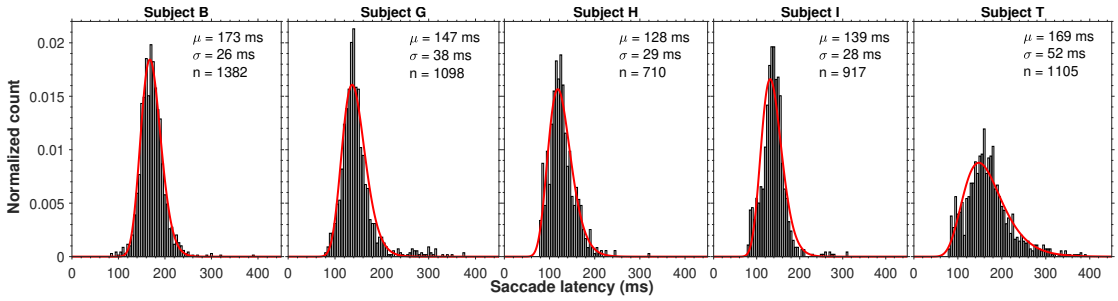

Figure 4-3: Saccade latency distributions for five self-reported healthy individuals from the Validation Stage. μ is the sample mean, σ is the associated sample standard deviation, and n is the total number of observations. Saccade latencies below 80 ms were censored. The estimated log-normal probability density functions are shown in red.

across all subjects, 155 (or 98.1%) distributions were not significantly different from a log-normal distribution ($p<0.05$). When the data from across different recordings sessions were aggregated into a single distribution for each subject, 26 out of the 29 (89.7%) distributions were not significantly different from a log-normal distribution ($p<0.05$).

4.1.3 Day-to-day Variations

Besides the variations in the distributions, there are also day-to-day variations. To analyze these variations, we group the measurements by days. For each day of measurements, we calculate four eye-movement features – median pro-saccade latency, pro-saccade error rate, median anti-saccade latency, and anti-saccade error rate. We use the median rather than the mean to reduce the impact of outliers. We then can estimate the day-to-day variations by calculating the standard deviation of these daily eye-movement features. Figure 4-4 shows the distribution of the standard deviations from subjects with more than five days of recordings. We notice significant inter-subject variability in the day-to-day variations. Since we assume that these day-to-day variations are not caused by disease progression, we aim to examine potential sources of these day-to-day variations as these sources may be confounding factors to disease progression.

The day-to-day variations can be introduced by measurement errors, changes in

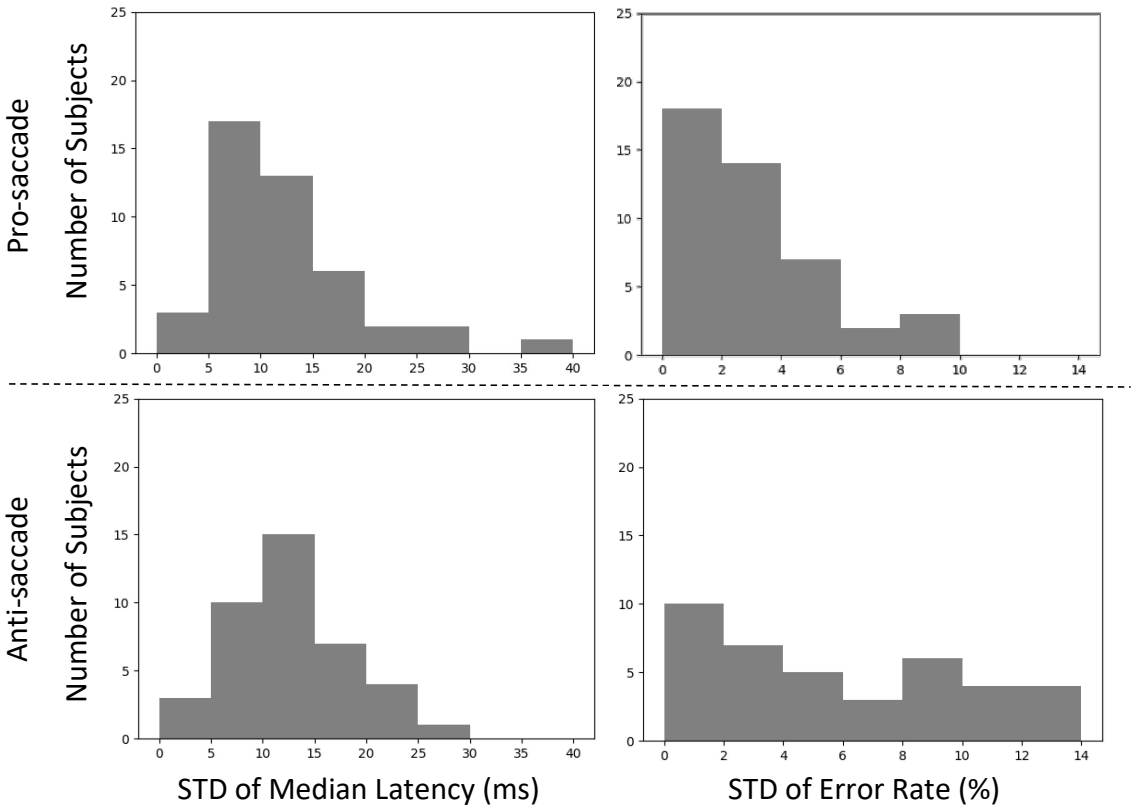

Figure 4-4: The histogram of the standard deviation of four daily eye-movement features – pro/anti-saccade latency/error rate from subjects with more than five days of recordings.

the task-performing strategies, and fatigue effects. As discussed in [64], measurement errors can be classified into random errors and systematic errors. Since random errors affect the measurements of each saccade randomly, they only contribute to the variations within a day. On the contrary, systematic errors bias all measurements in a recording session, and thus these errors contribute to the day-to-day variations. These systematic errors can, for example, be caused by the differences in the recording setup. To analyze the effect size of random errors, we can use bootstrapping to estimate the variations within a day. Figure 4-5 shows the four eye-movement features over days from two example subjects with the 95% confidence interval estimated by bootstrapping. We see that the variations across days are larger than the variations within a day. Therefore, we know that random errors cannot fully explain the day-to-day variations.

Besides systematic errors, the day-to-day variations can also be caused by a subject's task-performing strategy. This effect can be illustrated by Subject 4 in Figure 4-5. We observe that the trajectories of pro/anti-saccade latency are similar. The trajectories of pro/anti-saccade error rate are also similar. However, the trajectories of latency and error rate are opposite to each other. More precisely, we notice that latencies measured around Day 35 are larger whereas error rates measured around Day 35 are smaller. We hypothesize that the subject was trading-off between accuracy and speed when performing the tasks. That is, by moving their eyes faster, a subject may attain a lower latency and a higher error rate, and vice versa. However, not every subject has a clear strategy. As shown in Figure 4-5, Subject 5's strategy is not as clear as Subject 4's. A strategy naturally introduces correlation across features, which we analyze in the next section. Studying these correlations can help us understand how strategies vary across subjects.

Next, we study fatigue effects on day-to-day variations. As mentioned in Section 2.2, after each recording, the app asks the subject to answer how tired they felt on a score from 1 (not tired at all) to 5 (very tired). Similar to the eye-movement features, we can group the recordings within a day and calculate the median fatigue level. However, as shown in Figure 4-6, 4-7, 4-8, 4-9, these scores do not seem to cor-

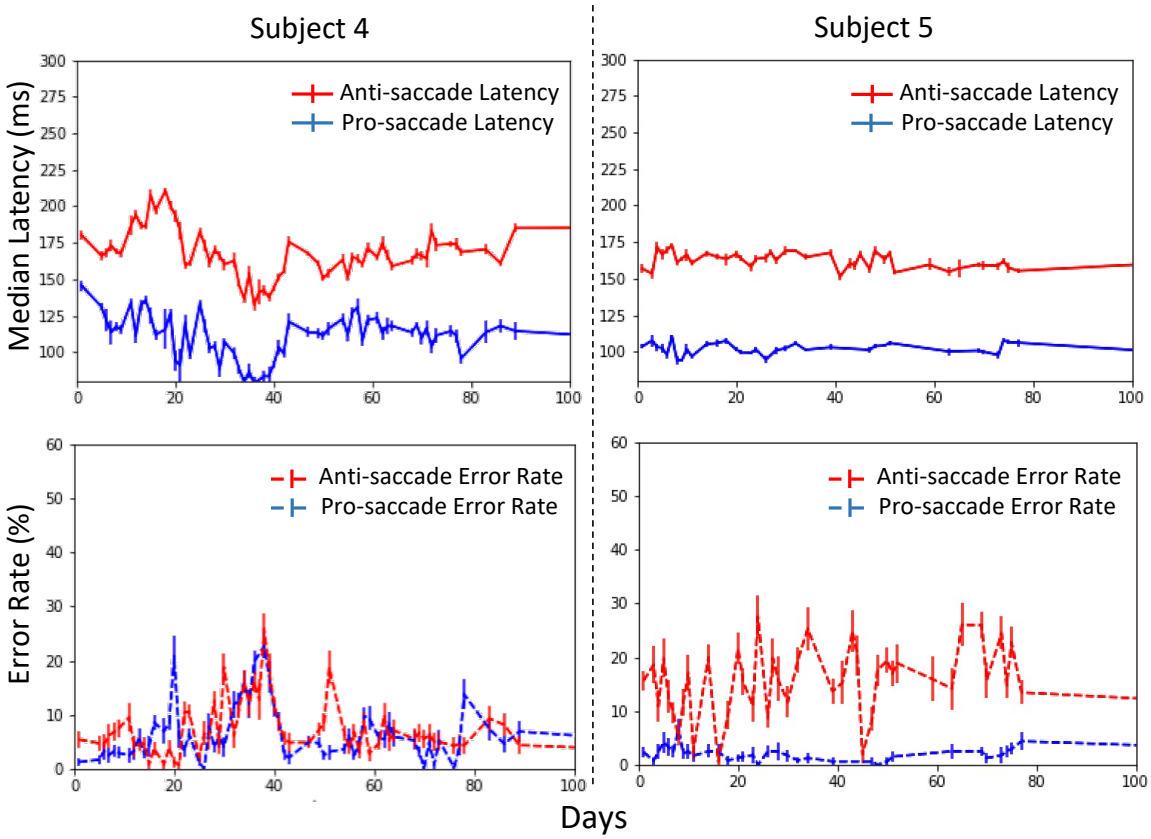

Figure 4-5: Median saccade latency and error rate over days from two subjects. The error bars indicate 95% confidence intervals. Here the index numbers for the subjects follow the experiment result shown in Figure 4-14 where Subject 4 and 5 are the subjects with the fourth and fifth most data in the experiment.

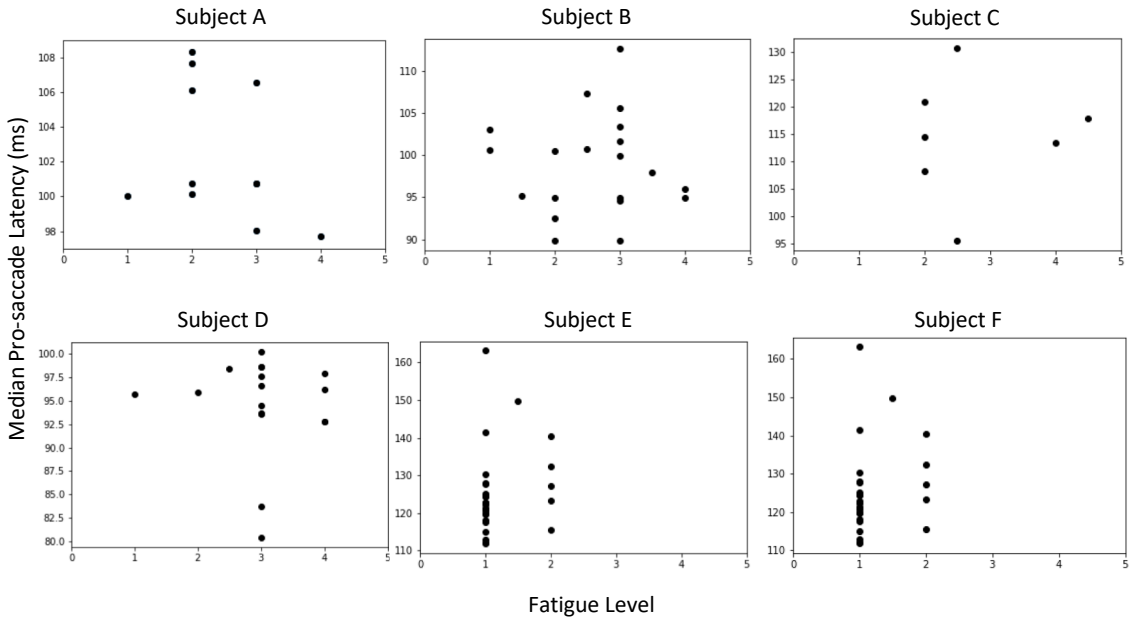

Figure 4-6: The daily median pro-saccade latency and the corresponding median self-reported fatigue level from six example subjects.

relate with any eye-movement features. Nevertheless, for several reasons, we cannot conclude that tiredness has no effect on eye-movement features. First, some subjects may interpret the tiredness as mental exhaustion while others interpret it as the soreness in the eyes. In addition, there may be confounding factors. For example, some subjects mentioned that their minds wandered off while performing the tasks. Therefore, we may need to consider asking subjects how focused they are during the task instead. Third, we notice that subjects tend not to choose 5 (very tired). Therefore, we hypothesize that a score from 1 to 3 might be more indicative.

Finally, we notice that in Subject 4 in Figure 4-5, the eye-movement features change gradually over time. It suggests that there is correlation across time. Systematic errors, task-performing strategies, and tiredness all may be the cause of this correlation. By contrast, the anti-saccade error rate in Subject 5 in Figure 4-5 changes more abruptly. The individualized model we develop should be able to learn these different characteristics.

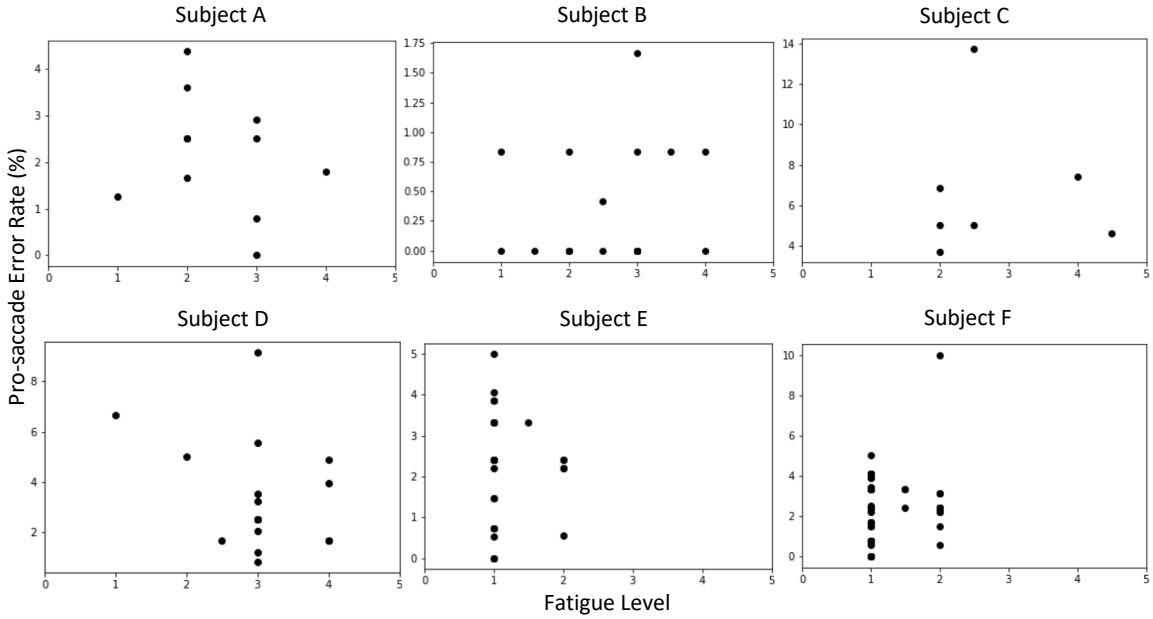

Figure 4-7: The daily pro-saccade error rate and the corresponding median self-reported fatigue level from six example subjects.

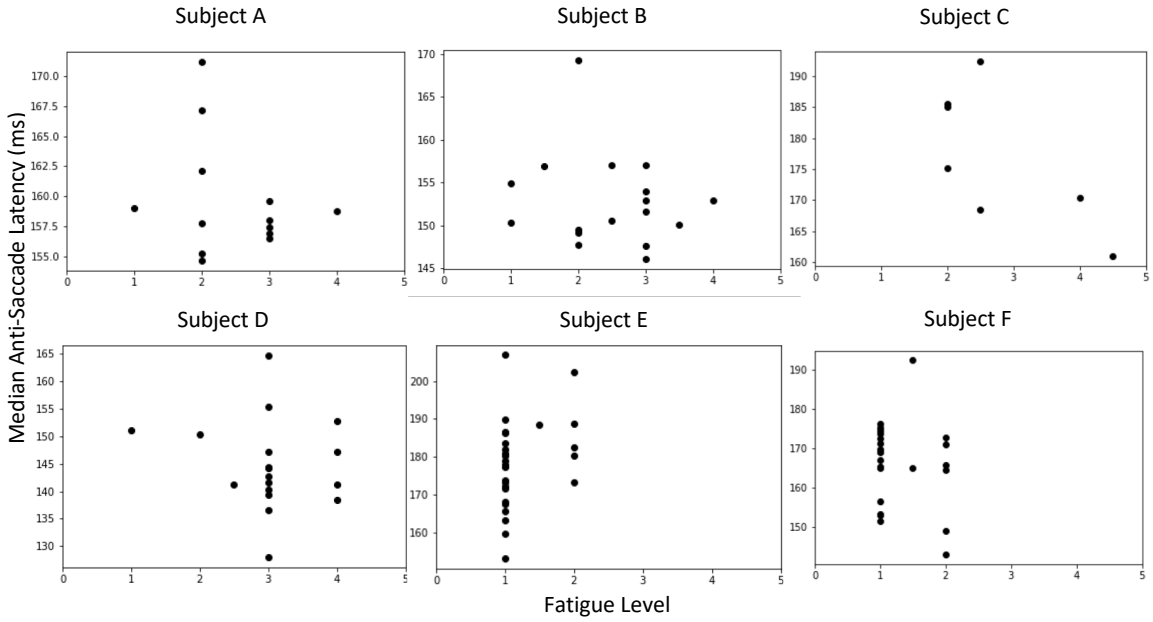

Figure 4-8: The daily median anti-saccade latency and the corresponding median self-reported fatigue level from six example subjects.

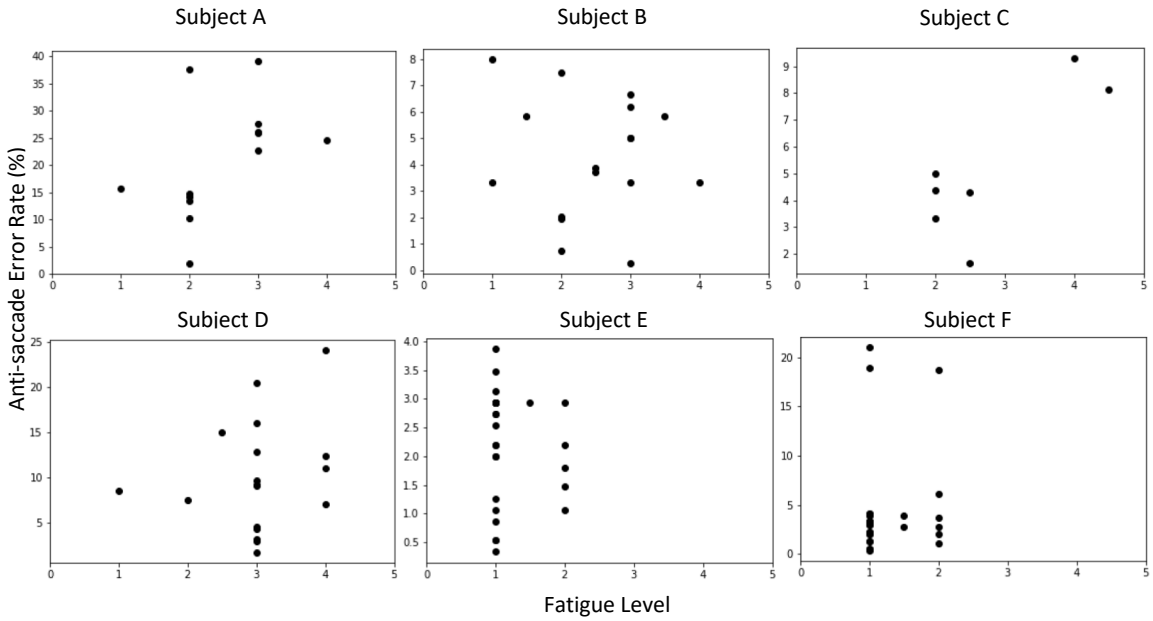

Figure 4-9: The daily anti-saccade error rate and the corresponding median self-reported fatigue level from six example subjects.

4.1.4 Correlation across Eye-Movement Features

As mentioned in the last section, the task-performing strategies may introduce the correlation across the eye-movement features. As shown in Figure 4-5, Subject 4 and 5 may have different strategies. To further understand the strategy differences among subjects, Figure 4-10 shows the correlations across the eye-movement features from five example subjects. Here, Subject 1 and 4 present a trade-off between latency and error rate. The strategies in Subject 2, 3 are slightly different from the latency and error rate trade-off. It is not clear what Subject 5's strategy is. To design an individualized longitudinal model, we need to design individualized parameters to learn the correlations across eye-movement features to account for these differences.

4.1.5 Relationship between Eye-Movement Features and Age

With the data collected in the Deployment Stage, we can also analyze the responses of eye movement features in different age groups (Figure 2-6). This is important because it gives us a baseline when we compare the results with data from patients.

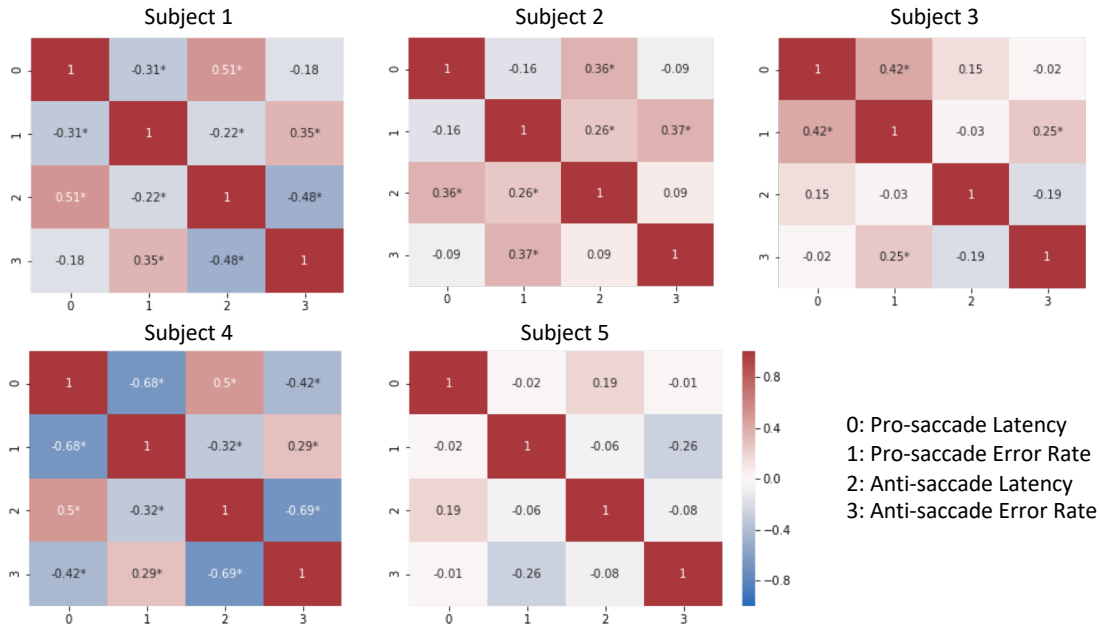

Figure 4-10: The correlation across the four eye-movement features from five example subjects. Stars mark the significance.

We calculate the mean saccade latency and error rate for each individual and then compute the mean and standard error of the individual mean saccade latencies and error rates per age group. As a result, the mean of an age group is not biased towards those subjects who provided more recordings. To evaluate the correlation between age and eye movement features, we compared our result with [65, 66], where data were collected from specialized equipment (DC electrooculography with a head rest) in a controlled environment. We notice that [65] defined an anticipatory saccade as any saccade (including errors) with latency < 90 ms. To evaluate how changing this threshold may affect the result, we show the data with and without this anticipation threshold (Figure 4-11).

Several observations are worth noting. First, since anti-saccade tasks are more complex and require more cognitive processing [16, 17, 15], the mean anti-saccade latency and anti-saccade error rate in every age group is larger than the corresponding mean pro-saccade latency and pro-saccade error rate. Moreover, we see that the saccade latency is positively correlated with age, whereas the correlation between error rate and age is not significant. These observations are in agreement with the data by

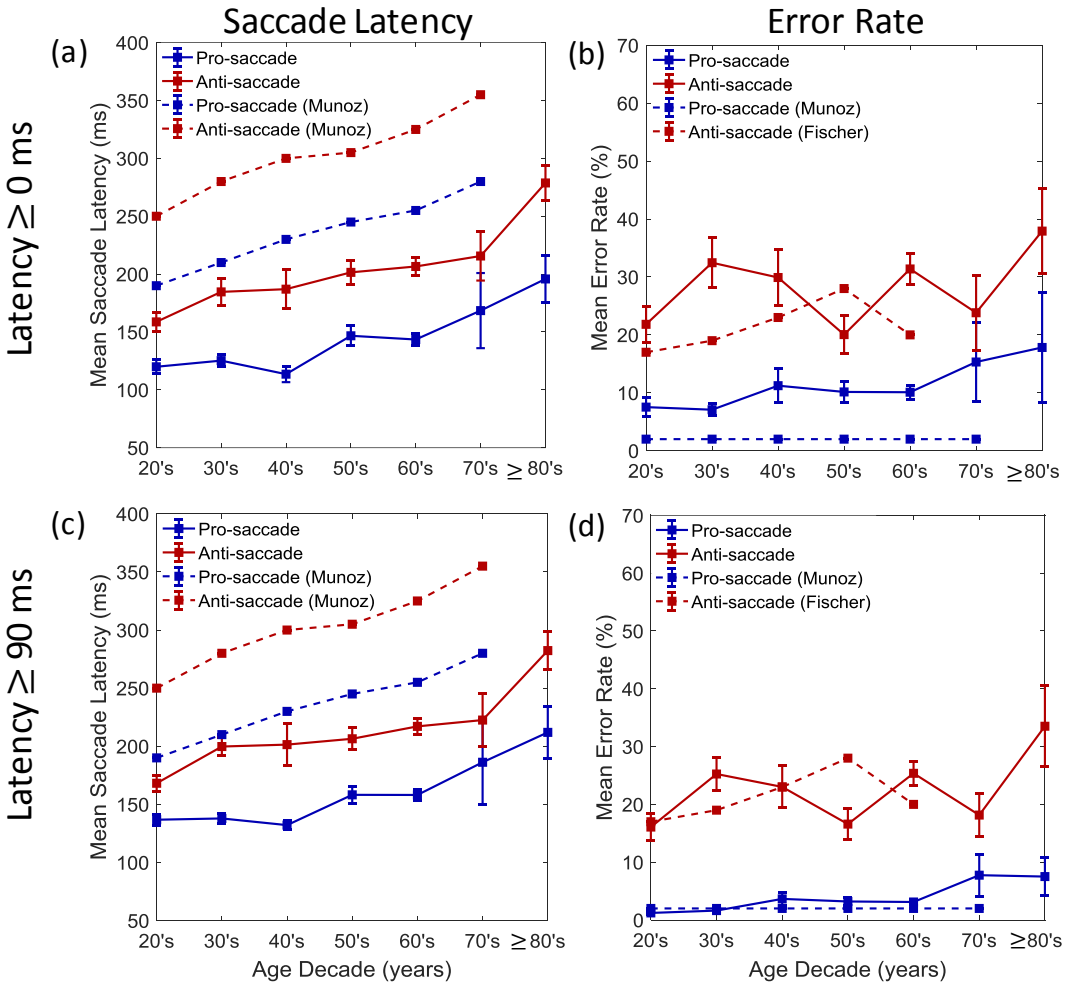

Figure 4-11: Eye movement features as a function of age with saccades > 0 ms: (a) mean saccade latency (b) mean error rate, and with saccades > 90 ms: (c) mean saccade latency (d) mean error rate. The bars showed one standard error.

Muñoz et al. [65], though the actual saccade latency values in our study tend to be lower than those reported by Muñoz et al. Our results suggest that our measurement system and processing pipeline can identify similar trends as shown in the clinical literature. Another observation is that the definition of anticipatory saccades affects the measured pro-saccade latency and error rate. On one hand, this observation is reasonable, since pro-saccade tasks are much easier to perform and errors tend to be caused by anticipation. On the other hand, while there is no consistent definition of anticipatory saccades in the literature, our observation highlights that they should be carefully defined.

We analyzed not only the mean but also the distribution of saccade latencies. To do so, we analyzed the mean pro-saccade latency of each subject in seven age groups and chose from each age group the subject with the median mean pro-saccade latency as the representative subject. In Figure 4-12, we showed example saccade latency distributions of these representative subjects. As in the Validation Stage, we still observed that there are significant intra- and inter-subject variations in saccade latency across our study cohort, which suggests that aggregated results may lose the information encoded in individual distributions.

4.2 Longitudinal Model

In this section, we design a Gaussian Process (GP) model that 1) can characterize longitudinal eye-movement features from healthy subjects and 2) is sufficiently flexible to be extended to a disease progression model. GP is popular for disease progression modeling for two reasons. First, GP is a nonparametric model and its complexity can be adapted to the complexity of the training data. Compared to a linear model which can only characterize a linear function, GP can characterize an infinite dimensional function. Thus, it is more flexible than a linear model or any model consisting of a finite number of basis functions. Secondly, because any finite samples from GP form a Gaussian distribution, its learning and inference steps are theoretically feasible. However, because the model adapts to the complexity of the

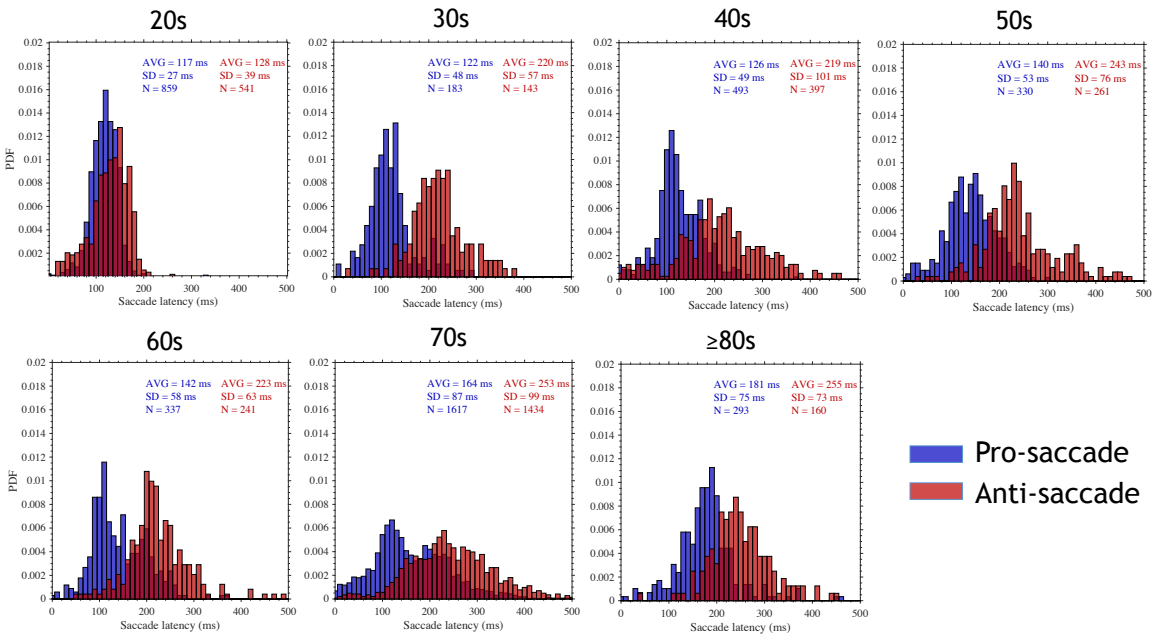

Figure 4-12: Representative normalized distributions, shown as probability density functions (PDFs), of pro-saccade (blue) and anti-saccade (red) latencies for each decade in age of the study population. Subjects whose mean pro-saccade latency is the median of the corresponding age group were chosen to represent each group. No censoring was applied to eliminate anticipatory saccades. AVG: average latency; SD: standard deviation; N: number of eye movements.

training data, the computation complexity of both learning and inference increases significantly as the size of the data grows. [67] provides a review of work on scalable GPs. While the computation complexity is not a concern with the amount of data we currently have, we show in Section 4.2.5 that stochastic variational inference for GP (SVI-GP) [68, 50] can be applied to our model to reduce the computation complexity.

We design candidate models and present the motivations behind the design in Appendix D. We then evaluate the performances of these candidate models and discuss how the final model can be extended to a disease progression model.

4.2.1 Data Preprocessing and Notations

Same as in Section 4.1.3, we group the measurements by day. We calculate the median pro/anti-saccacde latency and pro/anti-saccade error rate per day. Since the day-to-day variations vary across subjects, we normalize each subject's data by the mean and variance before fitting the data to the model. There are several implications from this preprocessing step. To begin with, with the variations normalized, the model is designed to learn the shape of the longitudinal data instead of the scale of the variability. Moreover, since after re-scaling, the variability across subjects is similar, the model we design to fit all subjects can be simpler and can be less prone to over-fitting. However, if the scale of the day-to-day variations are indications of difference disease states, we may need to extend the model.

Before introducing the candidate models, we first define the notations. We consider eye movement features $y_p = \{y_{pi}\}_{i=1}^{4}$ where $y_{pi} = \{y_{pin}\}_{n=1}^{N_p}$, p denotes the p-th subject, i denotes the i-th feature, n denotes the n-th day of measurements, and N_p denotes the number of days of measurements from the p-th subject. We denote the corresponding day of measurements as $t_p = \{t_{pn}\}_{n=1}^{N_p}$. Notice that we can still use GP to do inference if there is any missing measurements. Such condition may happen when 1) a subject decides to only take pro-saccade tasks or anti-saccade tasks in a day 2) the recordings are discarded because more than half of the saccades are LSs.

4.2.2 Model Setup

With the notations, we can present our candidate models. In addition, we provide some remarks about the strengths and the weaknesses of these models. The motivation behind these models is discussed in Appendix D.

Baseline model

$$p(y_p|g_p) = \prod_{i=1}^{4}\prod_{n=1}^{N} N(y_{pin}; \mu_{pi}, \sigma_i^2), \tag{4.1}$$

where μ_{pi} is the mean of the i-th feature from the p-th subject.

Remark: The baseline model assumes the day-to-day variations can be modeled as random noise. Therefore, the correlation across time and the correlation across the eye-movement features are assumed zero.

Multi-task model

$$p(y_p|g_p) = \prod_{i=1}^{4}\prod_{n=1}^{N} N(y_{pin}; w_{pi}g_p(t_n), \sigma_i^2), \tag{4.2}$$

where

$$g_p \sim GP(0, K^g(t, t')), \tag{4.3}$$

and $K^g(t,t') = (a^g)^2 \exp\{-\frac{|t-t'|}{l^g}\}$.

Remark: This model is a simplification of the multi-task model described in Appendix D. There are two reasons why we choose a simplified form. The first reason is interpretability. Motivated by Subject 4 in Figure 4-5, we assume that there is an underlying process $g_p(t_n)$ shared across the four eye-movement features, and that the scale w_{pi} of this underlying process on each feature is associated with each subject's task-performing strategy. For example, the signs of w_{pi} for pro-saccade latency and anti-saccade latency will be the same for Subject 4. The second reason is to avoid overfitting. With the number of data we have per subjects, learning four individualized parameters per subject is a reasonable choice.

In contrast to w_{pi} that is learned per subject per feature, the hyperparameters σ_i^2, a^g and l^g are shared across subjects. We notice that if for subject p, the effect

size of the shared process on feature i is smaller(larger) than another subject, i.e., $|w_{pi}| < |w_{pi'}|$ ($|w_{pi}| > |w_{pi'}|$), then since a^g is shared across subjects, $|w_{pi}g_p(t)|$ will also be smaller(larger). However, since σ_i is shared across subjects, it cannot be learned to be larger(smaller) to compensate for a smaller(larger) $|w_{pi}g_p(t)|$. Therefore, this model may suffer if the effect size of the shared process on the features is not uniform across subjects.

Feature-specific model

$$p(y_p|h_p) = \prod_{i=1}^{4}\prod_{n=1}^{N} N(y_{pin}; h_{pi}(t_n), \sigma_i^2), \tag{4.4}$$

where

$$h_{pi} \sim GP(0, K_i^h(t, t')), \tag{4.5}$$

and $K_i^h(t,t') = (a_i^h)^2 \exp\{-\frac{|t-t'|}{l_i^h}\}$.

Remark: This model assumes that all features are independent. This assumption contradicts with the observation in Figure 4-5. While one may still use this model to predict the values of missing eye-movement features, this model cannot learn individualized strategies. However, this model can be extended as suggested by [39, 40] to account for the correlation across the subjects.

Mixed model

Motivated by the limitations in the presented multi-task model and the feature-specific model, we designed a mixed model as follows:

$$p(y_p|g_p) = \prod_{i=1}^{4}\prod_{n=1}^{N} N(y_{pin}; w_{pi}g_p(t_n) + h_{pi}(t_n), \sigma_i^2), \tag{4.6}$$

where

$$
\begin{aligned}
g_p &\sim GP(0, K^g(t,t')), \\
h_{pi} &\sim GP(0, K^h_{pi}(t,t'), \\
K^g(t,t') &= \exp\{-\frac{|t-t'|}{l^g}\}, \\
K^h_{pi}(t,t') &= a_i^2(1-\tilde{w}_{pi}^2)\exp\{-\frac{|t-t'|}{l_i^h}\},
\end{aligned}
\tag{4.7}
$$

and

$$
w_{pi} = a_i \tilde{w}_{pi}, \tilde{w}_{pi} \in (-1, 1). \tag{4.8}
$$

Remark: As noted in the remark in the multi-task model, the multi-task model assumes that the effect size of the shared process is uniform across subjects. However, this belief contradicts with the intuition shown in Figure 4-5, where the data from Subject 4 can be explained by a shared process but not the data from Subject 5. Thus, in this model, we include a feature-specific GP. We notice that here if a subject's $|\tilde{w}_{pi}|$ is small, then the term $|w_{pi}g_p(t)|$ will be small. However, with a smaller $|\tilde{w}_{pi}|$, the covariance function of the $h_{pi}(t)$ will be larger. As a result, $|h_{pi}(t)|$ will be larger. That is, $|\tilde{w}_{pi}|$ not only controls how the four features are correlated, it also controls the effect size of the shared process.

4.2.3 Model Learning

The hyperparameters include $\{w_{pi}, \tilde{w}_{pi}, a^g, a_i^h, a_i, l^g, l_i^h, \sigma_i\}$. As suggested by [47], these hyperparameters are learned by maximizing the likelihood functions. The maximization is performed using gradient descent with momentum (learning rate= 0.001 and momentum= 0.9).

4.2.4 Model Evaluations

To evaluate the candidate models, we use two performance metrics – normalized L2 error and normalized log-likelihood. Say the testing data is (t_*, y_*) with N_* data points and the algorithm predicts the values at t_* to be distributed as $N(\mu_*, \Sigma_*)$. The normalized L2 error can be defined as $\frac{\|y_*-\mu_*\|_2}{\|y_*\|_2}$. Notice that since we remove

the mean before fitting the data, for the baseline model, we have $\mu_* = 0$. Therefore, the normalized L2 error for the baseline model is one. However, the normalized L2 error does not quantify the uncertainty estimate Σ_*. To incorporate the uncertainty estimate, we can define the normalized log-likelihood as follows:

$$\frac{1}{N_*}\log p(y_*) = -\frac{1}{2}\log(2\pi) - \frac{1}{2N_*}\log|\Sigma_*| - \frac{1}{2N_*}(y_* - \mu_*)^T\Sigma_*^{-1}(y_* - \mu_*). \tag{4.9}$$

A model performs well when the normalized L2 is small and the normalized log-likelihood is large.

We first evaluate the performance of these models over different number of days of recordings. We can imagine that since the mixed model is the most complex, it may overfit if the training data is not sufficiently large. However, the other candidate models do not provide the same flexibility in modeling the task-performing strategies as the mixed model. Therefore, as we collect more data, the mixed model may outperform the other models. After we understand how many days of recordings is sufficient to characterize a subject's eye-movement features, we next evaluate how well the candidate models characterize the correlation across features from subjects with sufficiently many data. Finally, we evaluate whether a linear trend should be included in the model, which may account for learning effects.

Number of Days of Recordings

In order to understand how many days of recordings is needed, we analyze subjects with more than 60 days of recordings. As shown in Figure 4-2, there are five subjects with more than 60 days of recordings. We remove one subject because the subject's pro-saccade latency is larger than the anti-saccade latency and we are uncertain whether the subject understands the task. To test the performance of the models with $N = 15, 25, 35, 45, 60$ days of recordings, we keep the first N days of recordings and perform 3-fold cross validation with data missing at random. For each fold, we average over the subjects and acquire one normalized L2 and one normalized log-likelihood. In Figure 4-13, we take average over the three folds and the error bars

mark the maximum and the minimum values from the three folds.

Several observations can be made. First, we notice that GP-based models outperform the baseline when there are more than 25 days of recordings regarding both normalized L2 and log-likelihood. Since the baseline model does not assume correlation across time, this observation suggests that there may be correlation across time. That is, we can characterize eye-movement features from healthy subjects better than assuming that healthy subjects have fixed eye-movement features and all the day-to-day variations are caused by random noise. In addition, with more than 25 days of recordings, a mixed model performs the best, followed in order by the feature-specific model, the multi-task model, and the baseline. Without assuming the correlation across the features, the feature-specific model can still perform similarly to a mixed model since it can predict a missing data point using its neighboring data. However, the mixed model outperforms the feature-specific model since the correlation across the features can help the prediction and reduce the uncertainty in the prediction. The multi-task model also learns the correlation across the features by assuming an underlying shared process across the four eye-movement features. However, as explained in the remark in Section 4.2.2, the multi-task model assumes that besides the shared process across all four features, all the other day-to-day variations are caused by noise. Therefore, it may not characterize eye-movement features from some subjects such as Subject 5. As a result, it generally performs worse than the mixed model and the feature-specific model.

Correlation across Features

In the last section, we notice that a GP model can use the correlation across time to predict the missing data from neighboring data. To evaluate how well a model characterizes the correlation across the features, we remove a continuous segment of a feature instead of randomly removing data as in the previous experiment. In this case, a GP model cannot use the neighboring data to predict the missing data but use the other features. More precisely, for each subject, we cut the data into three segments and remove the middle segment of each of the four features, one at a time.

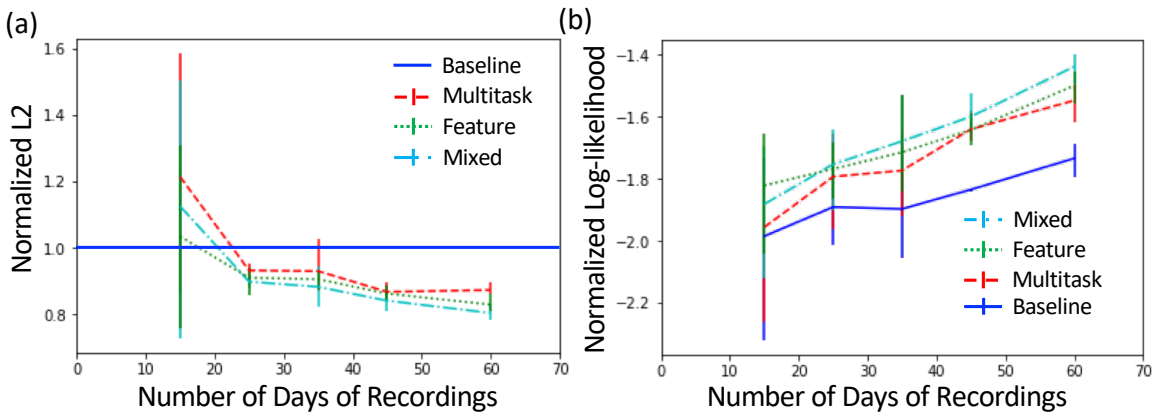

Figure 4-13: The performance of the baseline and the three GP models with different number of days of recordings regarding (a) normalized L2 and (b) normalized log-likelihood. The experiments were performed using 3-fold cross validation. The error bars show the maximum and minimum values from the three folds.

That is, there are two thirds of the recordings with four features intact and one thirds of the recordings missing one feature. We test it on subjects with more than 45 days of recordings since in there would be 30 (>25) days of recordings with the four features for the models to learn the correlation. We then average over the four features and show the performance of the models in Figure 4-14.

We notice that regarding normalized L2, the performance of the feature-specific model is comparable to the baseline in all five subjects. This is to be expected since the model assumes all the features are independent. As shown in Figure 4-10, almost all the features from Subject 1 and Subject 4 are significantly correlated. As a result, we see in Figure 4-14 that the multi-task model and the mixed model perform better than the baseline in Subject 1 and Subject 4. To observe it more closely, we show in Figure 4-15 how the missing pro-saccade values from Subject 4 are predicted by the three GP models. As shown in Figure 4-5, it is clear that the missing data can be predicted from the anti-saccade latency. We see that in Figure 4-15(b), since the feature-specific model fits each model independently, it can only assume the pro-saccade latency increases gradually from Day 25 to Day 60. However, with the assumption of a shared process, both the multi-task model and the mixed model can predict the trend of the missing data using the anti-saccade latency. In addition, since the mixed model is more flexible than the multi-task model regarding

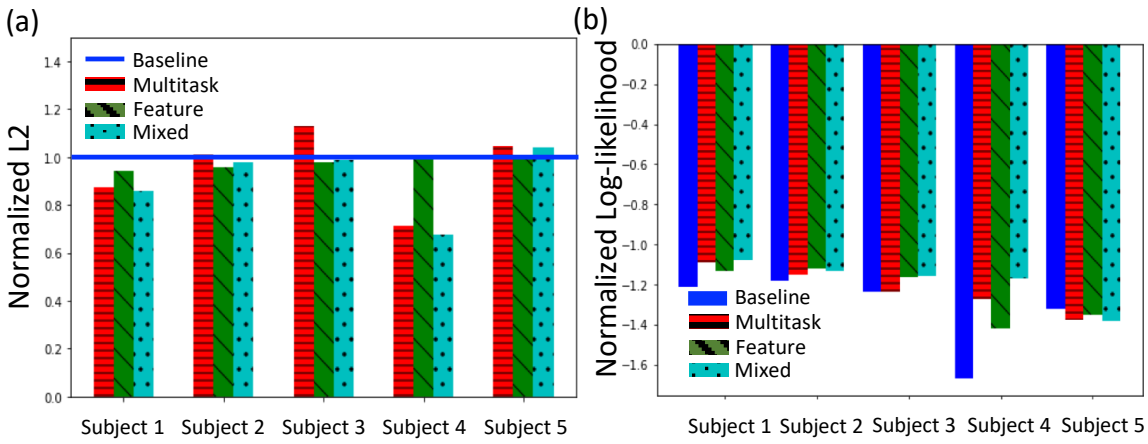

Figure 4-14: The performance of the baseline and the three GP models on subjects with more than 45 days of data regarding (a) normalized L2 and (b) normalized log-likelihood. Subjects are ordered by their number of recordings in decreasing order.

the effect size of the shared process, we see that the mixed model performs better than the multi-task model. As for normalized log-likelihood, we see that the mixed model generally performs the best. However, in Subject 5, the baseline model performs the best. If we look at the correlations across Subject 5's features in Figure 4-10, we notice that the features are not significantly correlated. As a result, a baseline model may be least prone to over-fitting and can perform the best.

We further compare the learned correlation from the mixed model with the estimated correlation from the data. As shown in Figure 4-16, the model can learn the signs of the correlation correctly if the correlation is significant. However, we also notice that the learned correlations seem to be smaller in general when compared to the estimated correlations from the data. There may be two reasons behind it. First, the mixed model only assumes a shared process across the four features. As presented in Appendix D, without considering the impact of the noise, it assumes a rank-one correlation across the features, which is a simplification. Second, the correlations are only learned from two thirds of the data. We expect the model to learn the correlation better as more data are used in the training process.

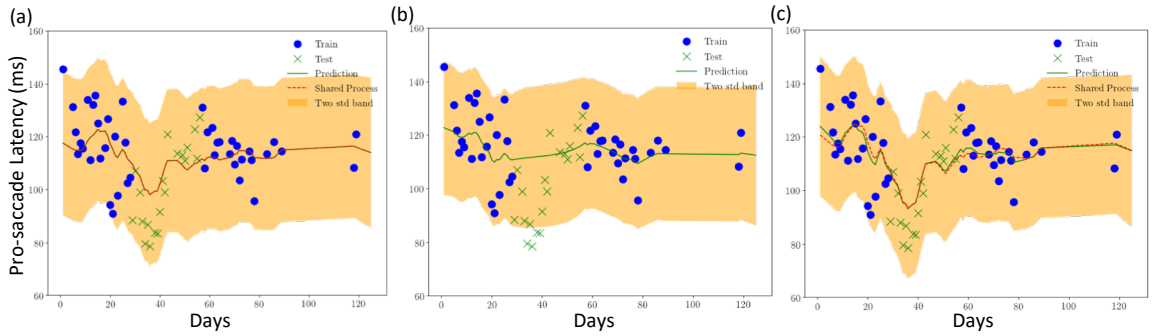

Figure 4-15: The performance of the three GP models on Subject 4 in Figure 4-5 with missing pro-saccade latency values – (a) the multi-task model, (b) the feature-specific model, and (c) the mixed model. The training data, the testing data, the predictions, the learned shared processes, and the two-standard-deviation bounds are shown. In the multi-task model, the prediction is the same as the learned shared process.

Linear Trend

In the mixed GP model, we assume the mean functions of $h_i(t)$ to be zero. In this section, we test whether the model performs better if we instead assume the mean functions to be linear. That is, whether there is a significant linear trend in the data. To do so, we modify the mixed model as follows:

$$p(y_p|g_p) = \prod_{i=1}^{4}\prod_{n=1}^{N} N(y_{pin}; w_{pi}g_p(t_n) + h_{pi}(t_n), \sigma_i^2), \tag{4.10}$$

where

$$\begin{aligned} g_p &\sim GP(0, K^g(t,t')), \\ h_{pi} &\sim GP(\Phi_{ind}^{(i)}(t)^T b_{pi}, K_{pi}^h(t,t'), \\ K^g(t,t') &= \exp\{-\frac{|t-t'|}{l^g}\}, \\ K_{pi}^h(t,t') &= a_i^2(1-\tilde{w}_{pi}^2)\exp\{-\frac{|t-t'|}{l_i^h}\}, \\ w_{pi} &= a_i\tilde{w}_{pi}, \tilde{w}_{pi} \in (-1,1), \end{aligned} \tag{4.11}$$

and

$$b_{pi} \sim N(0, B^{(i)}). \tag{4.12}$$

Here, $\Phi_{ind}(i)(t)$ are the two bases (the slope and the intersection) for the linear functions and b_{pi} are the corresponding coefficients. We assume that not all subjects may

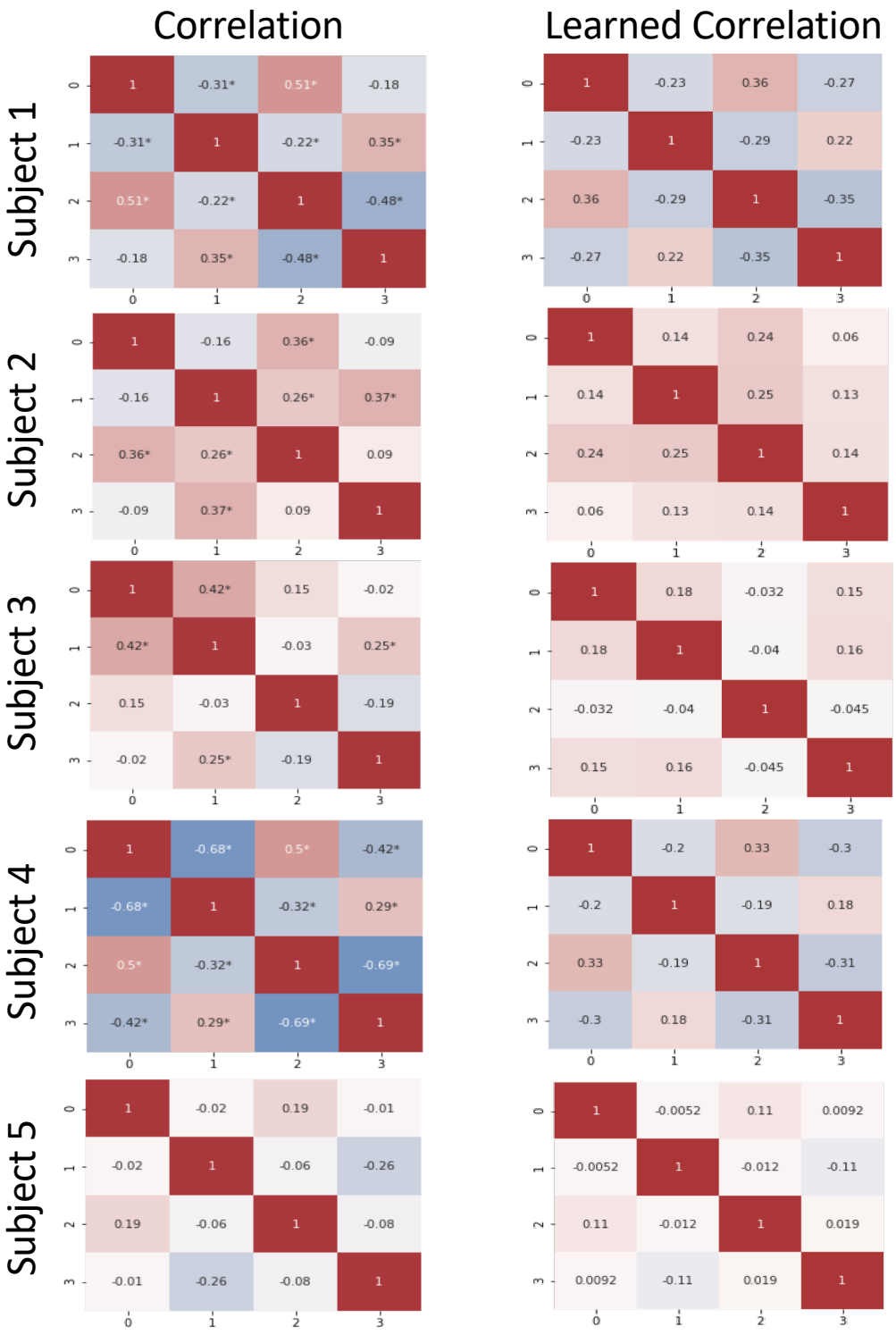

Figure 4-16: The correlation estimated from the data versus the correlation learned by the mixed model.

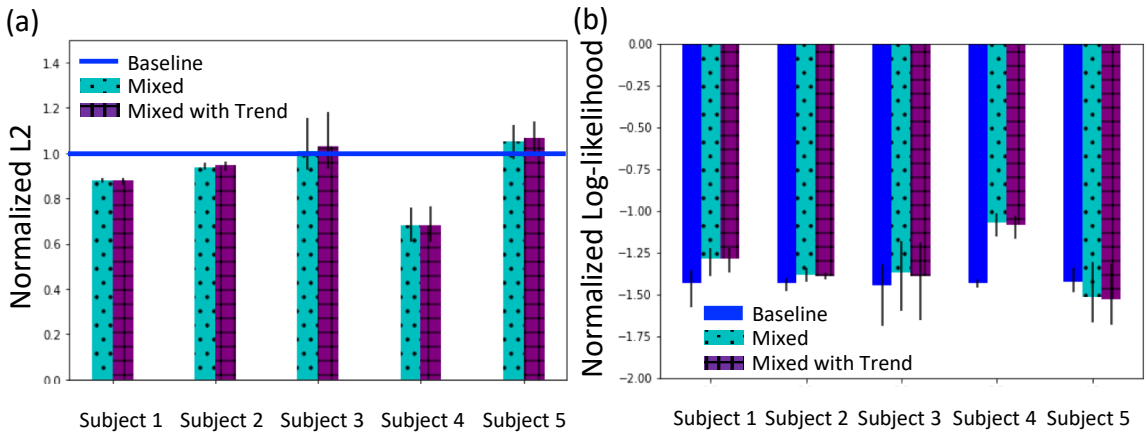

Figure 4-17: The performance of the baseline, the mixed model, and the mixed model with a linear trend on subjects with more than 45 days of data regarding (a) normalized L2 and (b) normalized log-likelihood. Subjects are ordered by their number of recordings in decreasing order.

present a linear trend. As a result, these coefficients are learned for each individual p and are drawn from a population distribution $N(0, B^{(i)})$. We simplify the setup by assuming that $B^{(i)}$ is diagonal. That is, there are two hyperparameters to learn for each i. In total, this model has eight more hyperparameters to learn than the mixed GP model.

To test whether a model with a linear trend helps, we test 3-fold cross validation on randomly missing data from subjects with more than 45 days of data. As shown in Figure 4-17, we observe that the performances with and without a linear trend are almost identical. Likely due to overfitting, the performance with a linear trend model is slightly worse the performance without a linear trend.

4.2.5 Extension

While the mixed model generally characterizes the data we have collected, it may need to be modified to account for the correlation across subjects as we collect data from more subjects. For example, we may discover that the data can be grouped into sub-population based on covariates such as gender, age, education levels, as suggested in [39, 40]. Moreover, once we have data from patients, we need to evaluate whether we

can use the data to learn their underlying disease progression. Thus, in Section 4.2.5, we propose how the multi-effect model in [39, 40] can be applied to our mixed model to characterize the effects of covariates and potentially learn the disease progression.

Another consideration is the computation complexity. The computation complexity for a multitask GP is $\mathcal{O}(P^3N^3)$, which grows significantly as N increases. Therefore, we also show how stochastic variational inference for GP [68] can be applied to our extended model. By using M induced points, the computation complexity can be reduced to $\mathcal{O}(P^3M^3)$.

Model Extension for Patients

As in the mixed model, the extended model can be written as follows:

$$p(y_p|g_p, h_p) = \prod_{i=1}^{4}\prod_{n=1}^{N} N(y_{pin}; w_i g_p(t_n) + h_{pi}(t_n), \sigma_i^2) \tag{4.13}$$

To incorporate the individual effect, the sub-population effect, and the population effect as discussed in [39, 40] and in Appendix D, we can assume h_{pi} as follows:

$$h_{pi} \sim GP(\Phi_{pop}^{(i)}(t)^T \Lambda^{(i)} x_p + \Phi_{sub}^{(i)}(t)^T \beta_{c_p,i} + \Phi_{ind}^{(i)}(t)^T b_{pi}, K_i^h(t,t')), \tag{4.14}$$

$$c_p|x_p \sim Multinomial(\pi_{c_p}), \pi_{c_p} \propto \exp w_c^T x_p, \tag{4.15}$$

where c_p is the cluster subject p belongs to. Once the cluster subject p belongs to is determined, a corresponding coefficient vector $\beta_{c_p,i}$ is assigned. Notice that Eq.(4.14) looks almost the same as Eq.(D.8), except that here the model is simplified. In [40], if two subjects belong to the same cluster, their features may still belong to different clusters. In our case, if two subjects belong to the same cluster, their sub-population effects will be the same.

Similar to the model in [40], we assume

$$b_{pi} \sim N(0, B^{(i)}). \tag{4.16}$$

Notice that here we do not let $b_p \sim N(0, B)$. Therefore, in this model, the dependency across h_{pi} is learned only through the sub-population term. One may consider generalizing this model to further impose dependency across features in the individual component. We do not consider it in this model to keep the model simpler.

As for g_p, we continue to assume that the mean function is zero. That is,

$$g_p \sim GP(0, K^g(t, t')). \tag{4.17}$$

As explained in Appendix D, the dependency across the features imposed by g_p is designed to model the task-performing strategies whereas the dependency imposed by the sub-population effect is designed to model how the disease progression affects all four features.

The derivation of how inference can be performed on this model is shown in Appendix E. We can imagine that the mean functions of the GP models can be used to characterize the underlying disease progression. However, since we do not have data from patients, we leave the evaluation of this model for future work.

Stochastic Variational Inference

Several work has been proposed to reduce computation complexity for GP. We show in Appendix F that we can apply stochastic variational inference for GP [68] to our extended model.

4.3 Discussion and Summary

Our ultimate goal of this work is to evaluate whether eye-movement features can be used to track the progression of neurodegenerative diseases. Unfortunately, there are few studies that track the longitudinal changes in saccade latency among patients [19,

20], especially within the same cohort. Because the data in these studies were collected in clinical environments and the analyses usually involved manual removal of outliers, longitudinal measurements are sparse (typically with an interval greater than six months). Therefore, current methods cannot assess disease progression sufficiently frequently to detect disease onset or efficiently evaluate treatment effects.

With the system and the methods developed in Chapter 2 and Chapter 3, we are able to collect significantly more saccades and more sessions per subject than previously possible – 6,787 recordings and 235,520 eye movements from 80 subjects, 45 of whom with multiple recording sessions. These sizable data allow us to study the intra- and inter-subject variability of individual saccade distributions, the day-to-day variations in the eye-movement features, the correlation across the eye-movement features, and the correlation between the features and age. By understanding the characteristics of the eye-movement features from healthy subjects, we can put into better context the changes seen in patients with neurodegenerative disease and potentially use these features to track the disease progression.

4.3.1 Intra- and inter-subject variability in distributions

The rich information regarding the distinctive shape and parameters of the individual distributions is lost when saccade latency values are pooled, which is the case in most literature. As seen in Figure 4-3, some individuals have a tendency to make more saccades with shorter latencies and others to make more saccades with longer latencies. In combining all the data into a single distribution, these individual characteristics – that have been linked to specific brain pathologies [69, 70] – are lost. Saccade latency intra-subject variability is also lost when data is pooled. If instead the information regarding this variability were preserved, it could be used as a feature to assess the cognitive state of a subject. For example, some studies suggest that intra-subject variability is larger in some conditions compared to normal subjects [71, 72]. Our accessible, low-cost measurement system enables widespread data collection and hence avoids having to combine data from different subjects, allowing us to preserve the distinctive information in each individual saccade latency distribution (Figure 4-3).

In addition to the large intra- and inter-subject variability, we observed that the saccade latency distribution of the majority of the subjects may be modeled as a log-normal distribution. This observation is consistent with [73], in which neural mechanisms are discussed that might give rise to log-normally distributed reaction times. It might therefore be sufficient to characterize individual saccade-latency distributions using the two parameters of a log-normal distribution (log −mean and log −variance) and analyze how these parameters change through time.

In this work, when we build the individualized longitudinal model, we only analyze the daily median saccade latency over time. One can imagine that the model can be extended to characterize the changes in the daily distributions by including higher moments (e.g., variance) or by tracking the two parameters of the log-normal models. Since the variance of the estimate of a higher moment may be larger, one may need to consider grouping several days of data to reduce the variance. We leave this analysis for future work.

4.3.2 Day-to-day variations

Our longitudinal data collection also enables us to study the day-to-day variations in the eye-movement features. Figure 4-4 shows that not only there is significant inter-subject variability in the saccade latency distributions, there is also significant inter-subject variability in the day-to-day variations. We further analyze the variations within a day using bootstrapping and show that the variations within a day is smaller than the variations across the days. This observation suggests that the source of the day-to-day variations cannot be solely explained by random measurement noise.

To examine the fatigue effects on the day-to-day variations, we inspect the correlation between the day-to-day variations and the self-reported tiredness levels. However, we cannot conclude the fatigue effects due to potential confounding factors such as a subject's concentration level and a subject's tendency to not choose the extreme scores. We may need to improve the tiredness question we ask at the end of each recording to acquire more meaningful data. We may also consider more objective measurements such as eyelid droops and the number of blinks.

4.3.3 Correlation across eye-movement features

Another source of day-to-day variations is the change of a subject's task-performing strategy. A subject may be testing different strategies throughout the course of the recordings. As shown in Figure 4-5, Subject 4 seems to tradeoff between speed and accuracy. Therefore, when the latency values decrease, the error rates rise, and vice versa. This task-performing strategy introduces the correlation across eye-movement features. As shown in Figure 4-10, we see that Subject 1 and Subject 4 present significant correlations across the eye-movement features whereas the correlations across the eye-movement features in Subject 5 are insignificant. This observation suggests that not all subjects have similar strategies. Therefore, when we design an individualized longitudinal model, we need to model individualized correlations.

4.3.4 Age and eye-movement features

Since we also collect data from subjects across the adult spectrum, we can study the correlation between the eye-movement features and age and compare the result with the literature. As in the literature, we observe that anti-saccade latency and error rate tend to be larger than pro-saccade latency and error rate, respectively. Across the age range, we also observe that saccade latency is positively correlated with age while a strong relationship between error rate and age is not apparent. This observation also matches the observation in prior work [65, 66]. Although our saccade latency values are smaller than values reported in [65, 66], our values are within the range of latency values reported in the clinical literature [52, 16, 31, 74]. Several hypotheses can be made to explain why our values may be smaller. First, our recording setup is less constrained. As mentioned in [9], recording subjects in dedicated environments may affect a subject's cognitive awareness. Second, our subjects are mostly graduate students or professors. It is likely that education level may affect reaction time. We also have fewer subjects in the 70's and 80's than in other age brackets. While one of the three subjects in the 70's has latency values much closer to the values reported in the literature, two other subjects have smaller latency values.

We also observe that the definition of an anticipatory saccade may significantly affect the measured pro-saccade latency and error rate. While the definition is not consistent across the clinical literature, our observation suggests that a more careful investigation into the effect of picking a latency threshold for anticipatory saccages on mean saccade latency is warranted. Some investigations designed tasks to avoid anticipatory saccades [75, 76], for example, by randomizing the length of the fixation period or by including more positions where a stimulus can be presented. However, we suspect that these modifications may result in an increased error rate. Since we aim to design and validate our error detection algorithm in this work, we did not implement either of these modifications. Nevertheless, it is worth analyzing how these modifications may affect saccade latency and error rate.

4.3.5 Longitudinal Model

With a better understanding of how eye-movement features change over time in healthy subjects, we can design individualized longitudinal models that can characterize the features in the hope that the models can be extended for monitoring disease progression. GP models have been commonly used in disease progression modeling [39, 40, 41]. In particular, we evaluated the performances of three GP models. While all these models are special cases of a multi-task GP model, the mixed model particularly was designed based on the intuition we learned about individual task-performing strategies. The mixed model can model the effect size of the strategy flexibly. We compare the three GP models with a baseline model where we assume that the day-to-day variations are caused by random noise. We notice that when we have collected more than 25 days of recordings, all three models out-perform the baseline. It suggests that the eye-movement features are correlated over time and that we can characterize the eye-movement features better than assuming that they are fixed over time in healthy subjects. In addition, we evaluate the abilities of the three GP models in characterizing the correlation across the eye-movement features. We see that the mixed model performs the best when the correlations across the eye-movement features are significant. We further inspect the correlations learned by the

mixed model. We notice that the signs of the correlations can be learned correctly when they are significant, which means that the mixed model may learn individual task-performing strategies.

Last but not least, we test whether the performance can be improved by adding a linear trend in the mixed model. We notice that the performance hardly changes after we assume a linear trend. We hypothesize that it is because 1) the learning effect only lasts for a short period of time and may not be noticeable after 25 days of recordings 2) the eye-movement features were not affected by disease progression.

Given the number of recordings we have collected, we imagine that while the mixed model can be a good candidate model, it can be improved to better characterize more subjects and data collected over a much longer period. We suggest two different extensions of our current model. The first extension is motivated by [39, 40] to account for the correlation across subjects. The second extension is motivated by [68, 50] to reduce the computation complexity. We imagine that as more types of eye-movement features are collected, one may also consider a model extension that assumes sparsity in the correlation across eye-movement features as in [41].

In summary, in this chapter, we have studied longitudinal characteristics of pro/anti-saccade latency/error rate from healthy subjects, which was barely studied in the literature due to the constrained environment setup. We then use the studied characteristics to design a GP model that can track saccade latency and error rate from healthy subjects with more than 25 days of recordings, learn the correlation across the features, and be extended for disease progression modeling. Thus, we can conclude that we have enabled individualized tracking of saccade latency and error rate from healthy subjects, which can help put into context how disease progression may affect these eye movement features.

Chapter 5

Conclusion and Next Steps

5.1 Conclusion

In this thesis, we developed, validated, and deployed an app to allow for self-recording of pro/anti-saccade tasks. We then present a robust and automated pipeline to measure saccade latency and error rate from these mobile-device recordings. The pipeline includes a) an eye-tracking algorithm –iTracker-face that is robust to various recording conditions, b) a tanh model for saccade latency measurement that allows for automated outlier rejection, and c) an error-rate measurement algorithm that can automatically detect low-signal recordings that should not be further analyzed and can identify directionally erroneous eye movements.

With this platform in place, we collected over 235,000 eye movements from 80 self-reported healthy volunteers ranging in age from 20 to 92 years, two orders of magnitude more measurements than in most previous work. These data enabled us to study the day-to-day variations in saccade latency and error rate from healthy subjects. We observed significant intra- and inter-subject variability in these day-to-day variations, which highlights the importance of individualized tracking of eye-movement features. We then showed that we can track the eye-movement features from healthy subjects with more than 25 days of recordings using an individualized GP model. Such a model can help put into context how neurocognitive impairment may affect eye-movement features. In summary, by enabling app-based saccade latency

measurements and error rate determination, our work paves the way to use these digital biomarkers to aid in the quantification of neurocognitive decline and possibly from the comfort of a subject's home.

5.2 Future Work

5.2.1 System

We have shown that our app-based eye-movement measurement is user-friendly for healthy subjects. However, it may need to be improved once we have interacted with patients. First, we need to re-adjust the app and task design to ensure that the measurement is user-friendly for patients. Additionally, we may need to keep the app engaging. As mentioned in [26], the disease stage affects a subject's willingness to participate in a study. We may need to think about how to motivate patients to take recordings on a regular basis without making it burdensome. To achieve these goals, we should interact with patients and iterate the app design.

Moreover, our app is not restricted to monitor the progression of neurodegenerative diseases. Since saccade latency and error rate are assessments of a person's neurocognitive states, one may also consider using our app to test the impact of alcohol and anesthesia on cognitive ability.

In the long run, one may consider replacing our saccade task with a standard reading task, since the latter can be incorporated into a subject's usual reading routine rather than actively requiring a subject to perform a task [77]. However, the implementation of such an extension is extremely challenging because the precision of the gaze estimation required for such a task is much finer than our saccade task. As a result, it is not yet possible with the current state-of-the-art eye-tracking algorithm [78].

5.2.2 Methods

We have enabled app-based measurements of four eye-movement features – pro/anti-saccade latency/error rate. Eye-movement features such as gaze amplitude and velocity [3] may also be affected by disease progression. Thus, we may consider expanding on the types of eye-movement features we can measure to achieve a better understanding of the progression of neurodegenerative diseases. Moreover, as discussed in [26, 45], different modalities of symptoms may be related to different disease states. It is likely that no single modality can perfectly model the progression of a disease. Therefore, besides expanding on the types of eye-movement features we can measure, we should also expand on the modalities of the features, for example, by including gait and speech.

5.2.3 Data Analysis

In Section 4.2.5, we mentioned two potential extensions of our models. However, these extensions do not take into account the following considerations. First, currently we normalize the data by their day-to-day variations before fitting a model to them. If their day-to-day variations are affected by disease progression, we may need to modify the model accordingly. Second, as shown in Section 4.1.2, subjects have various latency distributions. Since we only consider the values of daily median latency, we do not consider how the distributions may also be affected by the disease progression. We may consider including more moments in the eye-movement features or consider the log-normal distribution model fit. Third, the model assumes stationarity. If it does not hold, we may need to modify the model accordingly.

Monitor of the disease progression becomes very challenging with the impact of medication. Medication may affect the eye-movement features without affecting the disease states. It requires careful analysis on how to separate short-term effects of the medication from long-term effects of the medication. Xu et al. provided a framework to estimate individual treatment response [79]. Whether this model can be applied to ours is worth studying.

Appendix A

App Synchronization

In this appendix, we detail how we bound the error associated with saccade latency determination using the app. The accuracy of the saccade latency determination depends on the following four timings:

- the real timing of the i-th stimulus presentation time s_i,
- the estimated timing of the i-th stimulus presentation time $\hat{s}_i$ from the screen timestamps,
- the real timing of the i-th saccade onset o_i,
- the estimated timing of the i-th saccade onset $\hat{o}_i$.

The i-th saccade latency is $o_i - s_i$ whereas the estimated i-th saccade latency is $\hat{o}_i - \hat{s}_i$. Therefore, the accuracy of the i-th saccade latency estimation is $(o_i - s_i) - (\hat{o}_i - \hat{s}_i) = (o_i - \hat{o}_i) - (s_i - \hat{s}_i)$. We can define $D_i^s \coloneqq s_i - \hat{s}_i$ as the error in the screen timestamps for the i-th stimulus. Similarly, we have $o_i - \hat{o}_i = D_i^r + D_i^t$ where D_i^r is the error in the recording timestamps for the i-th saccade and D_i^t is the error introduced by the tanh fitting algorithm. The accuracy of the saccade latency estimation then becomes $D_i^r - D_i^s + D_i^t$. While D_i^t was evaluated in [80] to be close to zero, $D_i \coloneqq D_i^r - D_i^s$ is affected by queued access to the processor clock and is related to our app design.

To estimate D_i, we designed the following experiment. We placed the device in front of a mirror and ran a 40-saccade task. With the mirror, we can identify

the recording frame in which each of the 40 stimuli appears first. In Fig. A-1, for example, the first stimulus was presented in Frame 85. With the 40 frame indices and the associated recording timestamps, we can translate these indices into time instants $\hat{r}_i$ (ms), $i = 1, \ldots, 40$. In Fig. A-1, $\hat{r}_1 \approx 4398.8322$ s. Similarly, from the screen timestamps, we can obtain the time $\hat{s}_i$ when the i-th stimulus is shown on the screen. Figure A-2, shows $\hat{s}_1$ to be approximately 4398.8324 s.

Since the stimulus appearing on the screen would be captured by the next camera frame and the time difference between two frames is $\frac{1000}{60}$ ms in a 60-fps recording, with the errors D_i^r and D_i^s in the timestamps, we have $\hat{r}_i + D_i^r - \frac{1000}{60} < \hat{s}_i + D_i^s \leq \hat{r}_i + D_i^r$. That is, $\hat{r}_i - \frac{1000}{60} < \hat{s}_i - D_i \leq \hat{r}_i$ where D_i is exactly the error in the saccade latency estimation introduced by the synchronization error between the screen and the recording.

From the recording timestamps, we can only find one time instant $\tilde{r}_i(D)$ as a function of D that satisfies $\tilde{r}_i(D) - \frac{1000}{60} < \hat{s}_i + D \leq \tilde{r}_i(D)$. In other words, if each recording timestamp is denoted as t_j where j denotes the frame index as in Fig. A-1, then $\tilde{r}_i(D) := \min\{t_j | t_j \geq s_i + D\}$. If our estimated synchronization error $\hat{D} = D_i$, we will have $\tilde{r}_i(\hat{D}) = r_i$. As a result, we can then define

$$\hat{D} = \arg\min_D \sum_i |\tilde{r}_i(D) - r_i|. \tag{A.1}$$

With careful app design, we can ensure $\sum_i |\tilde{r}_i(\hat{D}) - r_i| = 0$. That is, we achieve $\hat{D} = D_i$ where the estimation of the synchronization error is correct and is constant throughout each recording.

We observed that an iOS camera changes its shutter duration and ISO based on the lighting condition, which may affect the accuracy of the recording timestamps. We showed in Fig. A-3 that the shutter duration does not affect the synchronization error while ISO is positively correlated with the absolute value of the synchronization error. As a result, we set the shutter duration to 16 ms, which is close to the maximum duration 1000/60 ms in a 60-fps recording, to allow for adequate light. To bound the absoulte synchronization error to be within 5 ms, we restrict the ISO values to

be less than 1000 by asking the subject to move to a brighter environment if the automatically determined ISO exceeds 1000.

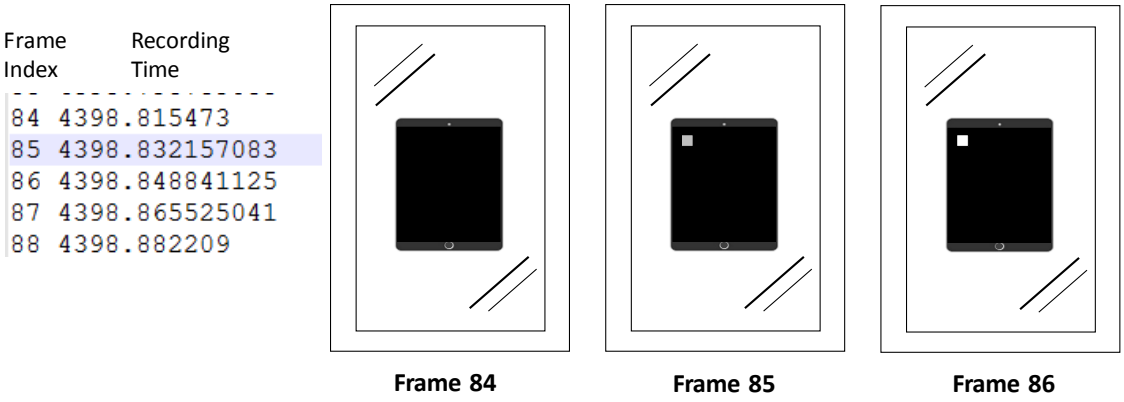

Figure A-1: Example for determining $\hat{r}_i$, the time when the i-th stimulus appears. In this example, the first stimulus appears in recording frame 85 at $\hat{r}_1 = 4398.8322$ s.

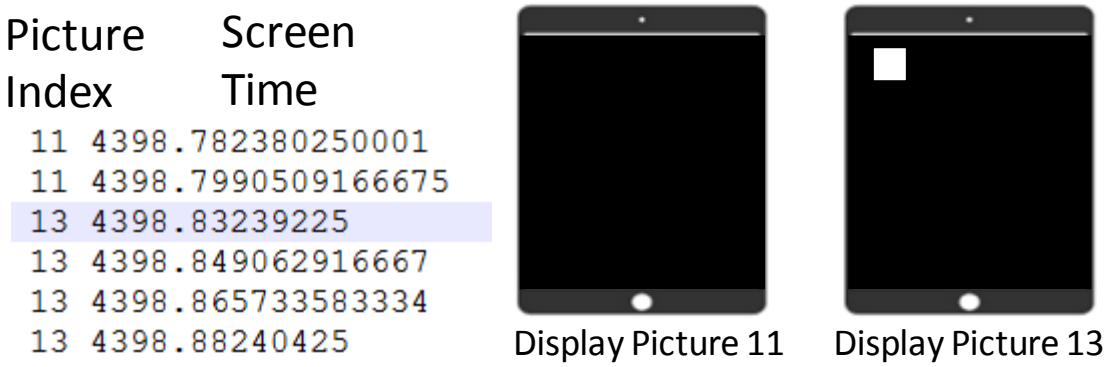

Figure A-2: Example for acquiring s_i, the time when the i-th stimulus presents on the screen. Picture 11 is a black image, and Picture 13 is the image with a left stimulus. The first stimulus shows up when Picture 13 is displayed. As a result, in this example, $\hat{s}_1 = 4398.8324$ s.

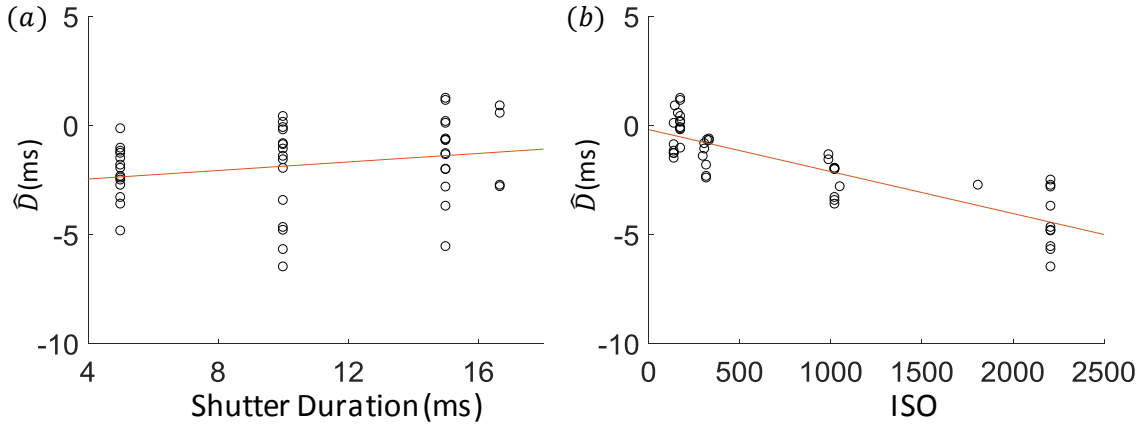

Figure A-3: The estimated synchronization error as a function of (a) shutter duration and (b) ISO. Each dot denotes one recording.

Appendix B

Face Crop Automation and Chinrest Removal

B.1 Face-crop Automation

To fully automate the signal-processing pipeline of Fig. 3-1, we replaced the manual face annotation and cropping (Fig. 3-5) with an automated face-detection step. We tested this automation by recording subjects with a chinrest. With the head supported by the chinrest, we can expect the position of the face to remain relatively stable throughout a sequence of saccade tasks and the manually determined face region to remain valid throughout the subsequent frames of a video recording. To automate the face-region determination, we used the Viola-Jones face detector [5] and evaluated the changes in the estimated saccade latencies after this automation on 158 sessions of recordings. The mean absolute differences in the mean per-session saccade latencies with an NRMSE<0.1 was 1.10 ms with an associated standard deviation of 1.24 ms (Fig. B-1). We therefore conclude that automating the face-detection step does not materially affect the saccade-latency determination in normal subjects. This result may be understood by considering that the convolutional layers in iTracker are trained to properly adjust gaze estimation under translation and scaling differences in the cropped face. As a result, the shape of the resulting eye-movement traces are hardly changed given slight differences in the cropped regions of the face.

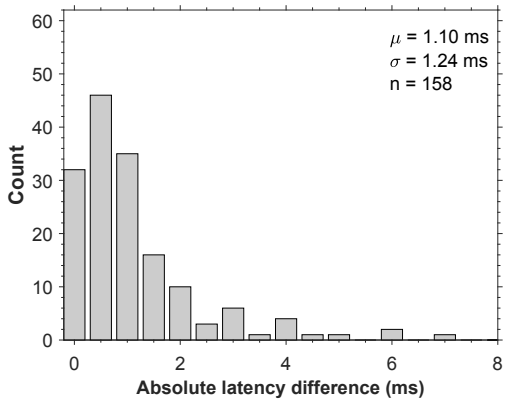

Figure B-1: The absolute difference in mean saccade latencies between face crop based on manual face annotation and automated face detection using the Viola-Jones algorithm [5].

B.2 Chinrest Removal

Ideally, we would like to enable eye-movement capture and analysis without the need for restraining the head. Without the chinrest in place, the assumption of limited head movement throughout the (approximately) two-minute 40-saccade sequence is bound to be violated. However, the assumption might still be reasonable over the course of a single saccadic eye movement, of which we typically analyze 600 ms (from 100 ms before till 500 ms after stimulus presentation). To test this hypothesis, we conducted two sessions of video recordings in four subjects each with and without the participants' heads resting on the chinrest (16 sessions in total). We applied the Viola-Jones face detector to the first frame of each individual saccade tracing and used the detected face region from the first frame and applied it to every subsequent frame. If there had been any significant head movements within a single saccade trial, we would have expected the tanh model to no longer attain low NRMSE fits. When the Viola-Jones face detector was applied to iTracker-face derived eye-movement traces on recordings obtained with and without chinrest, most of the traces have comparable signal-to-noise (Fig. B-2). After confirming that the null hypothesis of normally distributed mean saccade latency cannot be rejected at the 0.05 level (using the Anderson-Darling test), we performed a formal analysis of variance (ANOVA) to assess whether a significant difference existed between mean saccade latencies measured

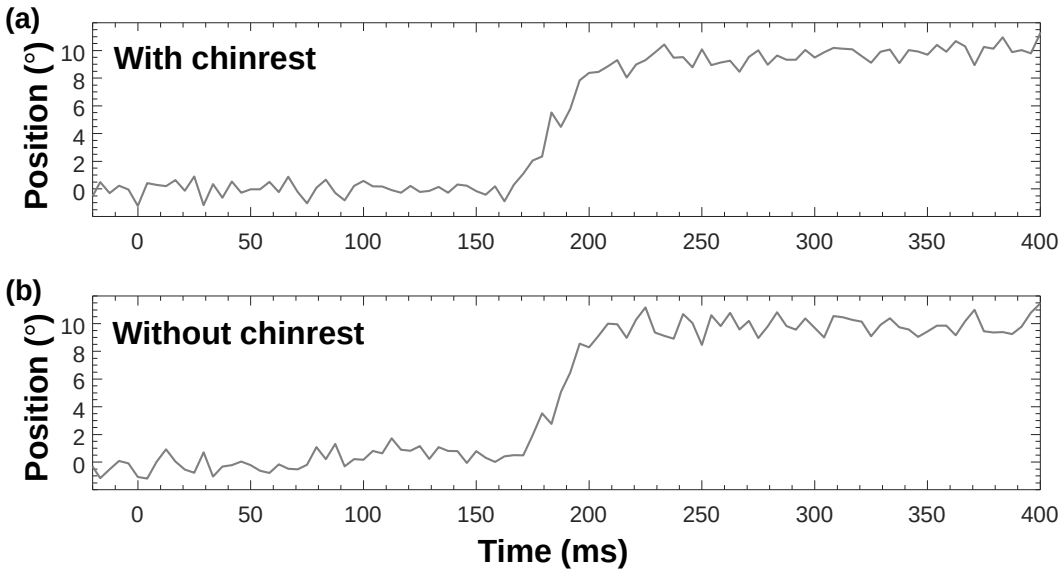

Figure B-2: Two examples of saccadic eye-movement traces in the same subject. (a) Recording with chinrest, and (b) recording without chinrest. They have a comparable signal-to-noise level.

with and without chinrest. The ANOVA null hypothesis of a significant difference was rejected ($p = 0.59$). We therefore conclude that in our cohort of self-reported healthy volunteers, the chinrest is not essential to obtaining recordings of sufficient quality for saccade-onset detection and saccade-latency determination.

The selection of iTracker-face to generate the eye-movement tracings, the NRMSE threshold value of 0.1 to select traces for inclusion in our analysis, and the Viola-Jones algorithm for automated face detection on the first frame of each saccade task video sequence completes the automation of the saccade-latency determination pipeline of Fig. 3-1. In the next section, we apply this pipeline to determine the intra- and inter-subject variability in saccade-latency measurements obtained from video sequences of self-reported healthy subjects, and explore the statistical modeling of the saccade-latency distributions.

Appendix C

Lognormal Distribution Fitting

To robustly fit a log-normal model to the session-by-session or aggregate (by subject) saccade-latency distributions, we log-transformed the saccade latency values and fit a probit model [81] to estimate the parameters of the log-normal model.

However, since we assume that a saccade with latency value smaller than 80 ms may be anticipatory, log-latency values below log(80) may not follow the same normal distribution as values above. As a result, we cannot estimate the mean and standard deviation of the log-normal model by calculating the mean and standard deviation of the entire log-latency data.

To address this difficulty, we consider a probit model [81], which fits normally distributed data into a linear model.

It calculates the cumulative density function $F_X(x) = P(X \leq x)$ of the data X, and find a transformation function $g(z)$ so that $g(F_X(x))$ becomes linear. With our assumption, we then can assume that for $x \geq \log(80)$, $g(F(x))$ is linear. Since one can use the slope and the x-axis intersection of the linear model to find the mean and standard deviation of the normally distributed data, we then can apply the linear model only on log saccade latencies greater than log(80) to find the log mean and log standard deviation. With a probit model, we are able to fit a distribution on the desired portion of the data (without fitting the anticipatory data). We can express the procedures as follows:

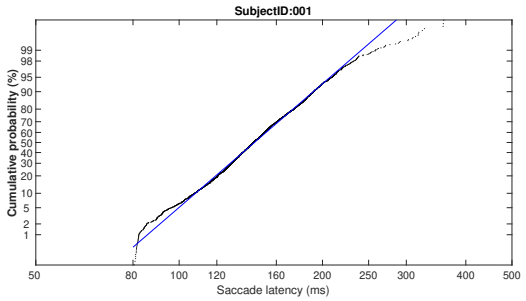

Figure C-1: The probit plot for the log-latency values of subject 001. The blue line shows the linear fit.

If $X \sim \mathcal{N}(\mu, \sigma^2)$, then one can show that

$$\sqrt{2}erf^{-1}(2(F_X(x) - 1)) = \frac{x - \mu}{\sigma}, \tag{C.1}$$

where $erf^{-1}(\cdot)$ denotes the inverse of the error function, which is defined as

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt, \tag{C.2}$$

and $F_X(x)$ denotes the cumulated density function of X.

If our y-axis is $y = g(x) = \sqrt{2}erf^{-1}(2(F_X(x) - 1))$, then the relationship between x and y should be linear (this is the probit model). The mean of the data X will be the intersection of the line with the x-axis, and the standard deviation will be the reciprocal of the slope.

A typical probit plot for a normal subject's log-saccade latency values is shown in Fig. C-1. Instead of fitting a linear line on data with log latency values above log(80), we fit a linear model on data points lying between 0.05 quantile and 0.95 quantile so that the estimated parameters will be even more robust to outliers. If the fitting line has a form of $y = ax + b$, then we estimate the mean value to be $-b/a$ and the standard deviation to be $1/a$. After the mean and the standard deviation of the log-normal model are estimated, we define the model for each distribution as the estimated log-normal model truncated at 80 ms.

Appendix D

Gaussian Process Models

Here we give a concise review of Gaussian Process (GP) models that motivate our model development. An in-depth overview can be found in [47]. In this chapter, we first introduce a one-dimensional GP model (where the output is one-dimensional) and show its learning and inference steps. We then discuss how such a model can be extended to a multi-dimensional GP. In particular, we study multi-task GPs and multi-level GPs.

D.1 One Dimensional Gaussian Process

A Gaussian process can be defined as follows: it is a random process where any finite samples are Gaussian distributed. In our work, the input and output to our model are time and eye-movement features respectively. As a result, our input is one-dimensional. In general, the input to a GP model can be multi-dimensional. However, to be consistent with the notations we use in Section 4.2.1, we keep the input as time t. We refer curious readers to [47] for the extension.

A GP model can be fully characterized by its mean function and covariance function. Let

$$f(t) \sim GP(m(t), k(t, t')), \tag{D.1}$$

where $m(t)$ is the mean function and $k(t, t')$ is the covariance function. If $m(t)$ is

not zero, we can simply re-define $f(t)$ as $f(t) - m(t)$. Therefore, without loss of generality, we can assume $m(t) = 0$. It is usually assumed that the measurements can be modeled as a GP with some noise, i.e.,

$$y(t) \sim (f(t), \sigma^2), \tag{D.2}$$

where the variance of the noise is σ^2. Assume that the training data is $(t, y) = \{(t_n, y_n)\}_{n=1}^{N}$ where N is the number of samples, to predict the value f_* at t_*, we can formulate the following:

$$\begin{bmatrix} y \\ f_* \end{bmatrix}) \sim N(0, \begin{bmatrix} k(t,t) + \sigma^2 I_n & k(t, t_*) \\ k(t_*, t) & k(t_*, t_*) \end{bmatrix}),$$

where I_n is an identity matrix of size n by n. We can then predict f_* using conditional distribution to attain

$$f_* \sim N(K_*^T [K + \sigma^2 I_n]^{-1} y, K_{**} - K_*^T [K + \sigma^2 I_n]^{-1} K_*), \tag{D.3}$$

where $K_* = k(t, t_*)$, $K = k(t, t)$, and $K_{**} = k(t_*, t_*)$.

Now we know how inference is performed if we know $k(t, t')$. The question is reduced to how to define $k(t, t')$ and how to learn it. In [47], several covariance functions are introduced. For our purpose, we use an Ornstein-Uhlenbeck (OU) prcess which is described as follows:

$$K(t, t') = a^2 \exp\{-l^{-1}|t - t'|\} \tag{D.4}$$

Here, a and l are hyperparameters to be learned from the data. This covariance function is widely used in real-world data (probably more explanation). (Probably some figures to show what is the meaning behind these hyperparameters.) The hyperparameters σ, a, l are commonly trained using maximum log-likelihood. That is, besides inference, we also know how learning is performed.

D.2 Multi-dimensional Gaussian Process

To extend a one-dimensional GP to a multi-dimensional GP, we need to model the correlation across the outputs. In particular, we first present a multi-task GP [48], which is used to model the correlation across the eye-movement features in our work. Then, we present a multi-level GP designed by [39, 40]. The idea from [39, 40] can help us extend our model to incorporate the correlation across subjects.

D.2.1 Multi-task Gaussian Process

The idea of a multi-task GP is known as linear models of coregionalization (LMC) in the geostatistics literature [49]. In this subsection, we gradually motivate a multi-task GP in a manner similar to the presentation in [82].

We first consider modeling the output features as a linear combination of latent processes. That is, we consider $f_i(t) = \Sigma_{q=1}^{Q} w_{iq} g_q(t)$ where $g_q(t) \sim GP(0, K_{time,q}(t, t'))$. Notice that the analogy to this model in linear algebra is to find the linearly independent vectors that span a vector space. The covariance function $K_{time,q}(t, t'))$ describes the correlation across time. With such a model, we can have the following observation:

$$K(t,t) = \text{Var}(\begin{bmatrix} f_1(t_1) \\ \vdots \\ f_1(t_N) \\ \vdots \\ f_P(t_N) \end{bmatrix}) = \Sigma_{q=1}^{Q} \begin{bmatrix} w_{1q}^2 & \dots & w_{1q}w_{Pq} \\ w_{2q}w_{1q} & \dots & w_{2j}w_{Pq} \\ \vdots & \ddots & \vdots \\ w_{Pq}w_{1q} & \dots & w_{Pq}^2 \end{bmatrix} \otimes K_{time,q}(t,t).$$

If we write

$$K(t,t) = \Sigma_{q=1}^{Q} K_{feature,q} \otimes K_{time,q}(t,t), \tag{D.5}$$

then we have:

$$K_{feature,q} = \begin{bmatrix} w_{1q}^2 & \dots & w_{1q}w_{Pq} \\ w_{2q}w_{1q} & \dots & w_{2j}w_{Pq} \\ \vdots & \ddots & \vdots \\ w_{Pq}w_{1q} & \dots & w_{Pq}^2 \end{bmatrix} = w_q w_q^T,$$

where $w_q = [w_{1q}, w_{2q}, \ldots, w_{Pq}]^T$. That is, $K_{feature,q}$ is a rank-1 matrix and it captures the correlation across features. Therefore, we can further generalize the model to become $f_i(t) = \Sigma_{q=1}^{Q}\Sigma_{j=1}^{R} w_{iqj} g_q(t)$. That is, we are allowing multiple copies of $g_q(t)$. Then, we will have $K_{feature,q} = \Sigma_{j=1}^{R} w_{qj} w_{qj}^T$. In this case, the rank of $K_{feature,q}$ will become R if w_{qj} are independent where $w_{qj} = [w_{1qj}, w_{2qj}, \ldots, w_{Pqj}]^T$.

D.2.2 Multi-level Gaussian Process

As for modeling the correlation across subjects, a pair-wise correlation model such as $K(t, t')$ is not suitable. A multi-level model characterizes the correlation across subjects by comparing these subjects with a population mean. For example, [39, 40] use the mean function of a GP model to characterize the difference between individual and population. [39] designed a model for univariate longitudinal data where as [40] extended the model for multivariate longitudinal data. Since we can modify the mean function of our model similarly to account for correlation across subjects as shown in Section 4.2.5, we present the model in [40] here.

The authors modeled the heterogeneity across individuals by considering three levels of resolution in the mean function of the GP model: population, subpopulation, and individual levels. We can express the model as:

$$y_{pi}(t) \sim N(f_{pi}(t), \sigma_i^2) \tag{D.6}$$

$$f_{pi}(t) \sim GP(\underbrace{\Phi_{pop}^{(i)}(t)^T \Lambda^{(i)} x_p}_{PopulationLevel} + \underbrace{\Phi_{sub}^{(i)}(t)^T \beta_{z_{pi}}^{(i)}}_{SubpopulationLevel} + \underbrace{\Phi_{ind}^{(i)}(t)^T b_{pi}}_{IndividualLevel}, K_i(t, t')) \tag{D.7}$$

- The population level: The population level was meant to design the population effect of the baseline covariates such as gender, race, and age. The effect is assumed to be linear. Here, $\Phi_{pop}^{(i)}(t) \in \mathbf{R}^2$ is a basis function for the i-th feature which is modeled as a linear expansion of time in [40]. The matrix $\Lambda^{(i)}$ determines the mapping from the baseline covariates to the coefficients of the basis function $\Phi_{pop}^{(i)}(t)$. This matrix is a hyperparameter to be learned from the data

and is shared across subjects.

- The subpopulation level: The subpopulation level would affect subjects in the same cluster in similar ways. The model can be designed to learn the clusters based on the baseline covariates. Consider there are K clusters of individuals. We can model the likelihood of subject p to belong to cluster c_p as the following:

$$c_p \sim Multinomial(\pi_p), \pi_{p,k} = \frac{e^{w_k^T x_p}}{\sum_{k'=1}^{K} e^{{w_{k'}}^T x_p}}. \tag{D.8}$$

For each feature i, consider there are G_i clusters, where $z_{pi} \in \{1, \ldots, G_i\}$. We can then assume that given subject p belongs to cluster c_p, the likelihood of their i-th feature belonging to cluster z_{pi} is as follows:

$$z_{pi}|c_p \sim Multinomial(\Psi_{c_p}^{(i)}), \tag{D.9}$$

where $\Psi_{c_p}^{(i)}$ is a hyperparameter to be learned from the data.

That is, once we know which cluster a subject belongs to, we can learn the distributions of the clusters their features belong to. Based on the cluster z_{pi} each feature belongs to, a set of coefficients $\beta_{z_{pi}}^{(i)}$ are assigned. These coefficients are also hyperparameters to be learned from the data. $\Phi_{sub}^{(i)}(t)$ are basis functions. They were chosen to be B-spline basis expansion with degree two and eight interior knots evenly spaced in time.

Remark: We notice that both the multi-task model and the subpopulation level account for the correlation across the features. The multi-task model controls the correlation across the features at any given time. In our work, this correlation can be explained as how a person performs a task. A negative correlation between latency and error rate may imply a strategy to trade off between these two features. The subpopulation level controls the correlation across the coefficients of the basis functions. To make it more intuitive, if the basis functions are linear, then the subpopulation level learns the correlation across the "slopes" of the features. That is, the subpopulation level models

what the overall trends of the features are and how they are correlated. For example, the disease progression may affect the overall trends of the features. In this case, we may model the disease progression as in [40].

- The individual level:

  This term changes per subject and is designed to capture individual characteristics. Here, [40] modeled it using linear expansion of time. To learn $b_{pi} \in \mathbf{R}^2$, the slope and the intercept coefficient for subject p, we may assume $b_{pi} \sim N(0, B^{(i)})$. The covariance matrix $\Sigma^{(i)}$ is a hyperparameter shared across subjects and can be learned from the data.

- The covariance function for each subject:

  $K_i(t, t') = a_i^2 \exp\{-l_i^{-1}|t-t'|\}$. Here, a_i and l_i are hyperparameters to be learned from the data and are shared across subjects.

Appendix E

Learning and Inference Steps for the Extended Model

Here we look at the inference for the p-th subject. Since there is no confusion, we denote y_p as y. We notice that when we fix the cluster c_p subject p belongs to, the distribution of the eye-movement features is Gaussian. Therefore, by introducing the sub-population effect, the distribution of the eye-movement features becomes a mixture of Gaussians. As a result, to derive the inference, we can first derive the inference when the cluster c_p is fixed and simply use the mixture-model assumption to derive the rest. We re-order the indices in y and wrote it as a column vector $y = [y_1^T, \ldots, y_P^T]^T$. If we let $f_i(t) = \Sigma_{j=1}^{Q} w_{ij} g_j(t) + h_i(t)$, then we have $\mathrm{Cov}(f_i, f_{i'}|c) = \Sigma_j^Q w_{ij} w_{i'j} \mathrm{Cov}(g_j, g_j)$ if $i \neq i'$ and $\mathrm{Cov}(f_i, f_i) = \Sigma_j^Q w_{ij}^2 \mathrm{Cov}(g_j, g_j) + \mathrm{Cov}(h_i, h_i|c)$. (This is because $\mathrm{Cov}(h_i, h_i'|c) = 0$ if $i \neq i'$.)

If we write it in matrices it will look like this:

$$\mathrm{Var}\left(\begin{bmatrix} f_1(t_1) \\ \vdots \\ f_1(t_N) \\ \vdots \\ f_P(t_N) \end{bmatrix} |c\right) = \Sigma_{j=1}^{Q} \begin{bmatrix} w_{1j}^2 & \ldots & w_{1j}w_{Pj} \\ w_{2j}w_{1j} & \ldots & w_{2j}w_{Pj} \\ \vdots & \ddots & \vdots \\ w_{Pj}w_{1j} & \ldots & w_{Pj}^2 \end{bmatrix} \otimes \mathrm{Cov}(g_j, g_j) + \begin{bmatrix} \mathrm{Cov}(h_1, h_1|c) & 0 & \ldots & 0 \\ 0 & \mathrm{Cov}(h_2, h_2|c) & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & \ldots & \mathrm{Cov}(h_P, h_P|c) \end{bmatrix}$$

which will become

$$\mathrm{Var}(\begin{bmatrix} f_1(t_1) \\ \vdots \\ f_1(t_N) \\ \vdots \\ f_P(t_N) \end{bmatrix} |c) = \Sigma_{j=1}^{Q} \begin{bmatrix} w_{1j} \\ \vdots \\ w_{Pj} \end{bmatrix} \begin{bmatrix} w_{1j} & \dots & w_{Pj} \end{bmatrix} \otimes \mathrm{Cov}(g_j, g_j) + \begin{bmatrix} \mathrm{Cov}(h_1, h_1|c) & 0 & \dots & 0 \\ 0 & \mathrm{Cov}(h_2, h_2|c) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \mathrm{Cov}(h_P, h_P|c) \end{bmatrix}$$

By denoting the first term as K_{gg} and the second term as K_{hh}, we have:

$$\mathrm{Var}(\begin{bmatrix} y_{11} \\ \vdots \\ y_{1N} \\ \vdots \\ y_{PN} \end{bmatrix} |c) = K_{gg} + K_{hh|c} + \begin{bmatrix} \frac{1}{\beta_1} I_N & 0 & \dots & 0 \\ 0 & \frac{1}{\beta_2} I_N & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \frac{1}{\beta_P} I_N \end{bmatrix}$$

.

If we denote the noise term as Σ, we can similarly get:

$$\begin{bmatrix} y \\ f_* \end{bmatrix} |c \sim N(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K_{gg} + K_{hh|c} + \Sigma & K_{gg_*} + K_{hh_*|c} \\ K_{g_*g} + K_{h_*h|c} & K_{g_*g_*} + K_{h_*h_*|c} \end{bmatrix}),$$

where $K_{gg_*}, K_{hh_*|c}, K_{g_*g_*}, K_{h_*h_*|c}$ are all defined similarly.

We can then get

$$p(f_*|y, t, t_*, c) = N(f_*|K_{f_*f|c}(K_{ff|c}+\Sigma)^{-1}y, K_{f_*f_*|c} - K_{f_*f|c}(K_{ff|c}+\Sigma)^{-1}K_{ff_*|c}), \quad \text{(E.1)}$$

where $K_{ff|c} = K_{gg} + K_{hh|c}$, $K_{f_*f} = K_{g_*g} + K_{h_*h|c}$.

Note that to separate out the impact of the mean functions as in Section 2.7 in [47], we can rewrite $K_{hh|c}$ as $K_{hh0|c} + K_{hh1|c}$ where $K_{hh0|c}$ is the covariance matrix when we do not assume any mean function, and $K_{hh1|c}$ can be expressed as follows:

$$K_{hh1|c} = \begin{bmatrix} \Phi^h_{ind,1}(t)^T & 0 & \dots & 0 \\ 0 & \Phi^h_{ind,2}(t)^T & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \Phi^h_{ind,P}(t)^T \end{bmatrix} \begin{bmatrix} B^h_1 & 0 & \dots & 0 \\ 0 & B^h_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & B^h_p \end{bmatrix} \begin{bmatrix} \Phi^h_{ind,1}(t) & 0 & \dots & 0 \\ 0 & \Phi^h_{ind,2}(t) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \Phi^h_{ind,P}(t) \end{bmatrix}.$$

Here we notice that neither $K_{hh1|c}$ nor $K_{hh0|c}$ depends on c. As a result, we can remove c in the notation and write $K_{hh1} = \tilde{\Phi}^T \tilde{B} \tilde{\Phi}$ and let $K = K_{gg} + K_{hh0} + \Sigma$. We then could have likelihood function as

$$\log p(y|t, \tilde{B}, c) = -\frac{1}{2} y^T K^{-1} y + \frac{1}{2} y^T C y - \frac{1}{2} \log |K| - \frac{1}{2} \log \tilde{B} - \frac{1}{2} \log A + constant, \quad \text{(E.2)}$$

where $A = \tilde{B}^{-1} + \tilde{\Phi} K^{-1} \tilde{\Phi}^T$ and $C = K^{-1} \tilde{\Phi}^T A^{-1} \tilde{\Phi} K^{-1}$. The prediction then will be

$$p(f_*|y, x, x_*, c) = N(f_*|\Phi^h_{sub,i}(t_*)^T r^h_{sub,c,i} + f_{*0} + R^T \beta, K_{f_{*0} f_{*0}} + R^T A^{-1} R), \quad \text{(E.3)}$$

where f_{*0} and $K_{f_{*0} f_{*0}}$ denote the mean and covariance functions without the impact of the mean functions, $R = \tilde{\Phi}_* - \tilde{\Phi} K^{-1} K_*$, and $\beta = A^{-1} \tilde{\Phi} K^{-1} y$. The derivation for $p(f_*|y, x, x_*, c)$ follows the derivation for a Gaussian mixture model.

Appendix F

Stochastic Variational Inference for the Extended Model

Here the model is slightly more generalized than presented in Section 4.2.5. We assume:

$$p(y|g,h) = \prod_{i=1}^{P}\prod_{n=1}^{N} N(y_{in}; \Sigma_{j=1}^{Q} w_{ij} g_j(t_n) + h_i(t_n), \beta_i^{-1}), \tag{F.1}$$

where

$$h_i \sim GP(\Phi_{sub,i}^h(t)^T r_{sub,c,i}^h + \Phi_i^h(t)^T r_i^h, K_i^h(t,t')), \tag{F.2}$$

$$c|x \sim Multinomial(\pi_c), \pi_c \propto \exp w_c^T x, \tag{F.3}$$

$$r_i^h \sim N(0, B_i^h), \tag{F.4}$$

$$g_j \sim GP(\Phi_j(t)^T r_{jp}, K_j(t,t')), \tag{F.5}$$

$$r_{jp} \sim N(0, B_j). \tag{F.6}$$

F.1 ELBO calculation

The idea to reduce complexity is to introduce *inducing variables* u and v (parallel to g and h correspondingly) with M data points and use variational inference to approximate the posterior probability $p(u, v, c|y)$. The approximate distribution is

$$q(u, v, c) = \prod_{j=1}^{Q} N(u_j; m_j, S_j) \prod_{i=1}^{P} N(v_i; m_i^h, S_i^h) p(c). \tag{F.7}$$

$$\begin{aligned}
&KL(q(u, v, c)||p(u, v, c|y)) \\
=&\mathbb{E}_{q(u,v,c)}[\log \frac{q(u, v, c)}{p(u, v, c|y)}] \geq 0 \\
=&\log p(y) - (\mathbb{E}_{q(u,v,c)}[\log p(u, v, c, y)] - \mathbb{E}_q[\log q(u, v, c)]) \\
=&\log p(y) - (\mathbb{E}_{q(u,v,c)}[\log p(y|u, v, c)] + \mathbb{E}_q[\log p(u, v, c)] - \mathbb{E}_q[\log q(u, v, c)]) \\
=&\log p(y) - (\mathbb{E}_{q(u,v,c)}[\log p(y|u, v, c)] - KL(q(u)||p(u)) - KL(q(v)||p(v)))
\end{aligned} \tag{F.8}$$

The variational lower bound then becomes

$$\log p(y) \geq \mathbb{E}_{q(u,v,c)}[\log p(y|u, v, c)] - KL(q(u)||p(u)) - KL(q(v)||p(v)) = ELBO \tag{F.9}$$

We now evaluate each term in Eq.(F.9). For the first term, we notice that

$$\begin{aligned}
\log p(y|u, v, c) &= \log \mathbb{E}_{p(g,h|u,v,c)}[p(y|g, h, c)] \\
&\geq \mathbb{E}_{p(g,h|u,v,c)}[\log p(y|g, h, c)]
\end{aligned} \tag{F.10}$$

The inequality is introduced to reduce the computation complexity from $\mathcal{O}(P^3N^3)$ to $\mathcal{O}(P^3M^3)$.

Now we need to evaluate $\mathbb{E}_{p(g,h|u,v,c)}[\log p(y|g, h, c)]$. This will require the following identity. Assume $p(y|g, h) = N(y; \Sigma_{j=1}^{Q} W_j g_j + Wh, \beta^{-1}I)$ with $p(g_j) = N(g_j; m_j, S_j)$

and $p(h) = N(h; m, S)$.

$$\begin{aligned}&\int \log p(y|g,h)dp(g)dp(h)\\ =&\log N(y; \Sigma_{j=1}^{Q} W_j m_j + Wm, \beta^{-1}I) - \frac{1}{2}\beta tr W^T W S - \frac{1}{2}\beta tr \Sigma_{j=1}^{Q} W_j^T W_j S_j.\end{aligned} \tag{F.11}$$

To apply Eq.(F.11) on $\mathbb{E}_{p(g,h|u,v,c)}[\log p(y|g,h,c)]$, we need to first evaluate $p(g|u)$ and $p(h|v,c)$. We let the inducing input be z_j and z_i^h correspondingly. We have

$$\begin{aligned}p(g|u) &= \prod N(g_j; \mu_j, K_{gj})\\ p(u) &= \prod N(u_j; 0, K_{uj})\\ p(h|v,c) &= \prod N(h_i; \mu_{ic}^h, K_{hi})\\ p(v) &= \prod N(v_i; 0, K_{vi})\end{aligned} \tag{F.12}$$

where

$$\begin{aligned}\mu_j &= K_j(t, z_j)K_j(z_j, z_j)^{-1}u_j + R_j^T \bar{r}_j = A_j u_j\\ K_{gj} &= K_j(t,t) - K_j(t, z_j)K_j(z_j, z_j)^{-1}K_j(z_j, t)\\ &\quad + R_j^T(B_j^{-1} + \Phi_j(z_j)K_j(z_j, z_j)^{-1}\Phi_j(z_j)^T)^{-1}R_j\\ \bar{r}_j &= [B_j^{-1} + \Phi_j(z_j)K_j(z_j, z_j)^{-1}\Phi_j(z_j)^T]^{-1}[\Phi_j(z_j)K_j(z_j, z_j)^{-1}u_j]\\ R_j &= \Phi_j(t) - \Phi_j(z_j)K_j(z_j, z_j)^{-1}K_j(z_j, t)\end{aligned} \tag{F.13}$$

$$K_{uj} = K_j(z_j, z_j) + \Phi_j(z_j)^T B_j \Phi_j(z_j) \tag{F.14}$$

$$\begin{aligned}\mu_{ic}^h &= K_i^h(x, z_i^h)K_i^h(z_i^h, z_i^h)^{-1}v_i + (R_i^h)^T \bar{r}_i^h + \Phi_{sub,i}^h(t)^T r_{sub,c,i}^h = A_i^h v_i + \Phi_{sub,i}^h(t)^T r_{sub,c,i}^h\\ K_{hi} &= K_i^h(x,x) - K_i^h(x, z_i^h)K_i^h(z_i^h, z_i^h)^{-1}K_i^h(z_i^h, x)\\ &\quad + (R_i^h)^T((B_i^h)^{-1} + \Phi_i^h(z_i^h)K_i^h(z_i^h, z_i^h)^{-1}\Phi_i^h(z_i)^{hT})^{-1}R_i^h\\ \bar{r}_i^h &= [(B_i^h)^{-1} + \Phi_i^h(z_i^h)K_i^h(z_i^h, z_i^h)^{-1}\Phi_i^h(z_i^h)^T]^{-1}[\Phi_i^h(z_i^h)K_i^h(z_i^h, z_i^h)^{-1}v_i]\\ R_i^h &= \Phi_i^h(x) - \Phi_i^h(z_i^h)K_i^h(z_i^h, z_i^h)^{-1}K_i^h(z_i^h, x)\end{aligned} \tag{F.15}$$

$$K_{vi} = K_j(z_i^h, z_i^h) + \Phi_i^h(z_i^h)^T B_i^h \Phi_i^h(z_i^h). \tag{F.16}$$

Since $\log p(y|h,g,c) = \Sigma_{i,n} \log N(y_{in}; \Sigma_{j=1}^Q w_{ij} g_j(x_n) + h_{ic}(x_n), \beta_i^{-1})$, by Eq(F.11) we have

$$\begin{aligned}
&\mathbb{E}_{p(g,h|u,v,c)}[\log p(y|g,h,c)] \\
=&\Sigma_{i,n} \log N(y_{in}; \Sigma_{j=1}^Q w_{ij}\mu_j + \mu_{ic}^h, \beta_i^{-1}) - \frac{1}{2}\beta_i (K_{hi})_{nn} - \frac{1}{2}\beta_i \Sigma_{j=1}^Q w_{ij}^2 (K_{gj})_{nn}.
\end{aligned} \tag{F.17}$$

Now we apply Eq(F.11) on $\mathbb{E}_{q(u,v,c)}[\log N(y_{in}; \Sigma_{j=1}^Q w_{ij}\mu_j + \mu_{ic}^h, \beta_i^{-1})]$, we get

$$\begin{aligned}
&\mathbb{E}_{p(c)}\mathbb{E}_{q(u,v|c)}[\log N(y_{in}; \Sigma_{j=1}^Q w_{ij} A_j(n,:)u_j + A_i^h(n,:)v_i + \Phi_{sub,i}^T r_{sub,i,c}^h, \beta_i^{-1})|c] \\
=&\sum_c p(c) \log N(y_{in}; \Sigma_{j=1}^Q w_{ij} A_j(n,:)m_j + A_i^h(n,:)m_i^h + \Phi_{sub,i}^h r_{sub,i,c}^h, \beta_i^{-1}) \\
&- \frac{1}{2}\beta_i tr\Lambda_{in}^h S_i^h - \frac{1}{2}\beta_i \Sigma_{j=1}^Q w_{ij}^2 tr\Lambda_{jn} S_j,
\end{aligned} \tag{F.18}$$

where

$$\begin{aligned}
\Lambda_{jn} &= A_j(n,:)^T A_j(n,:) \\
\Lambda_{in}^h &= A_i^h(n,:)^T A_i^h(n,:).
\end{aligned} \tag{F.19}$$

What is left in Eq.(F.9) is the two KL divergence terms. We can use the identity for KL divergence of two M-dimensional multivariate Gaussian distributions. Assume $q \sim N(\mu_1, \Sigma_1)$ and $p \sim N(\mu_2, \Sigma_2)$. Then

$$KL(q||p) = \frac{1}{2}\log|\Sigma_2\Sigma_1^{-1}| + \frac{1}{2} tr\Sigma_2^{-1}[(\mu_2 - \mu_1)(\mu_2 - \mu_1)^T + \Sigma_1] - \frac{M}{2}. \tag{F.20}$$

When we try to maximize ELBO, we can ignore the constant. As a result, we now

can use Eq.(F.17),Eq.(F.18),Eq.(F.20) to rewrite Eq.(F.9) as

$$\begin{aligned} ELBO = L = \Sigma_{i,n}[&\sum_c p(c) \log N(y_{in}; \tilde{\mu}_{inc}, \beta_i^{-1}) \\ &- \frac{1}{2}\beta_i (K_{hi})_{nn} - \frac{1}{2}\beta_i \Sigma_{j=1}^{Q} w_{ij}^2 (K_{gj})_{nn} \\ &- \frac{1}{2}\beta_i tr\Lambda_{in}^h S_i^h - \frac{1}{2}\beta_i \Sigma_{j=1}^{Q} w_{ij}^2 tr\Lambda_{jn} S_j] \\ &- \Sigma_{j=1}^{Q}[\frac{1}{2}\log|K_{uj}S_j^{-1}| + \frac{1}{2}trK_{uj}^{-1}(m_j m_j^T + S_j)] \\ &- \Sigma_{i=1}^{P}[\frac{1}{2}\log|K_{vi}(S_i^h)^{-1}| + \frac{1}{2}trK_{vi}^{-1}(m_i^h (m_i^h)^T + S_i^h)], \end{aligned} \tag{F.21}$$

where

$$\tilde{\mu}_{inc} = \Sigma_{j=1}^{Q} w_{ij} A_j(n,:) m_j + A_i^h(n,:) m_i^h + \Phi_{sub,i}^h(t_n)^T r_{sub,i,c}^h. \tag{F.22}$$

F.2 Stochastic Variational Inference

Our goal is to optimize ELBO. We first consider the derivatives of L with respect to m_j, S_j, m_i^h, S_i^h. Say o_i denotes the indices where y_i exist. The derivatives are as follows:

$$\begin{aligned} \frac{\partial L}{\partial m_j} &= \Sigma_{i=1}^{P} \beta_i w_{ij} A_j(o_i)^T [y_i - \Sigma_{j'=1}^{Q} w_{ij'} A_{j'}(o_i) m_{j'} - A_i^h(o_i) m_i^h] - K_{uj}^{-1} m_j \\ &= \Sigma_{i=1}^{P} \beta_i w_{ij} A_j(o_i)^T y_i^{/j} - [K_{uj}^{-1} + \Sigma_{i=1}^{P} \beta_i w_{ij}^2 A_j(o_i)^T A_j(o_i)] m_j \\ \frac{\partial L}{\partial S_j} &= \frac{1}{2} S_j^{-1} - \frac{1}{2}[K_{uj}^{-1} + \Sigma_{i=1}^{P} \beta_i w_{ij}^2 A_j(o_i)^T A_j(o_i)] \\ \frac{\partial L}{\partial m_i^h} &= \sum_c p(c) \beta_i A_i^h(o_i)^T [y_i - \Sigma_{j=1}^{Q} w_{ij} A_j(o_i) m_j - A_i^h(o_i) m_i^h - \Phi_{sub,i}^h(t_n)^T r_{sub,i,c}^h] \\ &\quad - K_{vi}^{-1} m_i^h \\ &= \beta_i A_i^h(o_i)^T y_i^{/h} - [K_{vi}^{-1} + \beta_i A_i^h(o_i)^T A_i^h(o_i)] m_i^h \\ &\quad - \sum_c p(c) \beta_i A_i^h(o_i)^T \Phi_{sub,i}^h(t_n)^T r_{sub,i,c}^h \\ \frac{\partial L}{\partial S_i^h} &= \frac{1}{2}(S_i^h)^{-1} - \frac{1}{2}[K_{vi}^{-1} + \beta_i A_i^h(o_i)^T A_i^h(o_i)], \end{aligned} \tag{F.23}$$

where

$$\begin{aligned} y_i^{/j} &= y_i - A_i^h(o_i)m_i^h - \Sigma_{j' \neq j} w_{ij'} A_{j'}(o_i) m_{j'} \\ y_i^{/h} &= y_i - \Sigma_{j=1}^{Q} w_{ij} A_j(o_i) m_j \end{aligned} \tag{F.24}$$

As shown in [68, 50], the natural gradient simplifies the formulations for updating the canonical parameters $\Psi_{1j} = S_j^{-1} m_j, \Psi_{2j} = -\frac{1}{2} S_j^{-1}$, since the natural gradients turned out to be $\partial L / \partial m_j$ and $\partial L / \partial S_j$. Thus we have:

$$\begin{aligned} \Psi_{1j(k+1)} &= S_{j(k)}^{-1} m_{j(k)} + l(\Sigma_{i=1}^{P} \beta_i w_{ij} A_j(o_i)^T y_i^{/j} - S_{j(k)}^{-1} m_{j(k)}) \\ \Psi_{2j(k+1)} &= -\frac{1}{2} S_{j(k)}^{-1} + l(\frac{1}{2} S_{j(k)}^{-1} - \frac{1}{2}\Lambda), \end{aligned} \tag{F.25}$$

where

$$\Lambda = K_{uj}^{-1} + \Sigma_{i=1}^{P} \beta_i w_{ij}^2 A_j(o_i)^T A_j(o_i). \tag{F.26}$$

Similarly, we have

$$\begin{aligned} \Psi_{1i(k+1)}^h &= (S_{i(k)}^h)^{-1} m_{i(k)}^h \\ &\quad + l[\beta_i A_i^h(o_i)^T y_i^{/h} - (S_{i(k)}^h)^{-1} m_{i(k)}^h - \sum_c p(c) \beta_i A_i^h(o_i)^T \Phi_{sub,i}^h(t_n)^T r_{sub,c,i}^h] \\ \Psi_{2i(k+1)}^h &= -\frac{1}{2}(S_{i(k)}^h)^{-1} + l[\frac{1}{2}(S_{i(k)}^h)^{-1} - \frac{1}{2}\Lambda^h], \end{aligned} \tag{F.27}$$

where

$$\Lambda^h = K_{vi}^{-1} + \beta_i A_i^h(o_i)^T A_i^h(o_i). \tag{F.28}$$

The hyperparameters includes all the coefficients in the covariance functions K_j and K_i^h, the noise β_i, the inducing inputs Z_j, Z_i^h, the matrices B_j and B_i^h, and the mapping w_c for the sub-population effect. These hyperparameters can also be learned by optimizing over ELBO. That is, one can interleave the optimization over the variational parameters and the hyperparameters.

Prediction: Let $p(g_{j*}|y, t_*) = N(g_{j*}; \mu_{j*}, s_{j*})$ and $p(h_{i*}|y, t_*) = N(h_{i*}; \mu_{i*}^h, s_{i*}^h)$, and also let $g_{j*} = A_{j*} u_j + \epsilon_j$ where A_{j*} can be derived similarly as A_j and $\epsilon_j \sim N(0, K_{gj*})$. We also know that $u_j = m_j + \tilde{\epsilon}_j$ where $\tilde{\epsilon}_j \sim N(0, S_j)$. As a result, $A_{j*} u_j \sim N(A_{j*} m_j, A_{j*} S_j A_{j*}^T)$ and $\mu_{j*} = A_{j*} m_j$ and $s_{j*} = K_{gj*} + A_{j*} S_j A_{j*}^T$. Similarly,

we could calculate $p(h_{i*}|y, t_*, c) = N(h_{i*}; \mu^h_{ic*}, s^h_{i*})$. The rest of the derivation follows the assumption of a Gaussian mixture.

Bibliography

[1] D. Li, D. Winfield, and D. Parkhurst, "Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) - Workshops*, 2005, pp. 79–87.

[2] 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 (CVPR)*, 2016, pp. 2176–2184.

[3] T. Anderson and M. MacAskill, "Eye movements in patients with neurodegenerative disorders," *Nature Reviews Neurology*, vol. 9, no. 2, pp. 74–85, 2013.

[4] R. Leigh and D. Zee, "The saccadic system," in *The Neurology of Eye Movements*. Oxford: Oxford University Press, 2015, ch. 4, pp. 169–288.

[5] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001, pp. 511–518.

[6] J. Owens, T. Dingus, F. Guo, Y. Fang, M. Perez, J. McClafferty, and B. Tefft, "Prevalence of drowsy driving crashes: Estimates from a large-scale naturalistic driving study (research brief)," *AAA Foundation for Traffic Safety*, 2018.

[7] S. Hoops, S. Nazem, A. Siderowf, J. Duda, S. Xie, M. Stern, and D. Weintraub, "Validity of the MoCA and MMSE in the detection of MCI and dementia in Parkinson disease," *Neurology*, vol. 73, no. 21, pp. 1738–1745, 2009.

[8] A. Mitchell, "A meta-analysis of the accuracy of the mini-mental state examination in the detection of dementia and mild cognitive impairment," *Journal of Psychiatric Research*, vol. 43, no. 4, pp. 411–431, 2009.

[9] National Academies of Sciences, Engineering, and Medicine, *Harnessing mobile devices for nervous system disorders: Proceedings of a Workshop*. Washington, DC: The National Academies Press, 2018.

[10] H. Posner, R. Curiel, C. Edgar, S. Hendrix, E. Liu, D. Loewenstein, L. Morrison, G. Shinobu, K. Wesnes, and P. Harvey, "Outcomes assessment in clinical trials of Alzheimer's disease and its precursors: Readying for short-term and long-term

clinical trial needs," *Innovations in Clinical Neuroscience*, vol. 14, no. 1-2, pp. 22–29, 2017.

[11] P. Harvey, S. Cosentino, R. Curiel, T. Goldberg, J. Kaye, D. Lowenstein, D. Marson, D. Salmon, K. Wesnes, and H. Posner, "Performance-based and observational assessments in clinical trials across the Alzheimer's disease spectrum," *Innovations in Clinical Neuroscience*, vol. 14, no. 1-2, pp. 30–39, 2017.

[12] S. Tabrizi, D. Langbehn, B. Leavitt, R. Roos, A. Durr, D. Craufurd, C. Kennard, S. Hicks, N. Fox, R. Scahill, B. Borowsky, A. Tobin, H. Rosas, H. Johnson, R. Reilmann, B. Landwehrmeyer, and J. Stout, "Biological and clinical manifestations of Huntington's disease in the longitudinal track-hd study: Cross-sectional analysis of baseline data," *The Lancet Neurology*, vol. 8, no. 9, pp. 791–801, 2009.

[13] D. Munoz and S. Everling, "Look away: the anti-saccade task and the voluntary control of eye movement," *Nature Review Neuroscience*, vol. 5, no. 3, pp. 218–228, 2004.

[14] J. M. JE, K. Dyckman, B. Austin, and B. Clementz, "Neurophysiology and neuroanatomy of reflexive and volitional saccades: evidence from studies of humans," *Brain and Cognition*, vol. 68, no. 3, pp. 255–270, 2008.

[15] R. Shafiq-Antonacci, P. Maruff, C. Masters, and J. Currie, "Spectrum of saccade system function in Alzheimer's disease," *Archives of Neurology*, vol. 60, no. 9, pp. 1275–1278, 2003.

[16] T. Crawford, S. Higham, T. Renvoize, J. Patel, M. Dale, A. Suriya, and S. Tetley, "Inhibitory control of saccadic eye movements and cognitive impairment in Alzheimer's disease," *Biological Psychiatry*, vol. 57, no. 9, pp. 1052–1060, 2005.

[17] U. Mosimann, R. Müri, D. Burn, J. Felblinger, J. O'Brien, and I. McKeith, "Saccadic eye movement changes in Parkinson's disease dementia and dementia with Lewy bodies," *Brain*, vol. 128, no. 6, pp. 1267–1276, 2005.

[18] S. Garbutt, A. Matlin, J. Hellmuth, A. Schenk, J. Johnson, H. Rosen, D. Dean, J. Kramer, J. Neuhaus, B. Miller, S. Lisberger, and A. Boxer, "Oculomotor function in frontotemporal lobar degeneration, related disorders and Alzheimer's disease," *Brain*, vol. 131, no. 5, pp. 1268–1281, 2008.

[19] C. Antoniades, Z. Xu, S. Mason, R. Carpenter, and R. Barker, "Huntington's disease: Changes in saccades and hand-tapping over 3 years," *Journal of Neurology*, vol. 257, no. 11, pp. 1890–1898, 2010.

[20] M. Proudfoot, R. Menke, R. Sharma, C. Berna, S. Hicks, C. Kennard, K. Talbot, and M. Turner, "Eye-tracking in amyotrophic lateral sclerosis: A longitudinal study of saccadic and cognitive tasks," *Amyotrophic Lateral Sclerosis & Frontotemporal Degeneration*, vol. 17, no. 1-2, pp. 101–111, 2015.

[21] E. Dorsey, S. Papapetropoulos, M. Xiong, and K. Kieburtz, "The first frontier: Digital biomarkers for neurodegenerative diseases," *Digital Biomarkers*, vol. 1, pp. 6–13, 2017.

[22] E. Dorsey, A. Glidden, M. Holloway, G. Birbeck, and L. Schwamm, "Teleneurology and mobile technologies: The future of neurological care," *Nature Reviews Neurology*, vol. 14, no. 5, pp. 285–297, 2018.

[23] D. Veitch, M. Weiner, P. Aisen, L. Beckett, N. Cairns, R. Green, D. Harvey, C. Jack, W. Jagust, J. Morris, R. Petersen, A. Saykin, L. Shaw, A. Toga, and J. Trojanowski, "Understanding disease progression and improving alzheimer's disease clinical trials: Recent highlights from the alzheimer's disease neuroimaging initiative," *Alzheimer's and Dementia*, vol. 15, no. 1, pp. 106–152, 2019.

[24] J. Neville, S. Kopko, S. Broadbent, E. Avilés, R. Stafford, C. Solinsky, L. Bain, M. Cisneroz, K. Romero, and D. Stephenson, "Development of a unified clinical trial database for alzheimer's disease," *Alzheimer's and dementia : the journal of the Alzheimer's Association*, vol. 11, no. 10, 2015.

[25] J. Latourelle, M. Beste, T. Hadzi, R. Miller, J. Oppenheim, M. Valko, D. Wuest, B. Church, I. Khalil, B. Hayete, and C. Venuto, "Large-scale identification of clinical and genetic predictors of motor progression in patients with newly diagnosed parkinson's disease: a longitudinal cohort study and validation," *The Lancet Neurology*, vol. 16, no. 11, pp. 908–916, 2017.

[26] S. Tabrizi, R. Scahill, G. Owen, A. Durr, B. Leavitt, R. Roos, B. Borowsky, B. Landwehrmeyer, C. Frost, H. Johnson, D. Craufurd, R. Reilmann, J. Stout, and D. Langbehn, "Predictors of phenotypic progression and disease onset in premanifest and early-stage huntington's disease in the track-hd study: analysis of 36-month observational data," *The Lancet Neurology*, vol. 12, no. 7, pp. 637–649, 2013.

[27] J. Paulsen, J. Long, H. Johnson, E. Aylward, C. Ross, J. Williams, M. Nance, C. Erwin, H. Westervelt, D. Harrington, H. Bockholt, Y. Zhang, E. McCusker, E. Chiu, and P. Panegyres, "Clinical and biomarker changes in premanifest huntington disease show trial feasibility: a decade of the predict-hd study," *Frontiers in Aging Neuroscience*, vol. 6, p. 78, 2014.

[28] G. Saavedra-Peña, "Saccade latency determination using video recordings from consumer-grade devices," Master's thesis, Massachusetts Institute of Technology, 2018.

[29] J. Hlavnička, R. Čmejla, T. Tykalová, K. Šonka, E. Růžička, and J. Rusz, "Automated analysis of connected speech reveals early biomarkers of Parkinson's disease in patients with rapid eye movement sleep behavior disorder," *Scientific reports*, vol. 7, no. 12, pp. 1–13, 2017.

[30] J. Hanuška, J. Rusz, O. Bezdicek, O. Ulmanová, C. Bonnet, P. Dušek, V. Ibarburu, T. Nikolai, T. Sieger, K. Šonka, and E. Růžička, "Eye movements in idiopathic rapid eye movement sleep behavior disorder: High antisaccade error rate reflects prefrontal cortex dysfunction," *Journal of Sleep Research*, vol. e12742, 2018.

[31] J. Holden, A. Cosnard, B. Laurens, J. Asselineau, D. Biotti, S. Cubizolle, S. Dupouy, M. Formaglio, L. Koric, M. Seassau, C. Tilikete, A. Vighetto, and F. Tison, "Prodromal alzheimer's disease demonstrates increased errors at a simple and automated anti-saccade task," *Journal of Alzheimer's Disease*, vol. 65, no. 4, pp. 1209–1223, 2018.

[32] A. Piau, K. Wild, N. Mattek, and J. Kaye, "Current state of digital biomarker technologies for real-life, home-based monitoring of cognitive function for mild cognitive impairment to mild alzheimer disease and implications for clinical care: Systematic review," *Journal of Medical Internet Research*, vol. 21, no. 8, p. e12785, 2019.

[33] L. Kourtis, O. Regele, J. Wright, and G. Jones, "Digital biomarkers for alzheimer's disease: the mobile/wearable devices opportunity," *npj Digital Medicine*, vol. 2, no. 9, 2019.

[34] A. S. de Lima, L. Evers, T. Hahn, L. Bataille, J. Hamilton, M. Little, Y. Okuma, B. Bloem, and M. Faber, "Freezing of gait and fall detection in parkinson's disease using wearable sensors: a systematic review. journal of neurology," *Journal of neurology*, vol. 264, no. 8, pp. 1642–1654, 2017.

[35] O. Ferhat and F. Vilariño, "Low cost eye tracking: The current panorama," *Computational Intelligence and Neuroscience*, vol. 3, pp. 1–14, 2016.

[36] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "It's written all over your face: Full-face appearance-based gaze estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 2299–2308.

[37] H.-Y. Lai, G. Saavedra-Peña, C. Sodini, T. Heldt, and V. Sze, "Enabling saccade latency measurements with consumer-grade cameras," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 3169–3173.

[38] X. Wang, D. Sontag, and F. Wang, "Unsupervised learning of disease progression models," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '14. Association for Computing Machinery, 2014, p. 85–94.

[39] P. Schulam and S. Saria, "A framework for individualizing predictions of disease trajectories by exploiting multi-resolution structure," in *Neural Information Processing Systems (NIPS)*, 2015.

[40] J. Futoma, M. Sendak, B. Cameron, and K. Heller, "Predicting disease progression with a model for multivariate longitudinal clinical data," in *Machine Learning for Healthcare*, 2016.

[41] L.-F. Cheng, G. Darnell, B. Dumitrascu, C. Chivers, M. Draugelis, K. Li, and B. Engelhardt, "Sparse multi-output gaussian processes for medical time series prediction," 2018.

[42] O. Rudovic, Y. Utsumi, R. Guerrero, K. Peterson, D. Rueckert, and R. Picard, "Meta-weighted gaussian process experts for personalized forecasting of ad cognitive changes," 2019.

[43] T. Wang, R. Qiu, and M. Yu, "Predictive modeling of the progression of alzheimer's disease with recurrent neural networks," *Scientific Reports*, vol. 8, no. 9161, 2006.

[44] T. Garcia and K. Marder, "Statistical approaches to longitudinal data analysis in neurodegenerative diseases: Huntington's disease as a model," *Current Neurology and Neuroscience Reports*, vol. 17, no. 2, 2017.

[45] N. Oxtoby, A. Young, D. Cash, T. Benzinger, A. Fagan, J. Morris, R. Bateman, N. Fox, J. Schott, and D. Alexander, "Data-driven models of dominantly-inherited alzheimer's disease progression," *Brain*, vol. 141, no. 5, pp. 1529–1544, 2018.

[46] N. Oxtoby, L.-A. Leyland, L. Aksman, G. Thomas, E. Bunting, P. Wijeratne, A. Young, A. Zarkali, M. Tan, F. Bremner, P. Keane, H. Morris, A. Schrag, D. Alexander, and R. Weil, "Sequence of clinical and neurodegeneration events in parkinson's disease progression," *Brain*, vol. 144, no. 3, pp. 975–988, 2021.

[47] C.E. Rasmussen and C.K.I Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2005.

[48] E. Bonilla, K. Chai, and C. Williams, "Multi-task gaussian process prediction," in *Advances in Neural Information Processing Systems*, vol. 20, 2008.

[49] H. Wackernagel, *Multivariate Geostatistics: An Introduction with Applications*. Springer-Verlag, Berlin, 2nd edition, 1998.

[50] V. Nguyen and E. Bonilla, "Collaborative multi-output gaussian processes," in *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, ser. UAI'14, 2014, p. 643–652.

[51] "Psychophysics Toolbox 3," http://psychtoolbox.org/, accessed: 02-09-2018.

[52] S. Rivaud-Péchoux, M. Vidailhet, J. Brandel, and B. Gaymard, "Mixing pro- and antisaccades in patients with parkinsonian syndromes," *Brain*, vol. 130, no. 1, pp. 256–264, 2006.

[53] Q. Yang, T. Wang, N. Su, S. Xiao, and Z. Kapoula, "Specific saccade deficits in patients with Alzheimer's disease at mild to moderate stage and in patients with amnestic mild cognitive impairment," *Age*, vol. 35, no. 4, pp. 1287–1298, 2013.

[54] A. Davis, M. Rubinstein, N. Wadhwa, G. Mysore, F. Durand, and W. Freeman, "The visual microphone: Passive recovery of sound from video," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 79:1–79:10, Jul. 2014. [Online]. Available: http://doi.acm.org/10.1145/2601097.2601119

[55] E. Simoncelli and W. Freeman, "The steerable pyramid: a flexible architecture for multi-scale derivative computation," in *Proceedings., International Conference on Image Processing*, vol. 3, 1995, pp. 444–447 vol.3.

[56] K. Holmqvist, M. Nyström, R. Andersson, R. Dewhurst, H. Jarodzka, and J. van de Weijer, "Eye-tracker hardware and its properties," in *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford: Oxford University Press, 2011, ch. 2, pp. 48–49.

[57] D. Mack, S. Belfanti, and U. Schwarz, "The effect of sampling rate and lowpass filters on saccades – a modeling approach," *Behavioral Research*, vol. 49, no. 6, pp. 2146–2162, 2017.

[58] "Package pROC," https://cran.r-project.org/web/packages/pROC/pROC.pdf, accessed: 12-12-2018.

[59] M. Nyström and K. Holmqvist, "An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data," *Behavior Research Methods*, vol. 42, no. 1, pp. 188–204, 2010.

[60] A. Savitzky and M. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[61] F. Gustafsson, "On-line approaches," in *Adaptive Filtering and Change Detection*. John Wiley & Sons, Ltd, 2001, ch. 3, pp. 55–87.

[62] E. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, no. 1/2, pp. 100–115, 1954.

[63] W. van der Linden, "A lognormal model for response times on test items," *Journal of Educational and Behavioral Statistics*, vol. 31, no. 2, pp. 181–204, 2006.

[64] J. Weir, "Quantifying test-retest reliability using the intraclass correlation coefficient and the SEM," *Journal of Strength and Conditioning Research*, vol. 19, no. 1, pp. 231–240, 2005.

[65] D. Munoz, J. Broughton, J. Goldring, and I. Armstrong, "Age-related performance of human subjects on saccadic eye movement tasks," *Experimental Brain Research*, vol. 121, no. 4, pp. 391–400, 1998.

[66] B. Fischer, M. Biscaldi, and S. Gezeck, "On the development of voluntary and reflexive components in human saccade generation," *Brain Res.*, vol. 754, no. 1-2, pp. 285–597, 1997.

[67] H. Liu, Y.-S. Ong, X. Shen, and J. Cai, "When gaussian process meets big data: A review of scalable gps," 2019.

[68] J. Hensman, N. Fusi, and N. D. Lawrence, "Gaussian processes for big data," 2013.

[69] S. Rivaud, R. Müri, B. Gaymard, A. Vermersch, and C. Pierrot-Deseilligny, "Eye movement disorders after frontal eye field lesions in humans," *Experimental Brain Research*, vol. 102, no. 1, pp. 110–120, 1994.

[70] C. Pierrot-Deseilligny, R. Müri, C. Ploner, B. Gaymard, S. Demeret, and S. Rivaud-Pechoux, "Decisional role of the dorsolateral prefrontal cortex in ocular motor behaviour," *Brain*, vol. 126, no. 6, pp. 1460–1473, 2003.

[71] C. Theleritis, I. Evdokimidis, and N. Smyrnis, "Variability in the decision process leading to saccades: A specific marker for schizophrenia?" *Psychophysiology*, vol. 51, no. 4, pp. 327–336, 2014.

[72] T. Karantinos, E. Tsoukas, A. Mantas, E. Kattoulas, N. Stefanis, I. Evdokimidis, and N. Smyrnis, "Increased intra-subject reaction time variability in the volitional control of movement in schizophrenia," *Psychiatry Research*, vol. 215, no. 1, pp. 26–32, 2014.

[73] R. Ulrich and J. Miller, "Information processing models generating lognormally distributed reaction times," *Journal of Mathematical Psychology*, vol. 37, no. 4, pp. 513 – 525, 1993.

[74] C. Bonnet, J. Rusz, M. Megrelishvili, T. Sieger, O. Matoušková, M. Okujava, H. Brožová, T. Nikolai, J. Hanuška, M. Kapianidze, N. Mikeladze, N. Botchorishvili, I. Khatiashvili, M. Janelidze, T. Serranová, O. Fiala, J. Roth, J. Bergquist, R. Jech, S. Rivaud-Péchoux, B. Gaymard, and E. Růžička, "Eye movements in ephedrone-induced parkinsonism," *PLoS one*, vol. 9, no. 8, pp. 1–8, 2014.

[75] A. Boxer, S. Garbutt, W. Seeley, A. Jafari, H. Heuer, J. Mirsky, J. Hellmuth, J. Trojanowski, E. Huang, S. DeArmond, J. Neuhaus, and B. Miller, "Saccade abnormalities in autopsy-confirmed frontotemporal lobar degeneration and Alzheimer's disease," *Archives of Neurology*, vol. 69, no. 4, pp. 509–517, 2012.

[76] S. Hopf, M. Liesenfeld, I. Schmidtmann, S. Ashayer, and S. Pitz, "Age dependent normative data of vertical and horizontal reflexive saccades," *PLoS One*, vol. 13, no. 9, p. e0204008, 2018.

[77] G. Fernández, J. Laubrock, P. Mandolesi, O. Colombo, and O. Agamennoni, "Registering eye movements during reading in Alzheimer's disease: Difficulties in predicting upcoming words," *Journal of Clinical and Experimental Neuropsychology*, vol. 36, no. 3, pp. 302–316, 2014.

[78] S. Park, "Representation learning for webcam-based gaze estimation," Ph.D. dissertation, ETH Zurich, 2020.

[79] Y. Xu, Y. Xu, and S. Saria, "A non-parametric bayesian approach for estimating treatment-response curves from sparse time series," in *Proceedings of the 1st Machine Learning for Healthcare Conference*, ser. Proceedings of Machine Learning Research, vol. 56. PMLR, 18–19 Aug 2016, pp. 282–300. [Online]. Available: http://proceedings.mlr.press/v56/Xu16.html

[80] H.-Y. Lai, G. Saavedra-Peña, C. G. Sodini, V. Sze, and T. Heldt, "Measuring saccade latency using smartphone cameras," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 3, pp. 885–897, 2020.

[81] C. Bliss, "The method of probits," *Science*, vol. 79, no. 2037, pp. 38–39, 1934.

[82] M. Álvarez and N. Lawrence, "Computationally efficient convolved multiple output gaussian processes," *Journal of Machine Learning Research*, vol. 12, no. 41, pp. 1459–1500, 2011.