Spring 2014

Short term modification of vergence ramp eye movements in the convergent direction

Chang Yaramothu
New Jersey Institute of Technology

Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

ABSTRACT

SHORT-TERM MODIFICATION OF VERGENCE RAMP EYE MOVEMENTS IN THE CONVERGENT DIRECTION

by
Chang Yaramothu

Prior oculomotor studies have investigated the various effects of short-term modification on vergence, saccadic and smooth pursuit eye movements. Previous vergence studies have concentrated on step modification stimuli. Few have investigated the effects of short-term modification on vergence ramp movements. Thus, this study explores the trends observed within a short-term modification experiment studying smoothly tracking vergence eye movements responses elicited from convergent ramp stimuli. A short-term modification experiment is composed of three phases: baseline, modification and recovery. Baseline and recovery phases contain only test stimuli; whereas, during modification, the subject is presented test and conditioning stimuli in a ratio of 1:5 test to conditioning. The test stimulus is a 0.5 deg/sec vergence ramp presented from a 3 deg vergence angle to a 5 deg vergence angle. The conditioning stimulus is a 2 deg/sec ramp presented over the same visual range. The root mean square error (RMSE) is calculated on all slower (0.5 deg/ sec) ramp responses and compared over the three phases. A significant statistical change is observed between the three stages on day one, but not on day two. A trend that can be attributed to motor memory. This study additionally explores for potential differences between the left and right eye movements. No statistical significant difference of the RMSE is observed between the left and right eye movements. Data supports that the preprogrammed portion of vergence is significantly influenced by the short-term modification experiment described here.

SHORT TERM MODIFICATION OF VERGENCE RAMP EYE MOVEMENTS IN THE CONVERGENT DIRECTION

by
Chang Yaramothu

A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Biomedical Engineering

Department of Biomedical Engineering

May 2014

APPROVAL PAGE

SHORT-TERM MODIFICATION OF VERGENCE RAMP EYE MOVEMENTS IN THE CONVERGENT DIRECTION

Chang Yaramothu

Dr. Tara L. Alvarez, Thesis Advisor Professor of Biomedical Engineering, NJIT	Date
Dr. Bharat Biswal, Committee Member Professor and Chair of Biomedical Engineering, NJIT	Date
Dr. Mesut Sahin, Committee Member Associate Professor of Biomedical Engineering, NJIT	Date

BIOGRAPHICAL SKETCH

Author: Chang Yaramothu

Degree: Master of Science

Date: May 2014

Undergraduate and Graduate Education:

- Master of Science in Biomedical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2014

- Bachelor of Science in Biomedical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2013

Major: Biomedical Engineering

Presentations:

Yaramothu, Chang; Alvarez, Tara L., (2014). Short-term modification of vergence ramp eye movements in the convergent direction. *IEEE Bioengineering Conference*. Boston, MA. (Submitted.)

Enjoy the small things in life.

ACKNOWLEDGMENT

I would like to thank my Thesis Advisor, Dr. Tara Alvarez, for providing me with the knowledge and skills required for me to complete this project. I would also like to acknowledge my other committee members, Dr. Bharat Biswal and Dr. Mesut Sahin for their advisement and guidance in my thesis. Special thanks to Raj, Henry, Steve and other members of the Vision and Neural Engineering Lab, for helping me when needed and making this project fun. Thanks to NJIT for providing me with this wonderful environment to complete my work. Also, I would like to acknowledge my funding source, NSF grant #MRI CBET1228254. Finally, a big thank you to all my friends and family who have always supported me and will continue to do so.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Objective

The primary objective of this thesis is to observe whether short-term modification influences the vergence eye system in the convergent direction and if so, to what extent it may influence. Short–term modification vergence experiments have concentrated on vergence step responses in the past (Alvarez, 2005). However, vergence ramp movements have yet to be studied within a short-term modification experiment. The secondary objective of this thesis is to understand the uniformity or the variation present between the individualized movements of left and right eyes. This section will provide insight into the visual system and its ability to process the visual information.

1.2 Visual Physiology

The visual system senses visible light of 380 nm to 700 nm from the surroundings to create a perception of the world. This visual system utilizes two eyes that work together as one comprehensive system. To maintain a single binocular vision and perceive the three-dimensional world as one entity the visual system utilizes six extraocular muscles, per eye, for highly accurate and precisely controlled movements.

The six basic eye muscles used for the movement of the eyes are the superior rectus, inferior rectus, lateral rectus, medial rectus, superior oblique, and inferior oblique as shown in Figure 1.1 (Marieb, 2010). The superior and inferior recti control the vertical movement of the eyes, whereas the lateral and medial recti guide the horizontal

movements of the eye. Finally, the torsional movements of the eyes are assisted by the superior and inferior obliques. With these six muscles, the eyes can perform the three basic eye movements of vergence, saccadic, and smooth pursuit movements. Vergence is the disconjugate motion of the eyes facilitated by the medial and lateral recti muscles that rotate the eyes within the horizontal plane. Saccadic eye movements are quick movements where both eyes are moving in the same direction. Smooth pursuit is a motor system which stabilizes the retina to follow a target in all directions in smooth motion.

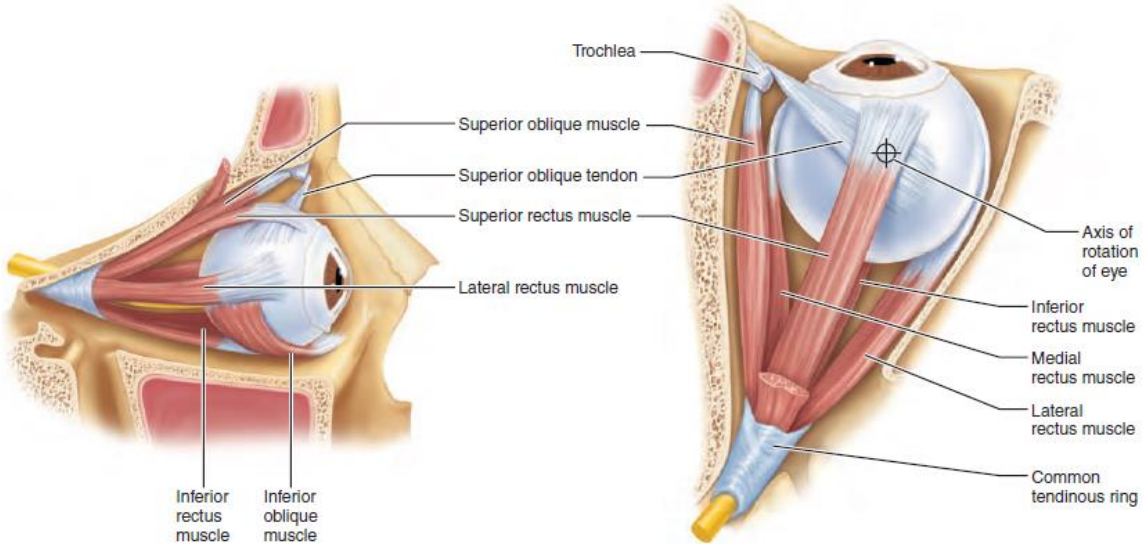

Figure 1.1 Schematic drawing of the eye with extraocular muscles. (Marieb, 2010)

The optical top layer of the eye is composed of three layers, the sclera (outer most layer), the choroid (middle layer), and the retina. The shape of a transparent lens within the eye is constantly adjusted to help focus light into the macula. The macula lutea is a region in the back of the eye that has specialized structures for high acuity vision. Located in the center of the macula is the fovea centralis.

The fovea is the region responsible for the sharp central vision. This sharp vision emanates from the high density of cones located in the fovea. The macula contains a mix of cones and rods, with a greater concentration of cones. The density gradually decreases

form the fovea to the retina periphery region (Marieb, 2010). The cones and rods aid in the processing of images perceived by both the individual eyes. The retinal images of the two eyes are essentially two-dimensional. Rods process vision in dim light, where the cones process color vision. There are three basic types of cones: red cones, blue cones, and green cones, which process those respective colors. This processed visual data are then transmitted to the cerebral cortex through the optic nerve.

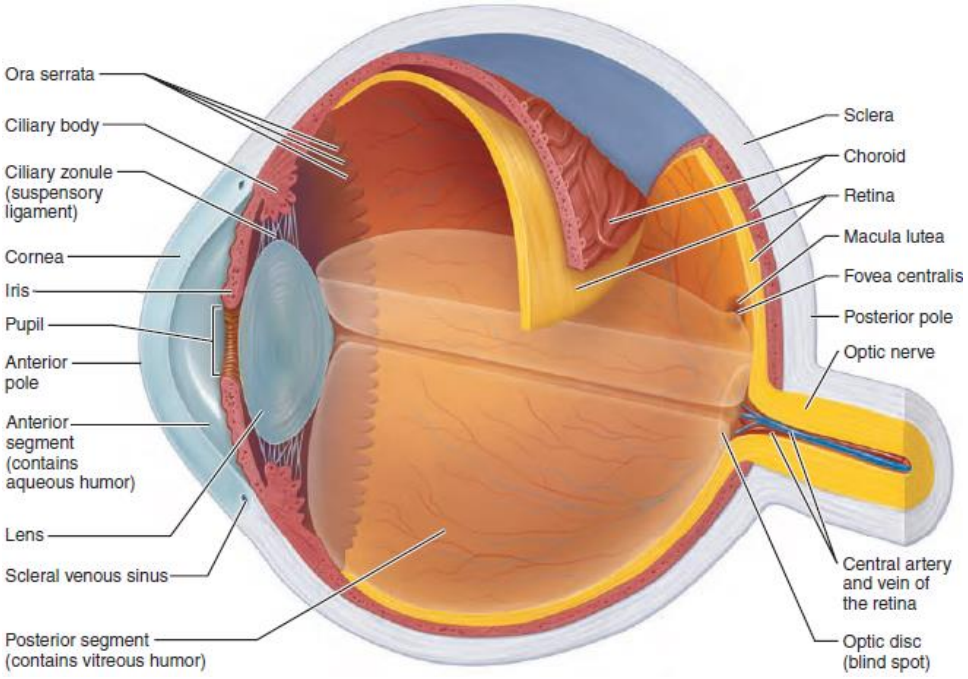

Figure 1.2 Internal structure of the eye. (Marieb, 2010)

The optic nerves from each of the eyes cross at the optic chiasm in the hypothalamus, causing the right side of the primary visual cortex to be responsible for the left half of the visual field and the left side to be responsible for the right visual field as shown in Figure 1.3. The information from the nerves are combined at the optic chiasm and divided according to their respective visual fields. The retinas are divided into quadrants (virtually divided by a vertical and horizontal line that would intersect at the center of the fovea) to create their visual fields. In addition to the previously stated visual

fields, two additional fields of binocular and monocular are present. The binocular visual view is divided into the left and right binocular hemifields. The left binocular hemifield includes the nasal visual field of the right eye and the temporal visual field of the left eye and its respective opposite on the right binocular hemifield. The peripheral field of view is monocular as seen in Figure 1.3.

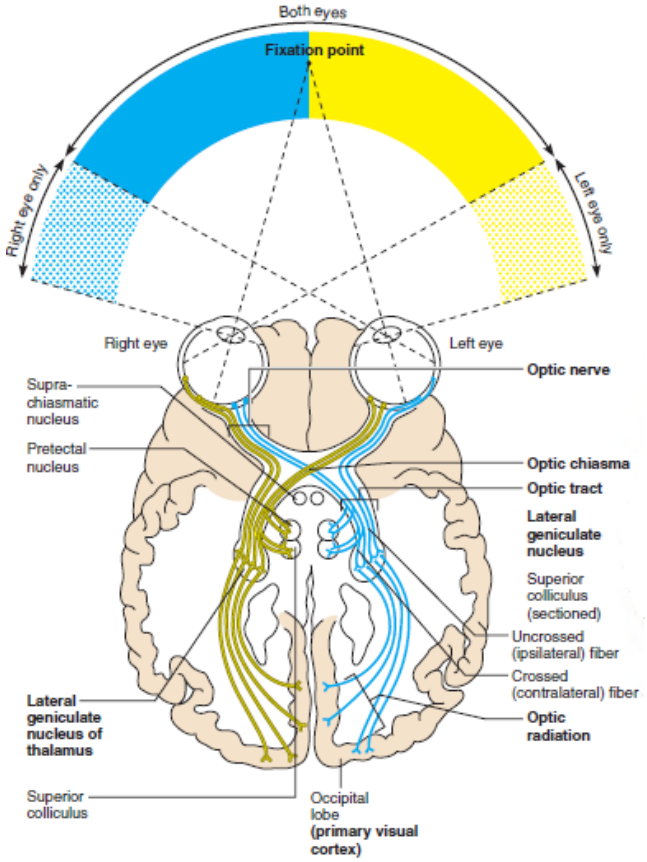

Figure 1.3 Visual pathway to the primary visual cortex. (Marieb, 2010)

Axons from the optic chiasm that bind the cerebral peduncles of the midbrain are called the optic tract. These axons extend to the lateral geniculate nucleus (LGN). The majority of the optic nerve axons project onto the lateral geniculate nucleus of the thalamus, which are later projected to the primary visual cortex (Brodmann's area 17/V1) and extrastriate areas (B18/V2 and B19/V3), all are of the occipital lobe, via a process

called optic radiation.

1.3 Vergence System

Vergence eye movements are the disconjugate motion of the eyes facilitated by the medial and lateral recti muscles that rotate the eyes within the horizontal plane. There are two types of movements in the vergence system: convergence and divergence. Convergence is the inward rotation of the eyes which gives the eyes the ability to track from far to near as depicted in Figure 1.4. The Dual-Mode theory models convergence using two main components: a transient preprogrammed component that controls the speed of the movement and a sustained feedback controlled component which provides the accuracy of the final movement (Yuan, 1999). Behavioral and neurophysiology studies supports that vergence has two distinct control systems (Semmlow, 1986; Alvarez, 1998). Prior research has shown the preprogrammed component is modified. Specifically, the transient gain of the system can be increased or decreased for both convergence and divergence (Munoz, 1999; Semmlow, 2002; Alvarez, 2005). However, it is unknown whether the feedback controlled system will substantially change within a short-term modification experiment.

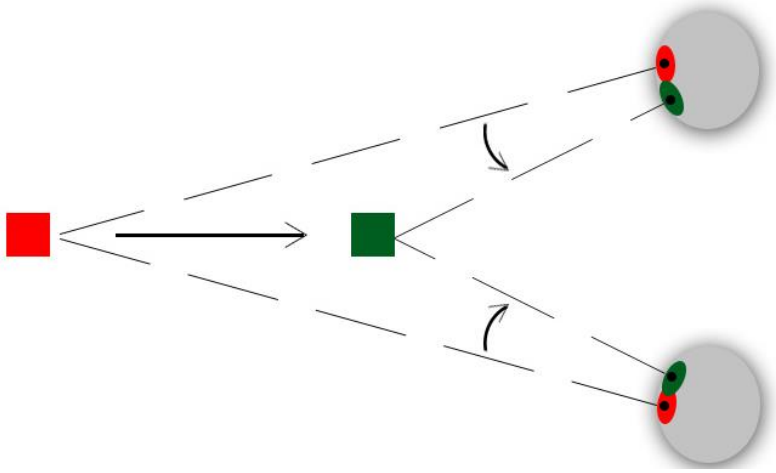

Figure 1.4 Convergence eye movement.

1.3.1 Disparity and Accommodative Vergence

The near triad is the relationship between vergence, accommodation, and pupillary constriction to a near response. When looking at a near object, multiple systems are activated such as the constriction of the pupils, the lens becoming more spherical, and the either a convergence or divergence movement. Disparity is the difference between the image location of an object seen by the left and right eyes, created by the separation between the eyes. Disparity-vergence is the response to disparity cues whereas accommodative-vergence is the response to blur.

Initially, Maddox hypothesized that the accommodative stimulus was the main contributor to the vergence system (Maddox, 1983), but recent studies suggest that disparity-vergence may be a more important contributor of the vergence system (Judge, 1991; Horwood 2009). Studies have shown that the disparity vergence response occurs about 100 msec before the accommodative response (Hung et al 1986; Lee, 2009). The authors reported that the vergence peak velocity was not significantly different in both methods.

1.3.2 Factors Influencing the Disparity-Vergence System

Fixation disparity is one of the factors that influences the precision of the horizontal vergence system. Fixation disparity is the vergence error between the fixation point and target plane. This fixation point may be either in front of or behind the target plane (Janchinski, 2010). Factors such as prediction (Alvarez, 2005), adaptation (Alvarez, 1999), age (Rambold, 2006), fatigue (Lee, 2009), and phoria adaptations (Patel, 1999) also influences vergence peak velocity and latency.

1.4 Related Studies

Past studies on vergence step movements in the convergent direction and smooth pursuit can be used as a basis for understanding vergence ramp movements. Short term modifications in the convergent directions result in significantly greater peak velocities after going through modification in vergence steps (Munoz, 1999). This trend can be translated into the vergence ramp movement as an increase in the ability for the eye to follow the targets at a faster rate. The same trends have also been observed in smooth pursuit movements. Short and long term training has resulted in a higher peak velocity (Lisberger, 1978). All of these previous studies concentrate on the preprogrammed portion of vergence or the transient portion of smooth pursuit which is subject to greater error. Previous studies have not studied the effects of fast or slower movements over longer period of time of 5 seconds. This longer stimulation will test not only the preprogrammed, but also feedback system.

CHAPTER 2

METHODOLOGY

This chapter will discuss the different methodologies of subjects, apparatus, calibration, and statistical analyses.

2.1 Subjects

A total of ten subjects participated in this study, seven males and three females. Two additional non-naïve subjects (one male and one female) participated within a pilot study to test the system and to help determine the optimal conditions. The naïve subjects have never received any form of oculomotor training and were naïve to the short term modification effects. The subjects were between 18 to 25 years of age with a mean age of 20.7±2.4 years and all had normal binocular vision assessed by the Randot Stereopsis Test and near point of convergence test (NPC). Stereopsis was assessed for all ten subjects with the Bernell Stereo Randot test using the Randot circles (Bernell, South Bend, IN). Normal is defined as better than 70 second arc and all subjects tested below 50 seconds (Alvarez, 2010).

NPC was measured by having the operator slowly bring a reduced Snellen 8cm x 8cm card toward the subjects along their midline. The subjects were instructed to concentrate on the 40 line and to keep the letters single and clear. When the letters became blurry, the operator would record the distance of the card and repeat when the letters are doubled. The distance was also recorded when there was a recovery single vision as the card was slowly moved away from the subject. Finally, the subjects were tested for their motor and sensory dominance as shown in Table 2.1. All subjects signed

informed consent before the experiments, approved by the New Jersey Institute of Technology Institutional Review Board (IRB).

Table 2.1 Subject Motor/Sensory Dominance and NPC Clinical Measurements

Subject	Motor Dominance	Sensory Dominance	Near Point Convergence		
			Blur	Break	Recovery
1	Left	Left	8	3	5
2	Left	Left	11	5	6
3	Right	Right	10	3	5
4	Left	Left	10	4	10
5	Left	Left	9	3	5
6	Right	Right	9	2	4
7	Right	Right	6	4	5
8	Right	Left	7	5	7
9	Right	Right	7	2	4
10	Right	Right	9	8	10

2.2 Apparatus

The stimuli were presented on a haploscope which stimulates disparity vergence while keeping accommodative vergence relatively constant. Two separate screens, one for each eye, 40 cm away from the subject's line of sight were used to present the stimuli. Two partially reflecting mirrors were positioned in front of the subject, which projected the stimuli from the monitors to the subject's line of sight. Eye movements were recorded with an ISCAN Eye Tracking Camera System (model ETL 400, ISCAN Inc., Burlington, MA). This system utilizes an infrared (λ = 950 nm) video based system. The manufacturer specifies that the accuracy for this system is 0.3° over a ±20° horizontal range. The two cameras were placed in front of subject, one in front of the left-eye and the other in front of the right-eye at a distance of 15 cm. Individual eye movements were quantified using the centroid of the pupil movements. The eye movement monitor power level is 1.2 mW/cm^2, which is well below the OSHA safety limits of 10mW/cm^2.

The subjects were placed in a customized space and covered by commercial blackout curtains (Blackout Curtains, St. Louis Park, MN, USA) to minimize the amount of light emitted into the experimentation space. There are no visible light sources inside the experimentation space and this space is separated from the monitors used by the operator. Before the start of each experiment, the subjects verbally confirmed that they did not perceive any light source other than the stimuli. The experimentation space was equipped with a head and chin rest where the subjects were secured and restrained with an elastic band to reduce the influence from the vestibular system (Khojasteh, 2007). The subjects were also asked to restrain from any head movement.

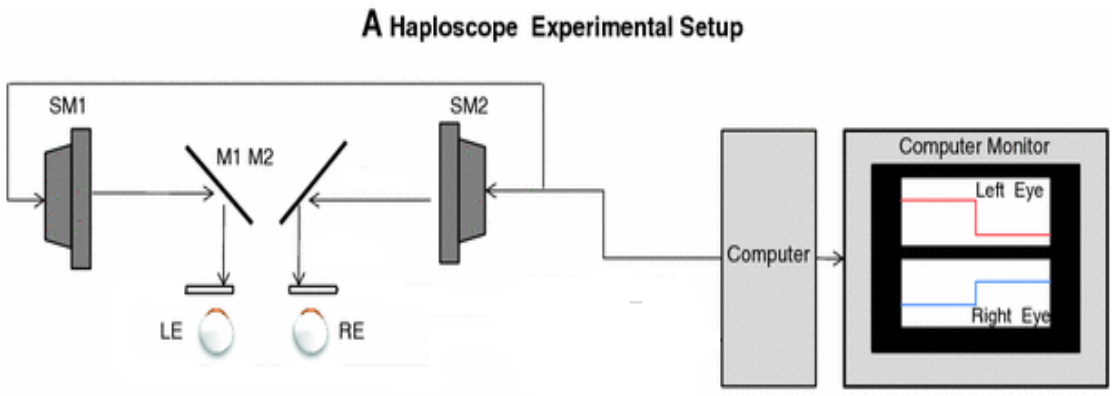

Figure 2.1 A haploscope utilizes two partially reflecting mirrors to project the stimuli from the two monitors. A computer controls the stimuli on the two monitors and saves the data obtained from the two cameras, one positioned in front of each eye, for offline analysis. (Kim, 2011)

The entire system is controlled by a custom LabVIEW™ 8.0 Virtual Instrument (National Instrument, Austin, TX, USA) which generated the stimuli and digitized the eye signals at 500 Hz (Guo, 2011). The signals were digitized using a 12-bit digital acquisition (DAQ) hardware card using the range of ±5 Volts (National Instruments 6,024 E series, Austin, TX, USA). The left and right eye movements were saved for

offline data analysis using a custom MATLAB version R2012B code (Waltham, MA, USA).

2.3 Calibration

Prior to each experimental session, the stimuli from both of the monitors were calibrated with real targets placed at measured distances from the subject's midline. This calibration was later used to correlate monitor pixel value with angular positions of the eye rotation in degrees. This procedure was performed before each experimental session to produce unique stimulations for each subject; eliminating the need to approximate the interpupillary distance and any other vergence angular position calculations.

2.4 Stimulation

All subjects participated in two identical experiments recorded on two different days at approximately the same time of the day. The calibrations obtained before each of the stimulations was six monocular positions located at three, five, and seven degree vergence angles. These calibrations were eventually not used in the analysis however served as a basis for the pixel to angular position correlation. There were three stages to the experiment: baseline, modification, and recovery. The baseline and recovery phases were identical where the subject received only test stimuli. The baseline stimulation is a series of ten stimuli that start from a 3 degrees vergence angle and end at 5 degrees, smoothly moving at the rate of 0.5 deg/sec (Hung 1991). The calibration and test stimulation is repeated once more for a total of 20 stimuli in the baseline stimulation.

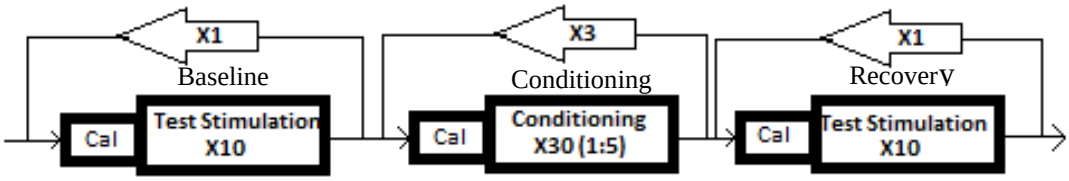

Figure 2.2 Experimental protocol for the three stages of stimulation (baseline, modification, and recovery). The test stimulation of 10 stimuli are repeated once for a total of 20 stimuli, the conditioning stimulation of 20 stimuli is repeated an additional 30 times, and the first phase is once again repeated.

The modification portion of the experiment is a series of slow and fast ramp stimulations presented at the ratio of 1:5. The fast ramp stimulations also begin from a 3 degree vergence angle and end at a 5 degree vergence angle, but this stimulus moves much faster at a rate of 2 deg/sec schematically presented in Figure 2.3. The subject is exposed to a total of 120 conditioning stimulations. The subject is finally displayed the test stimulation of 20 stimuli once again to record the eye movements in the recovery phase. The visual stimuli are schematic presented in Figure 2.2.

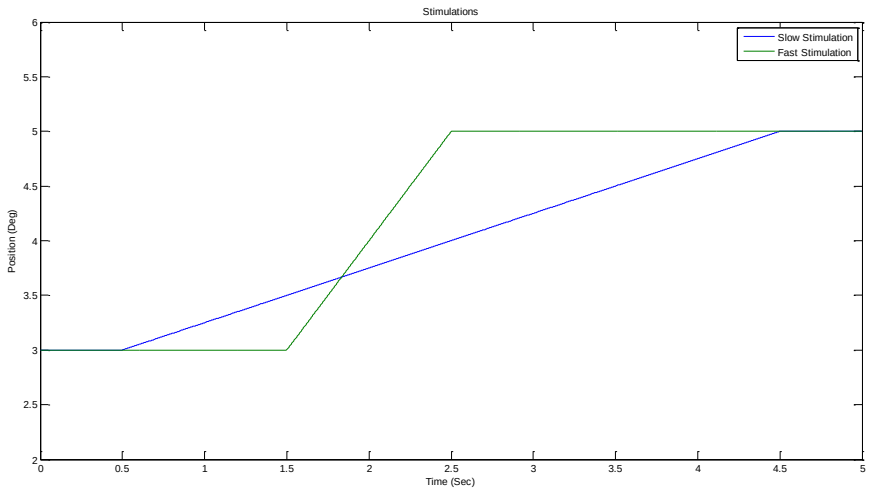

Figure 2.3 Experimental protocol for the slow (blue) and the fast (green) stimulation. Slow stimulation starts at three degrees at 0.5 seconds and incrementally moves to five degrees until 4.5 seconds with a velocity of 0.5 deg/sec. Fast stimulation also begins at three degrees at 1.5 seconds and rapidly moves to five degrees at a rate of 2 deg/sec until 2.5 seconds of the stimulation.

2.5 Root Mean Square Error

The root mean square error (RMSE) is a measure used to find the difference between the predicted and actual values. The RMSE takes the difference between a predicted and an observed value and these individual residuals are measured over the entire data sample as seen in Equation 2.1.

$$RMSE = \frac{\sqrt{\sum(X_{Observed} - X_{Predicted})^2}}{n} \qquad (2.1)$$

The $X_{Observed}$ in Equation 2.1 is the observed value, $X_{Predicted}$ is the value of the stimulation at the given time point, and the n is the total number of samples. Since the total number of samples of this investigation is going to 2500 for all data sets, the n value is going to be ignored for the calculations and Equation 2.2 will be used.

$$RMSE = \sqrt{\sum(X_{Observed} - X_{Predicted})^2} \qquad (2.2)$$

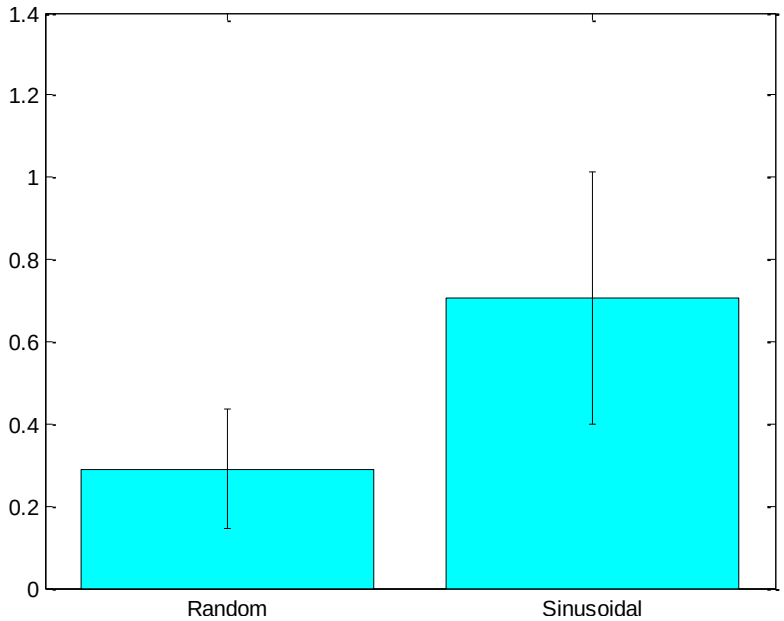

Figure 2.4 Root mean square error of the slow stimulations altered by a random set of numbers and by a sinusoidal, compared with the original slow stimulation.

To test the effectiveness of the RMSE MatLab® function, the slow stimulation has been compared with a slow stimulation that has been altered by a set of random numbers generated by the rand MatLab® function and another slow stimulation that has been altered by a sinusoidal. As seen in Figure 2.4, the sinusoidal function produced a greater root mean square error due the increased fluctuations caused by the sinusoidal distortion compared to the random distortion.

2.6 Data Analysis

The data were analyzed offline using a custom MATLAB version R2012B code (Waltham, MA, USA). The data were initially normalized to begin at an initial vergence angle of 3 degrees and end at a 5 degree vergence angle. The normalization points were chosen by taking the mean of the first and last 200 data points which equates to 0.4 seconds. The convergence direction is assumed to be the positive direction of all analysis. The individual normalized eye movements are plotted along with the disconjugate eye movement to determine which of the trials will be used for further analysis. Any disconjugate movements which have saccades or eye blinks were immediately removed. Additionally, any disconjugate trials which had movements that deviated by more than 1 degree from the stimulus were eliminated. Individual eye movements which effected the disconjugate movement by more than 1 degree were also eliminated.

The individual movements were than filtered using a sixth degree low pass butter worth filter (0-10 Hz bandwidth). Means and standard deviations were taken of the remaining trials for each eye movement data sets collected during the baseline, conditioning, and recovery phase for each subject. The standard deviation and means were calculated by using the basic “mean” and “std” MATLAB® functions. Mean

velocity traces were generated by computing the average eye velocity for each point in time using a two point central difference algorithm. Root mean square errors were also calculated as described in Section 2.5.

CHAPTER 3

RESULTS

In Chapter 3, the results of the stimulation from Section 2.4 are presented. The position traces, mean disconjugate movements, velocities, left and right eye movements are plotted in the sections below.

3.1 Position Traces of Movements

The first set of figures contains the position traces of all the movements that followed the target. All traces that had saccades or blinks were removed which would act as confounding variables to the RSME calculation. The mean of those movements are also drawn on those figures using a bold red line.

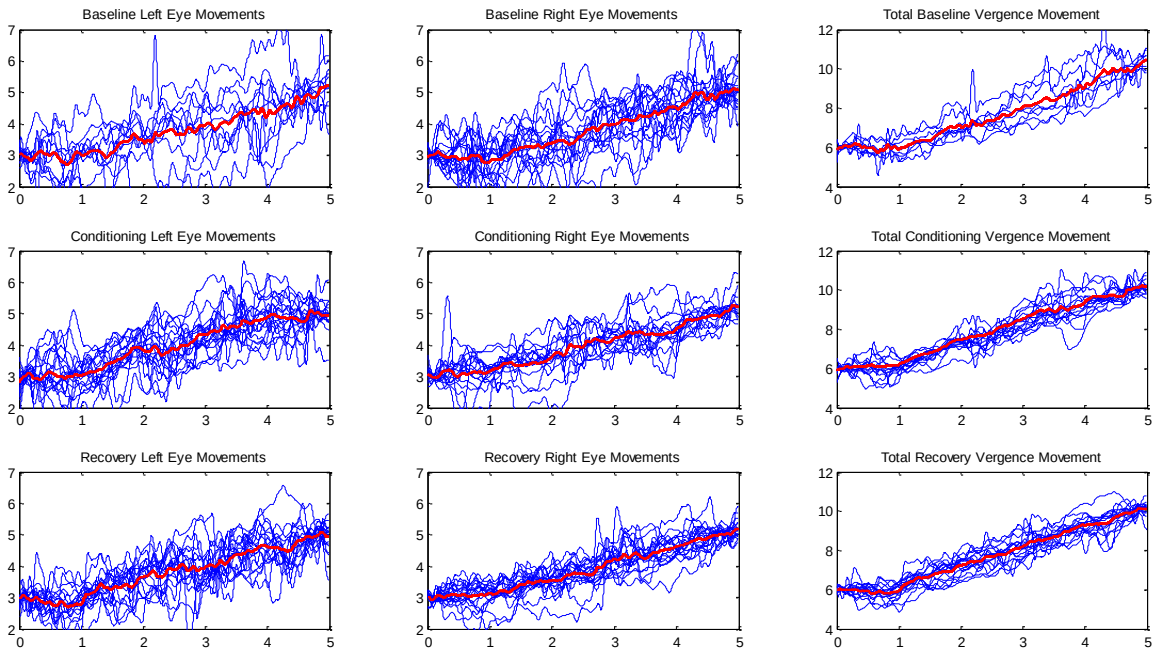

Figure 3.1 Subject 1 position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

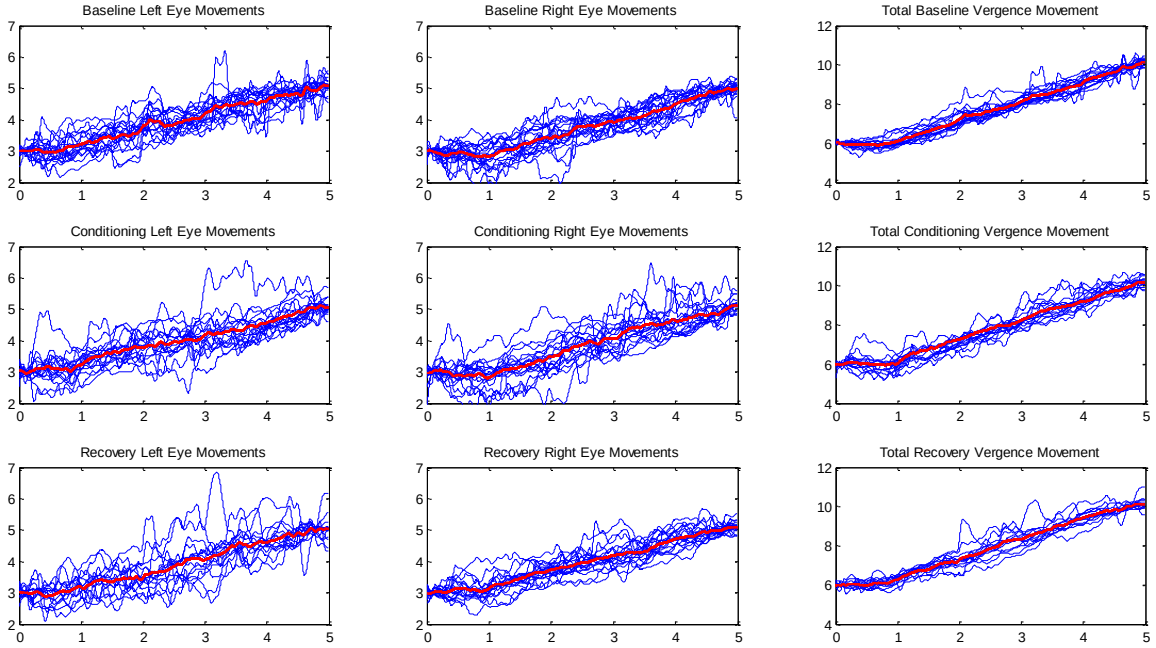

Figure 3.2 Subject 1 Position traces from Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

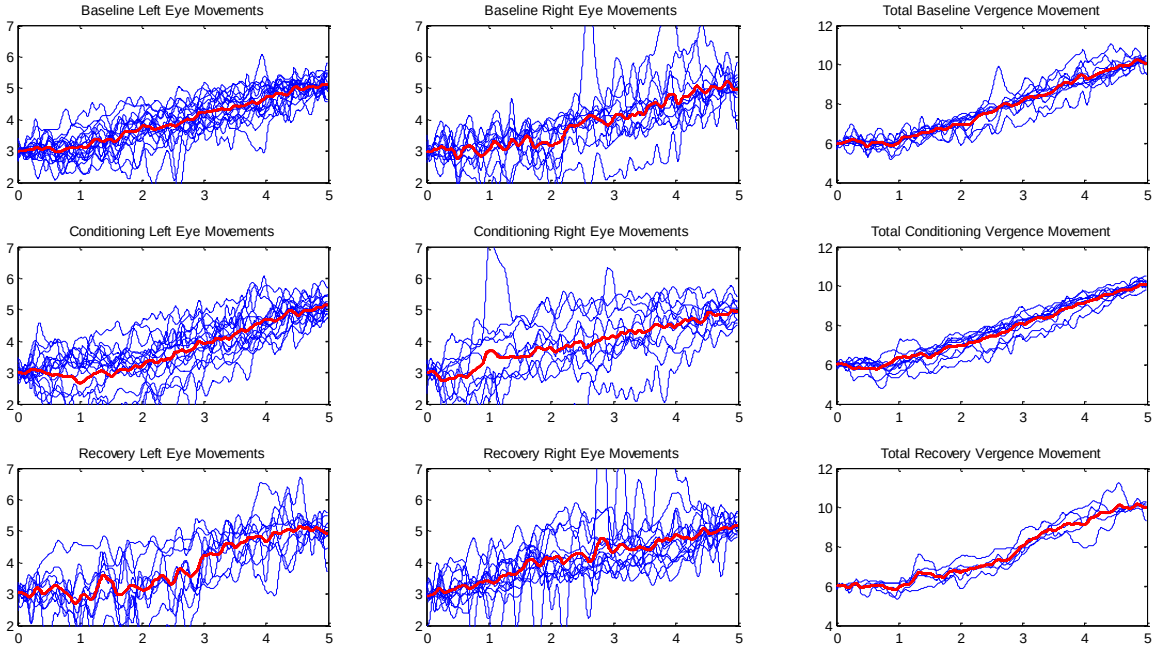

Figure 3.3 Subject 2 position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

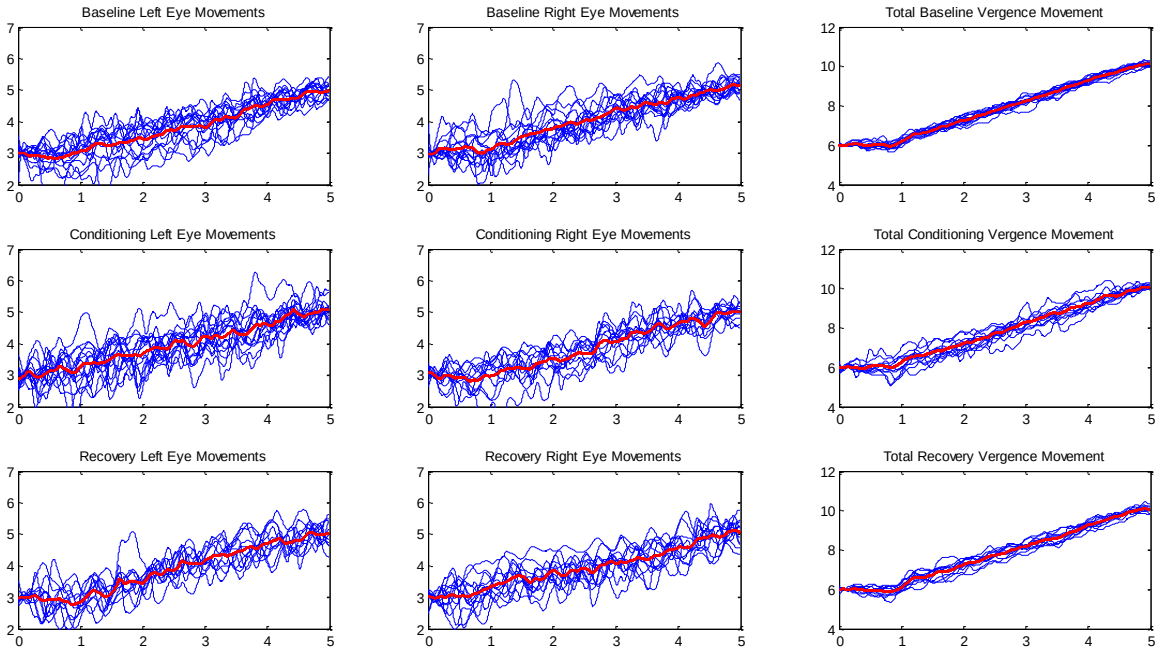

Figure 3.4 Subject 2 position traces from Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

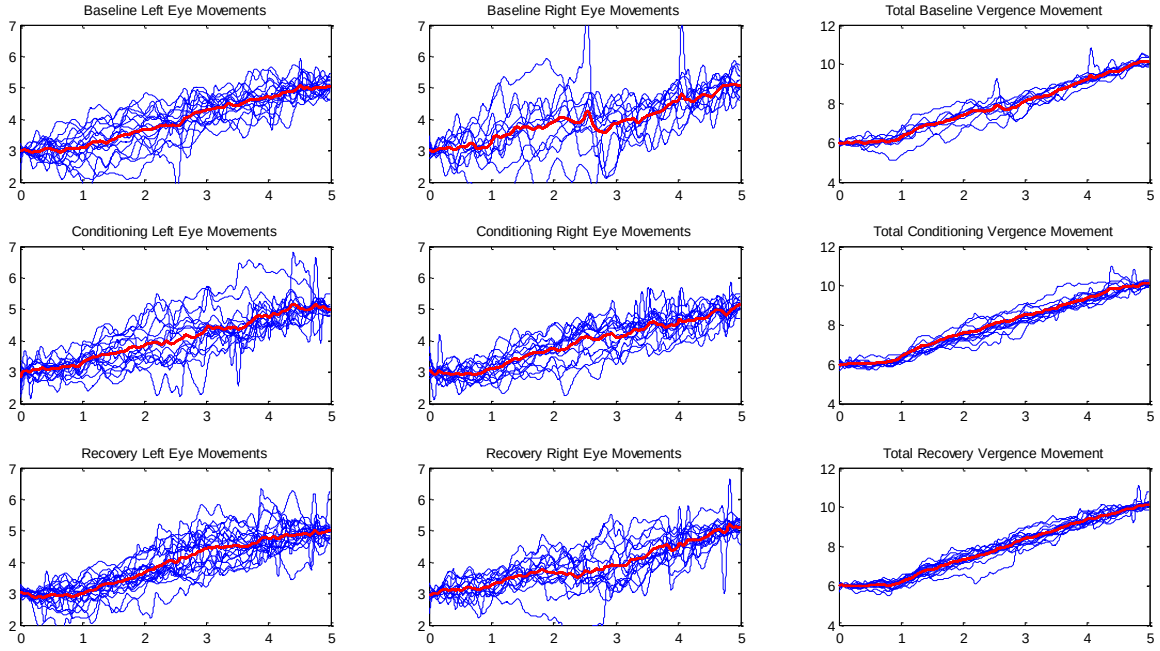

Figure 3.5 Subject 3 Position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

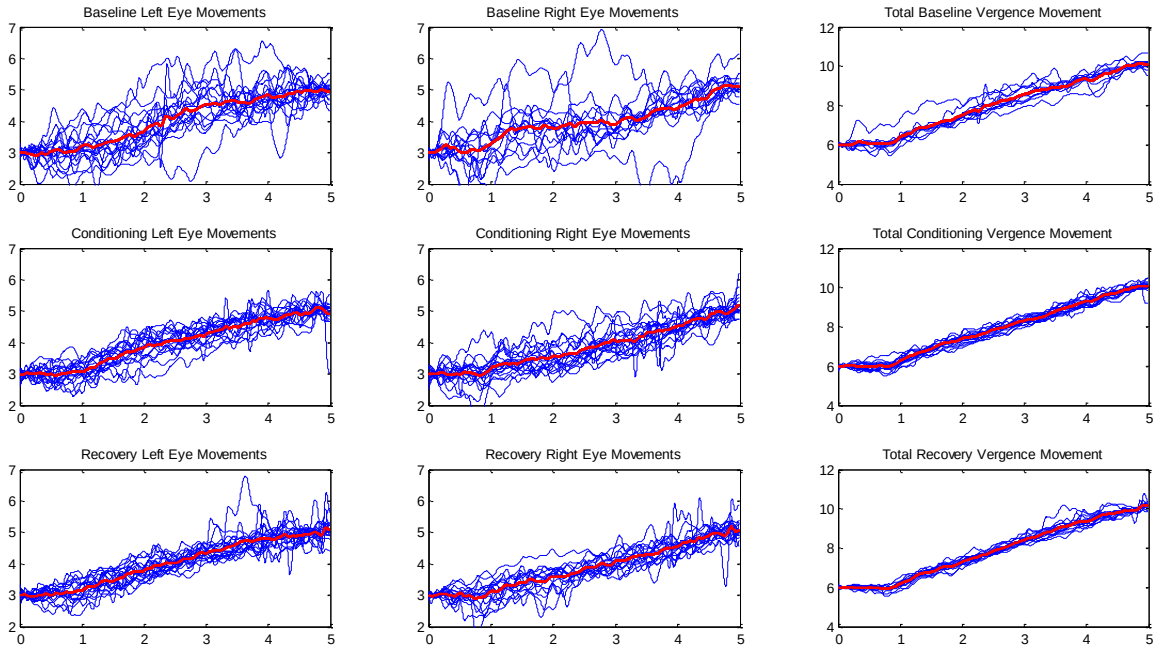

Figure 3.6 Subject 3 position traces from Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

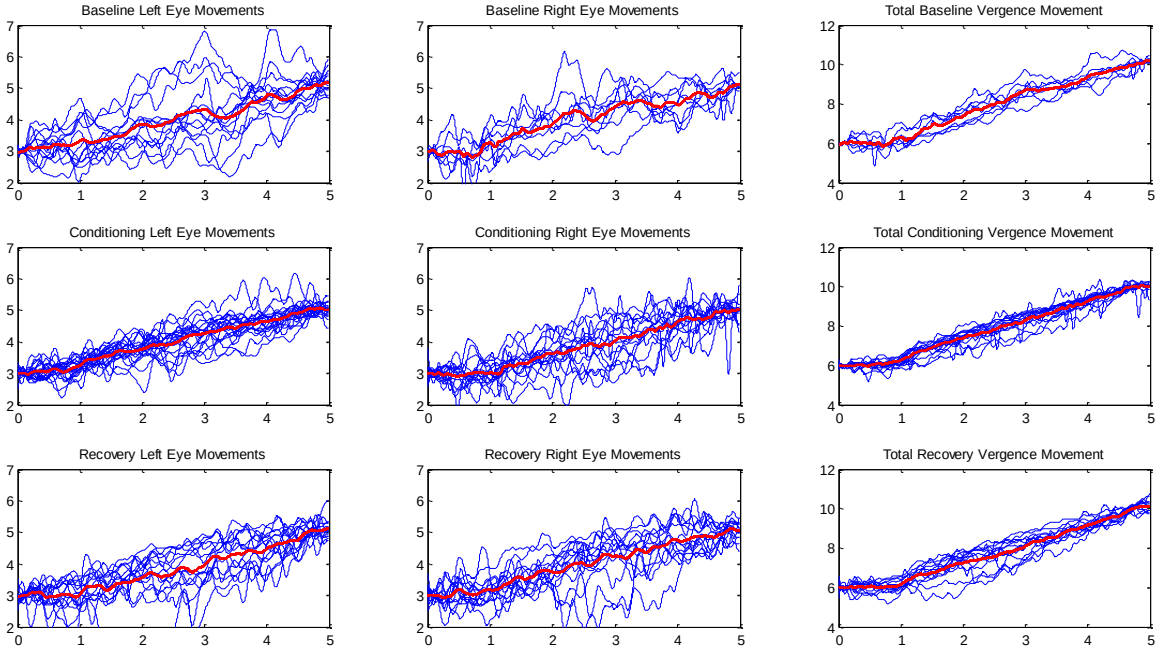

Figure 3.7 Subject 4 position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

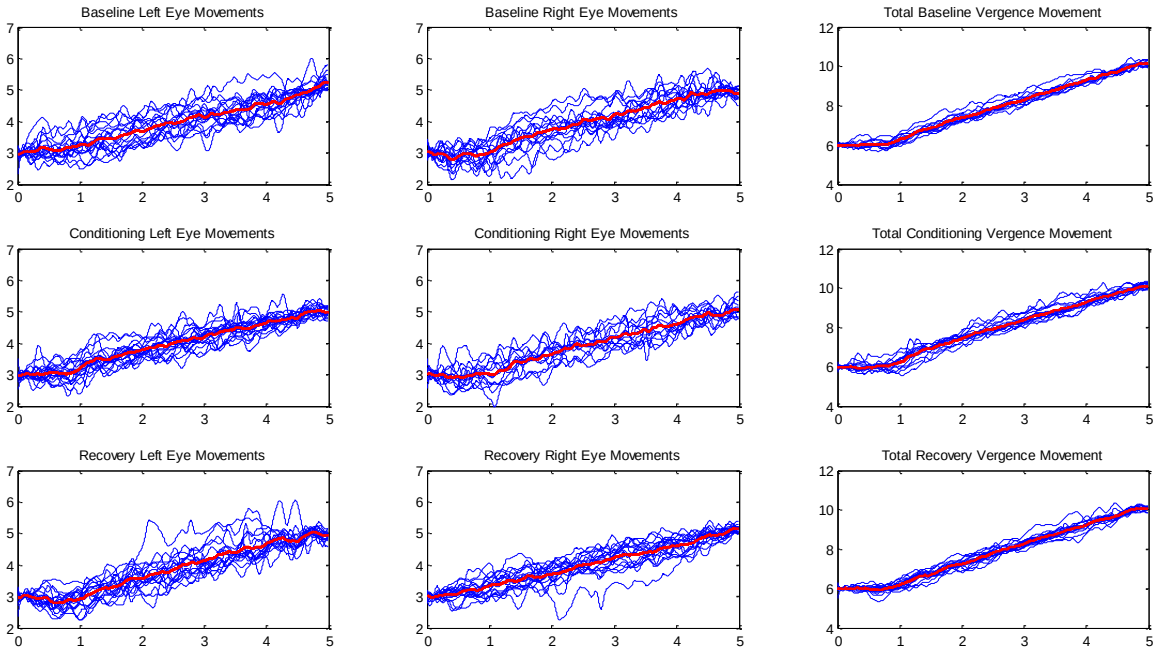

Figure 3.8 Subject 4 position traces from Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

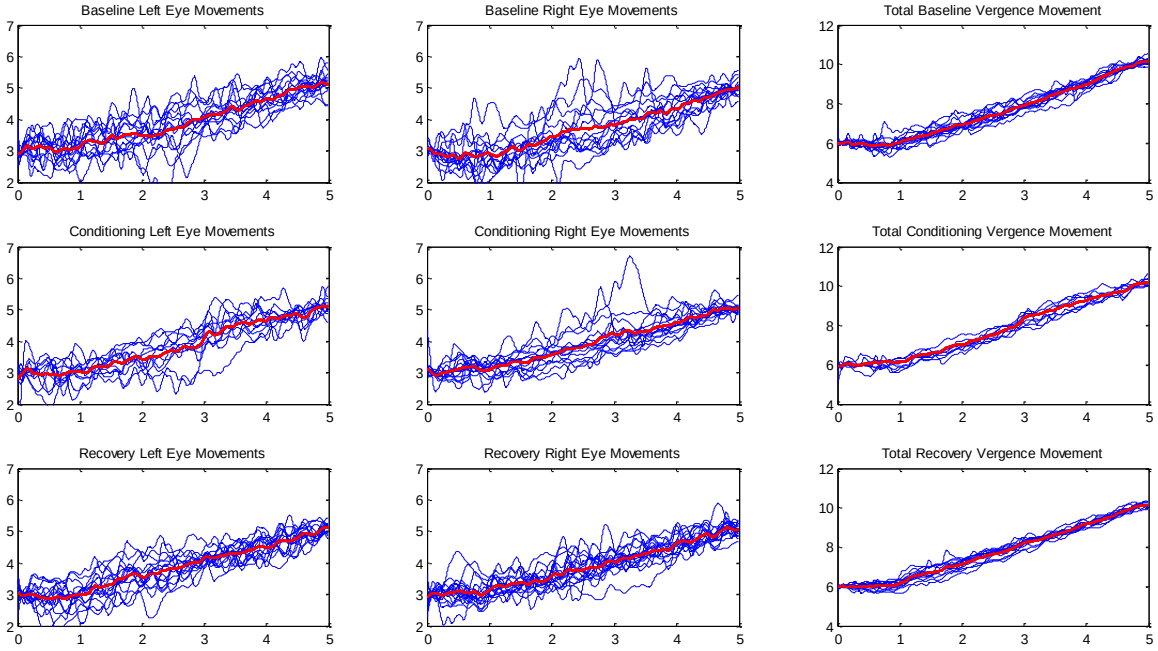

Figure 3.9 Subject 5 position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

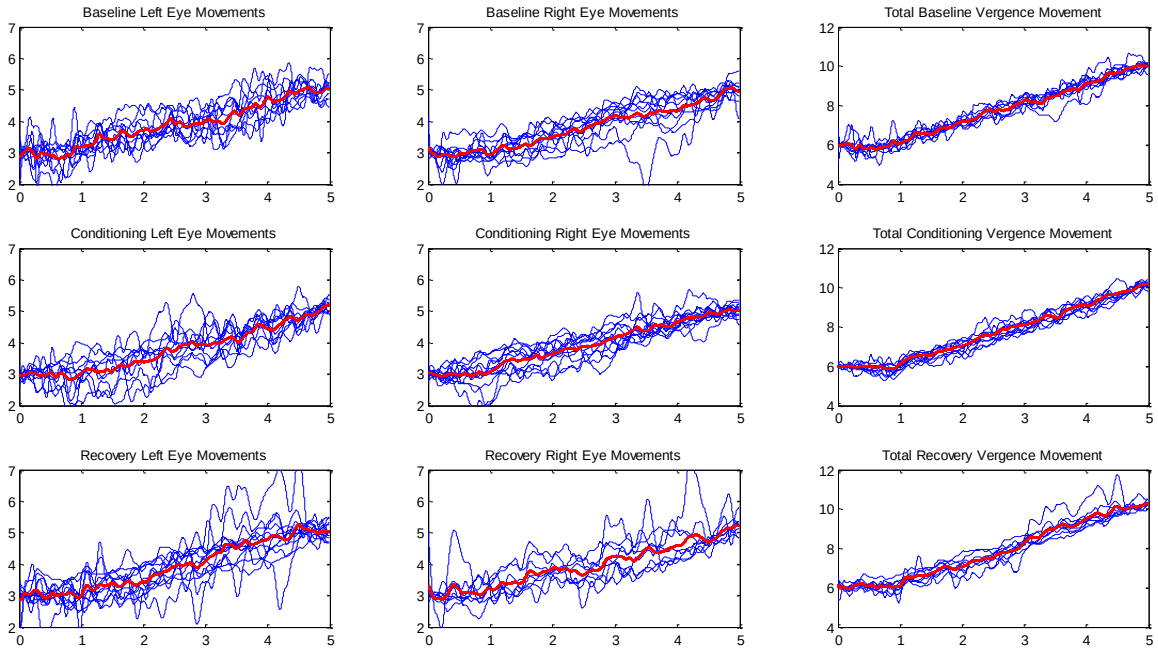

Figure 3.10 Subject 5 position traces from Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

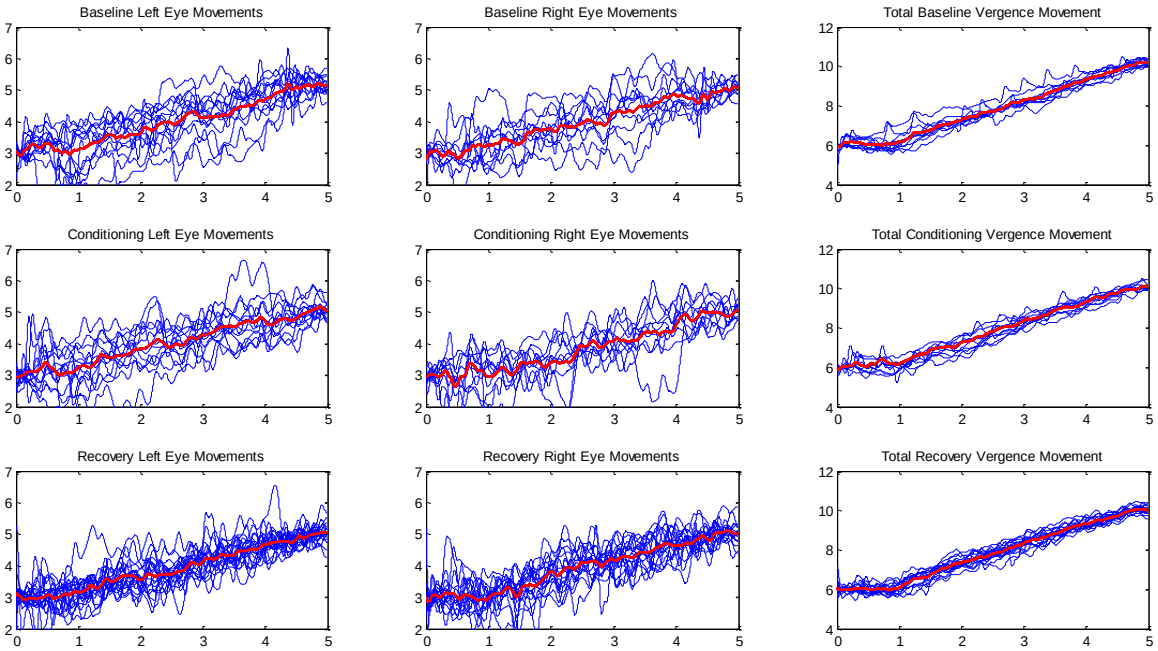

Figure 3.11 Subject 6 position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

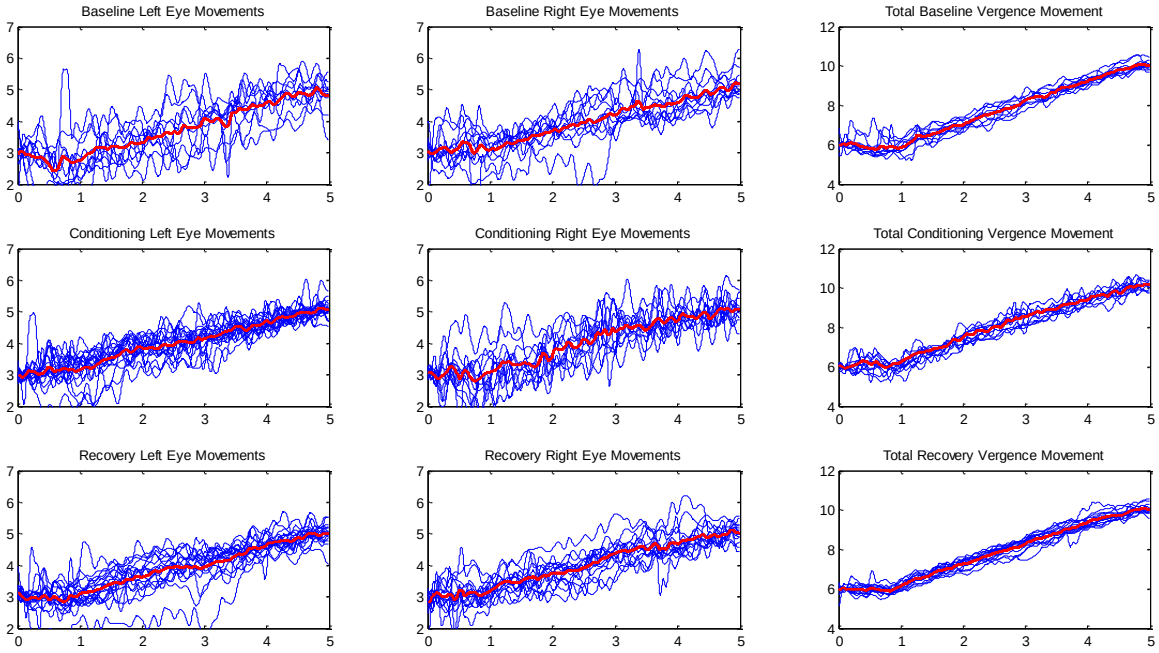

Figure 3.12 Subject 6 position traces from Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

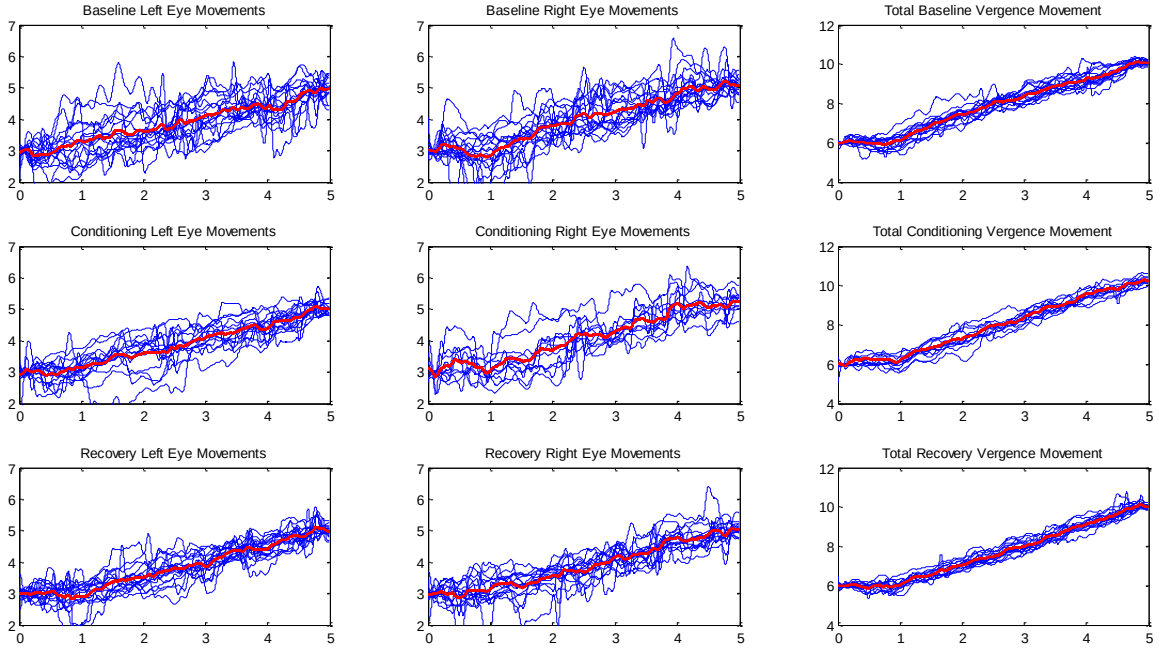

Figure 3.13 Subject 7 position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

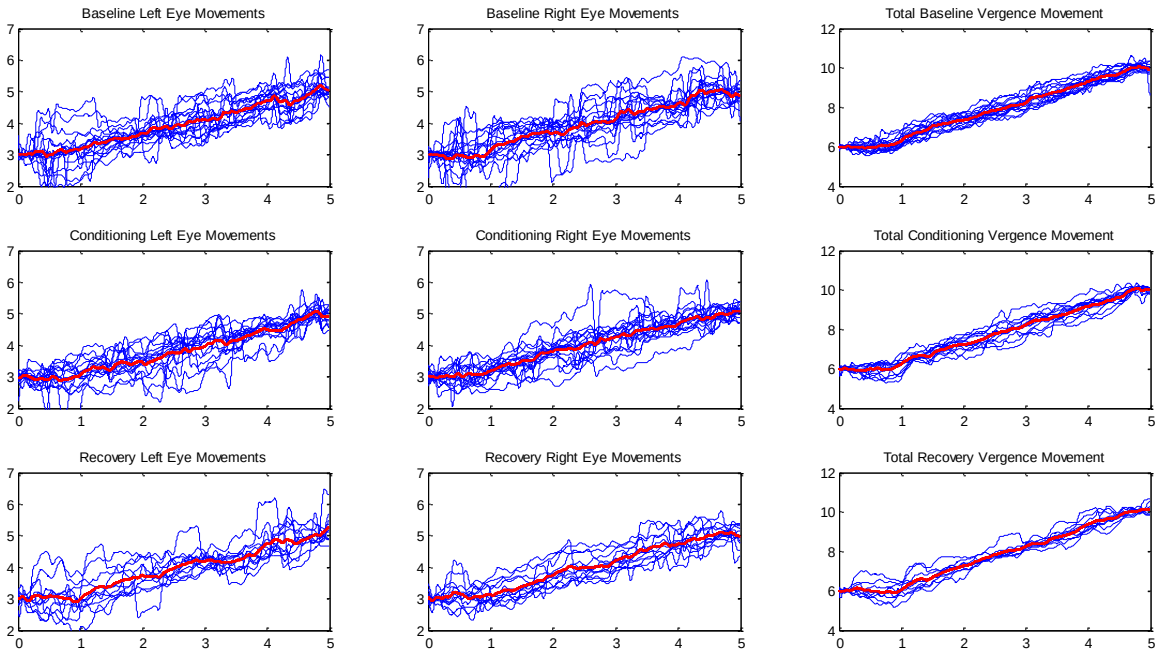

Figure 3.14 Subject 7 position traces from Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

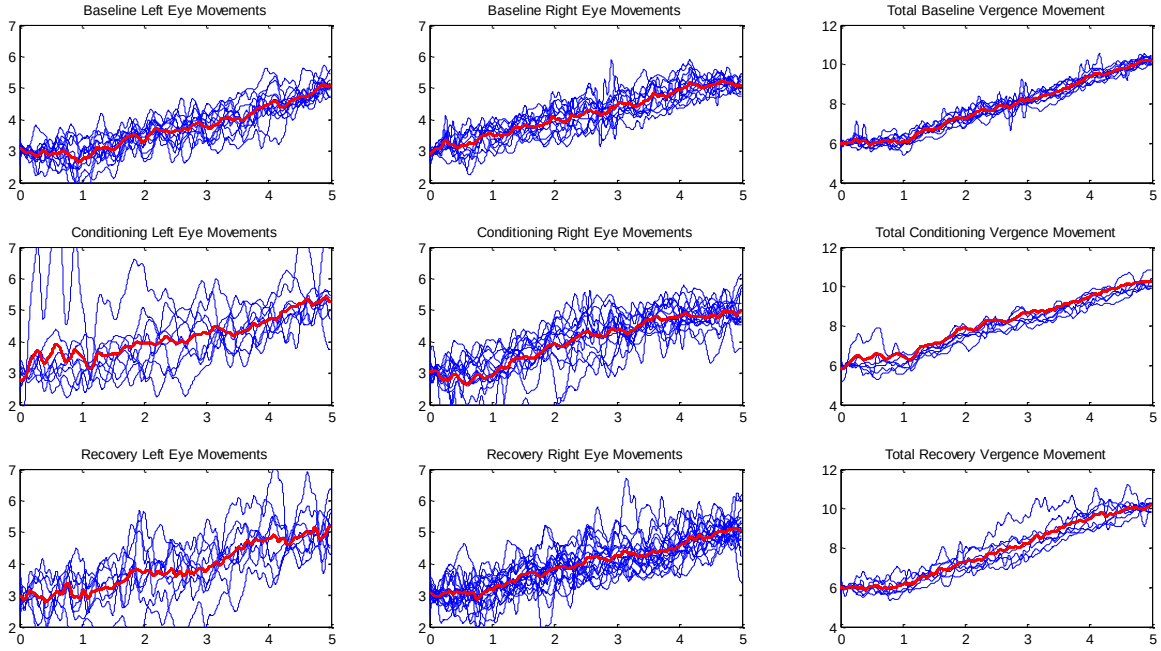

Figure 3.15 Subject 8 position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

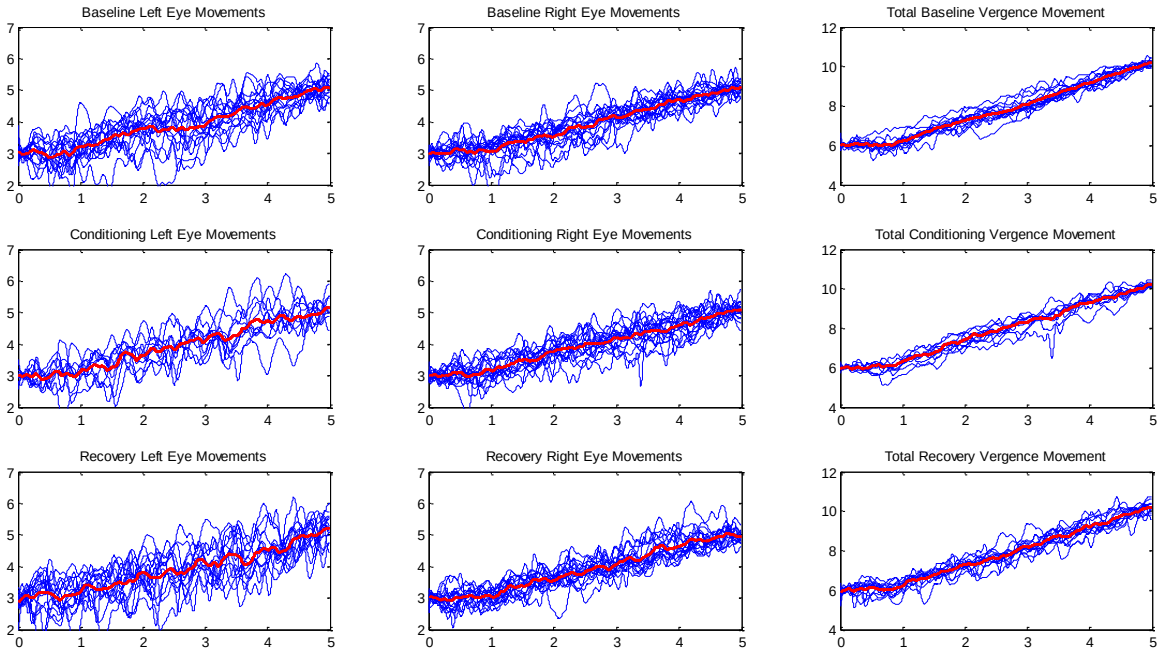

Figure 3.16 Subject 8 position traces Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

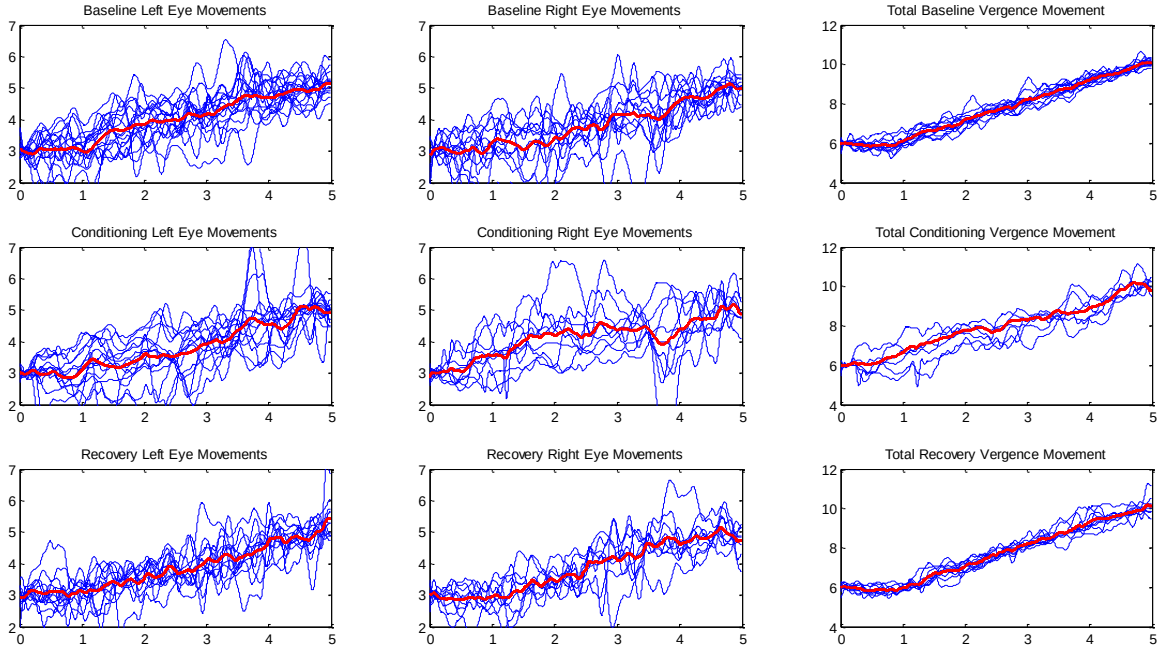

Figure 3.17 Subject 9 position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

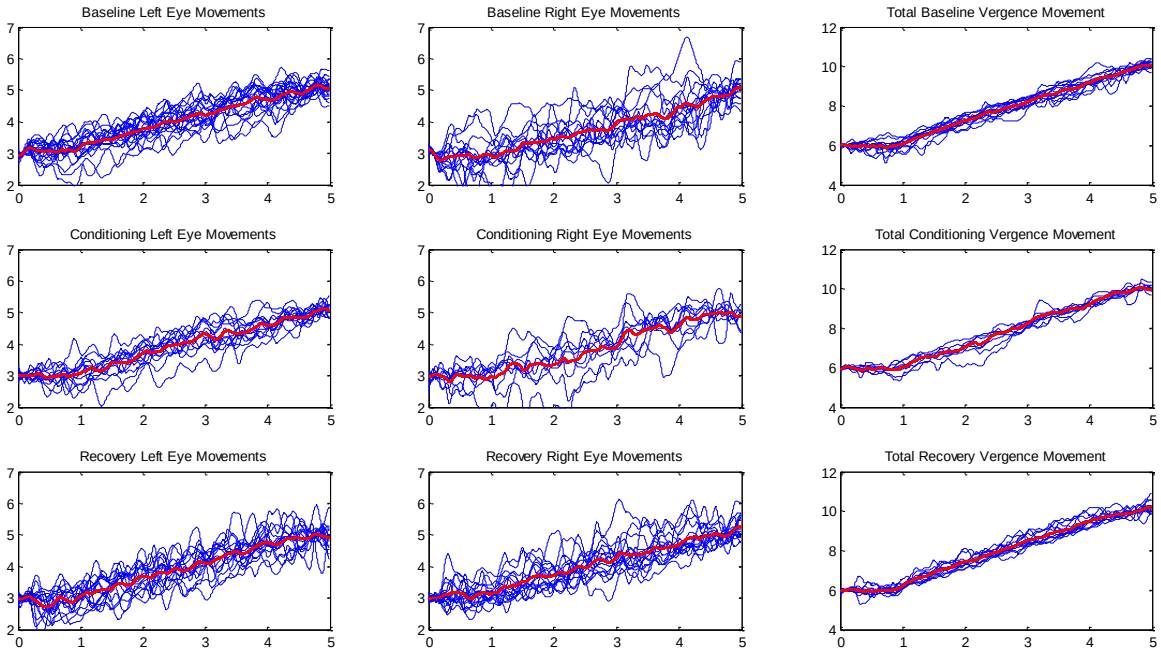

Figure 3.18 Subject 9 position traces from Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

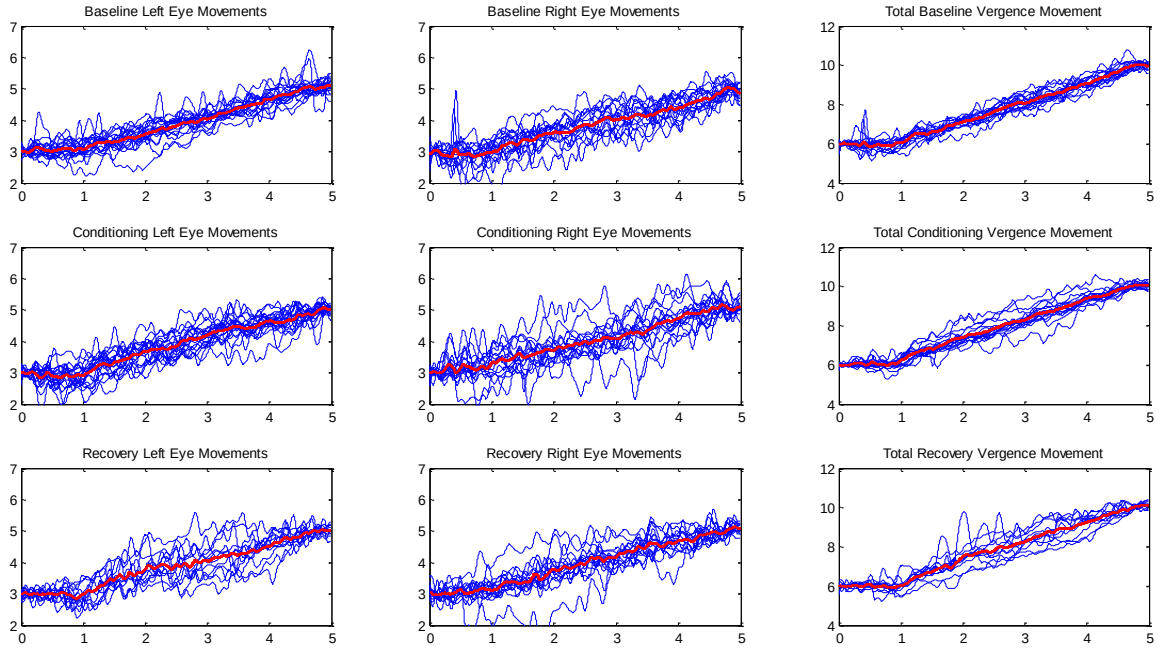

Figure 3.19 Subject 10 position traces from Day 1. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

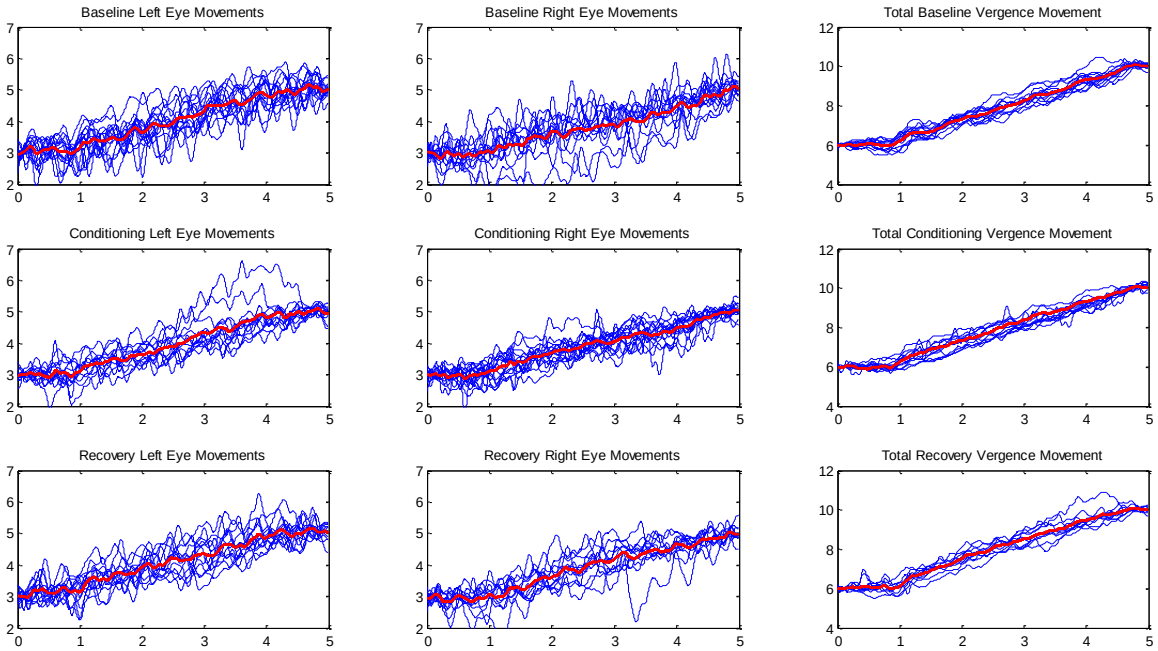

Figure 3.20 Subject 10 position traces from Day 2. Left column of figures contain the left eye movements, central column contain the right eye movements, and the right column contains the disconjugate eye movements.

3.2 Mean Eye Movements

Subsection 3.2.1 contains the mean movements of all 10 subjects in terms of position versus time (top row) and velocity versus time (bottom row). The three disconjugate means of baseline (blue), Conditioning (red), Recovery (Cyan) are depicted with their first standard deviation with dashed lines of the same color. The stimulation is also depicted with a solid black line. The left column contains data from day 1 and the right column contains data from day 2.

The subsequent Subsection 3.2.2 contains the mean position and velocity of the individual left and right eyes following the same color scheme. The velocities of the eye movements are in the lower row. The left two columns contain data from day 1 and the right two columns contain data from day 2.

3.2.1 Disconjugate Mean Movements

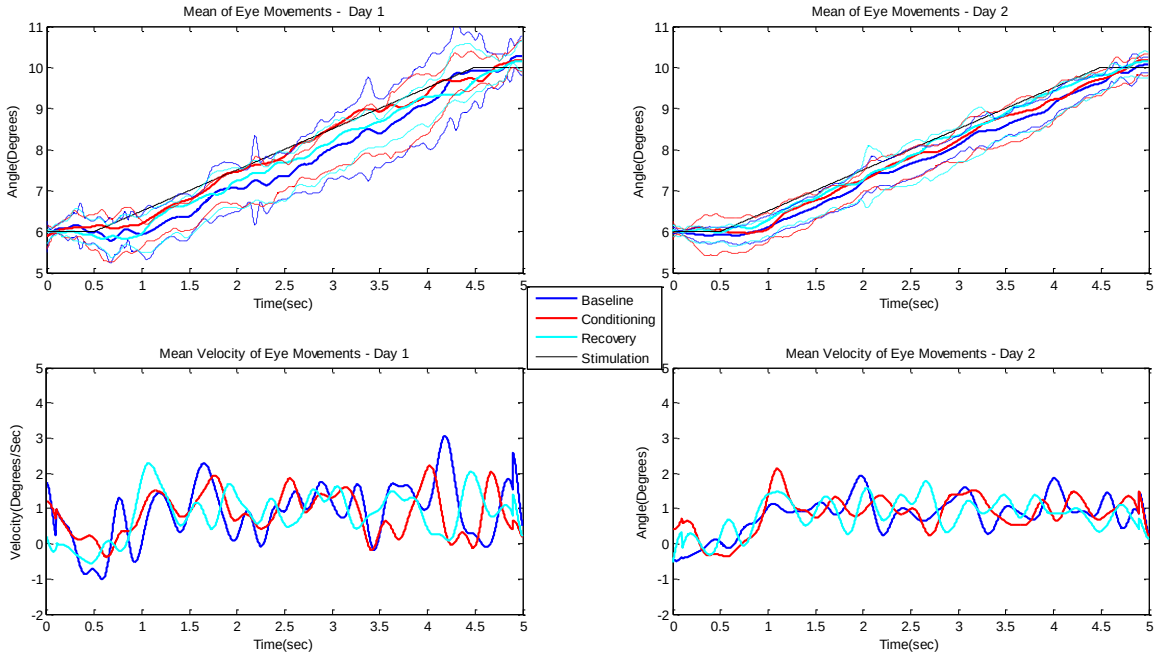

Figure 3.21 Mean disconjugate movements of subject 1 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

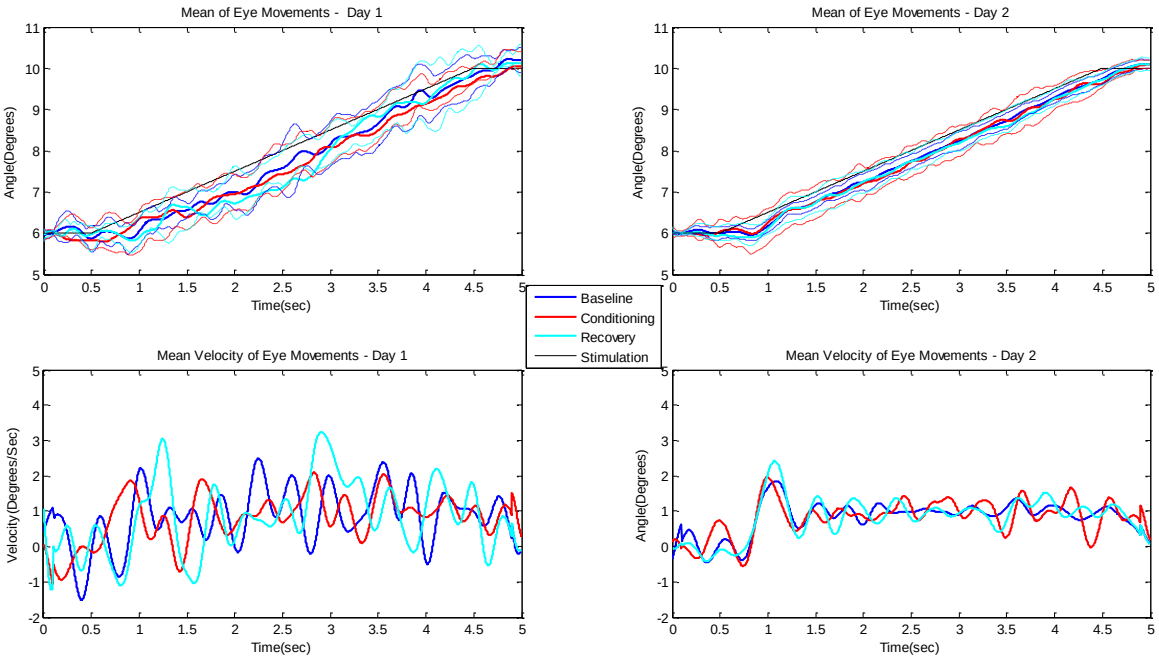

Figure 3.22 Mean disconjugate movements of subject 2 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

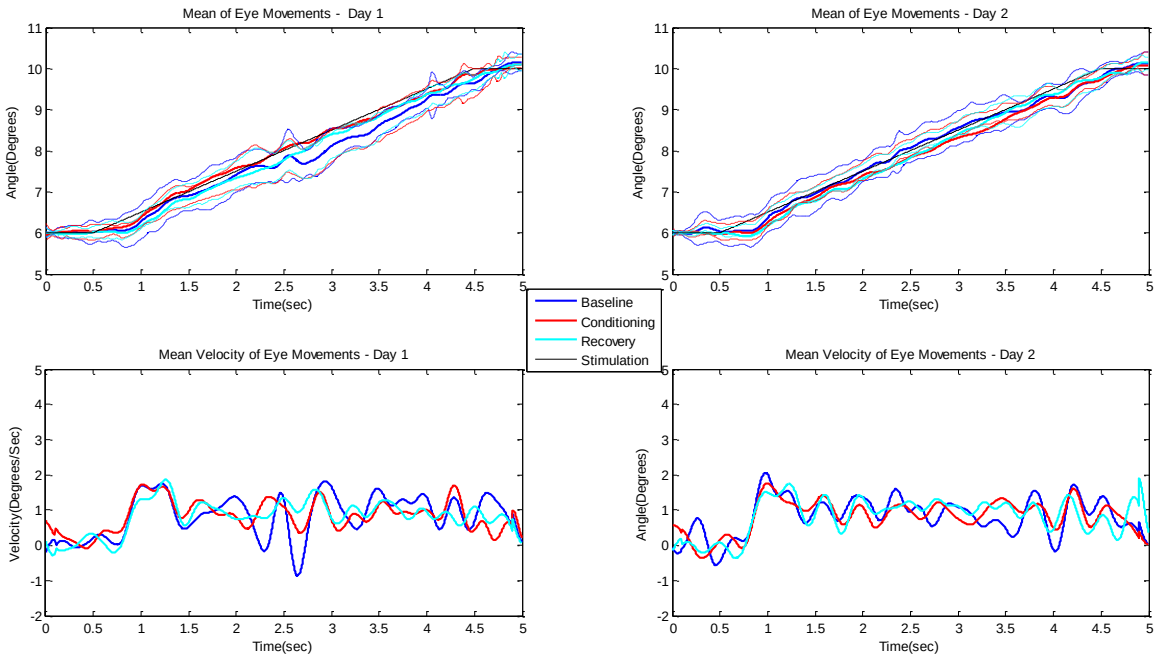

Figure 3.23 Mean disconjugate movements of subject 3 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

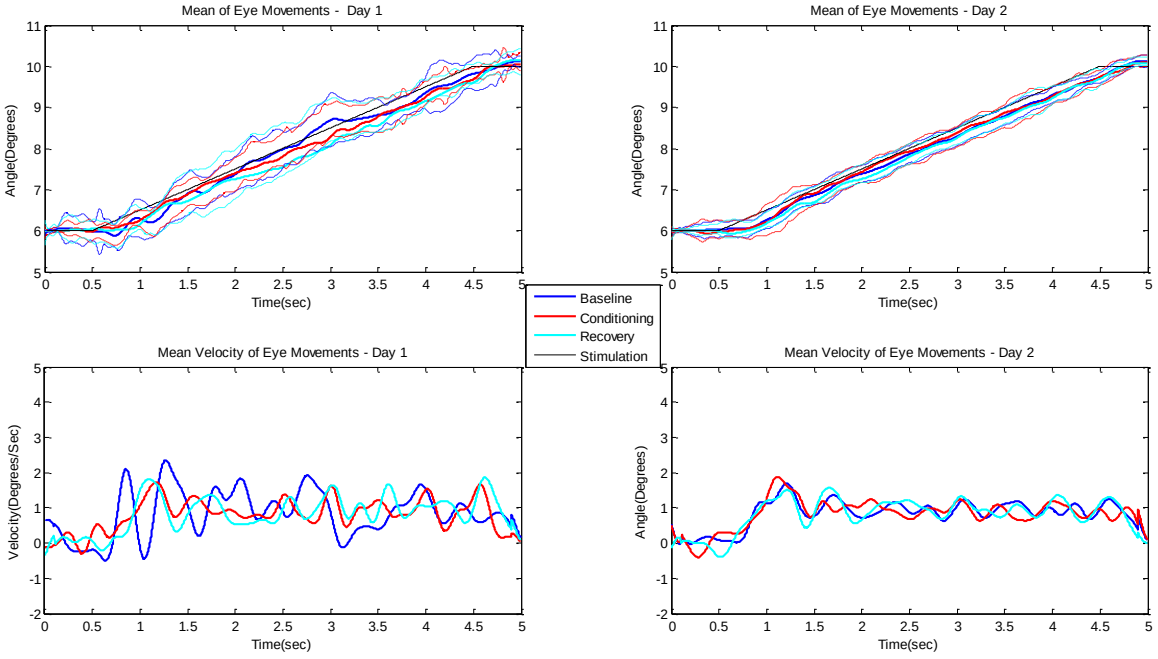

Figure 3.24 Mean disconjugate movements of subject 4 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

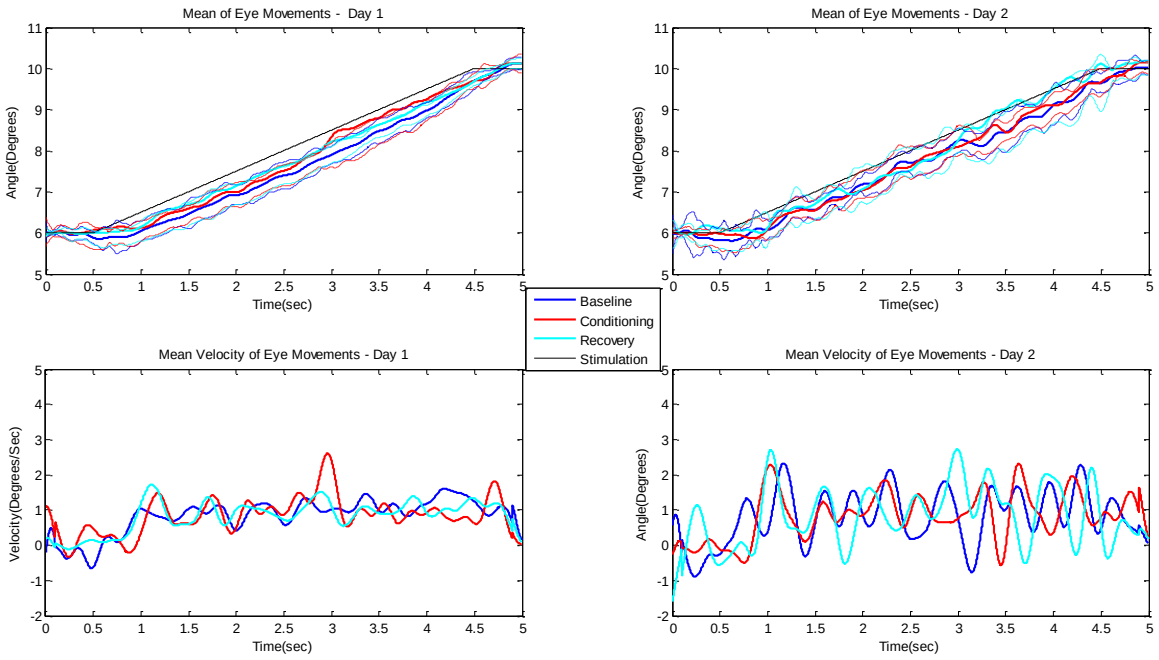

Figure 3.25 Mean disconjugate movements of subject 5 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

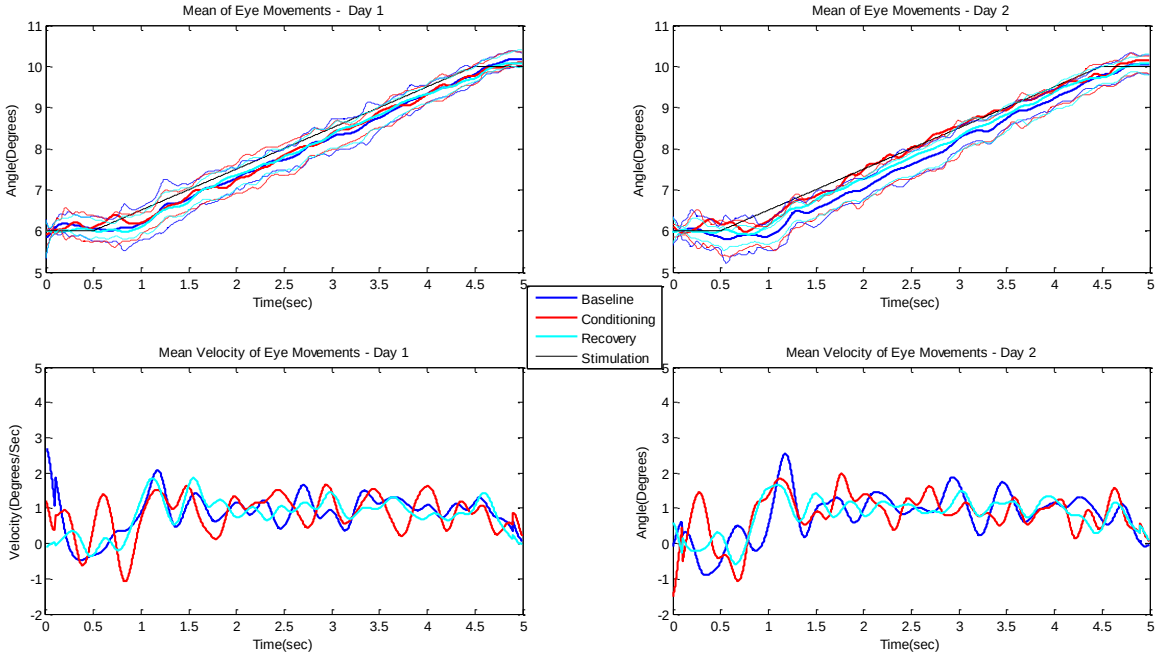

Figure 3.26 Mean disconjugate movements of subject 6 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

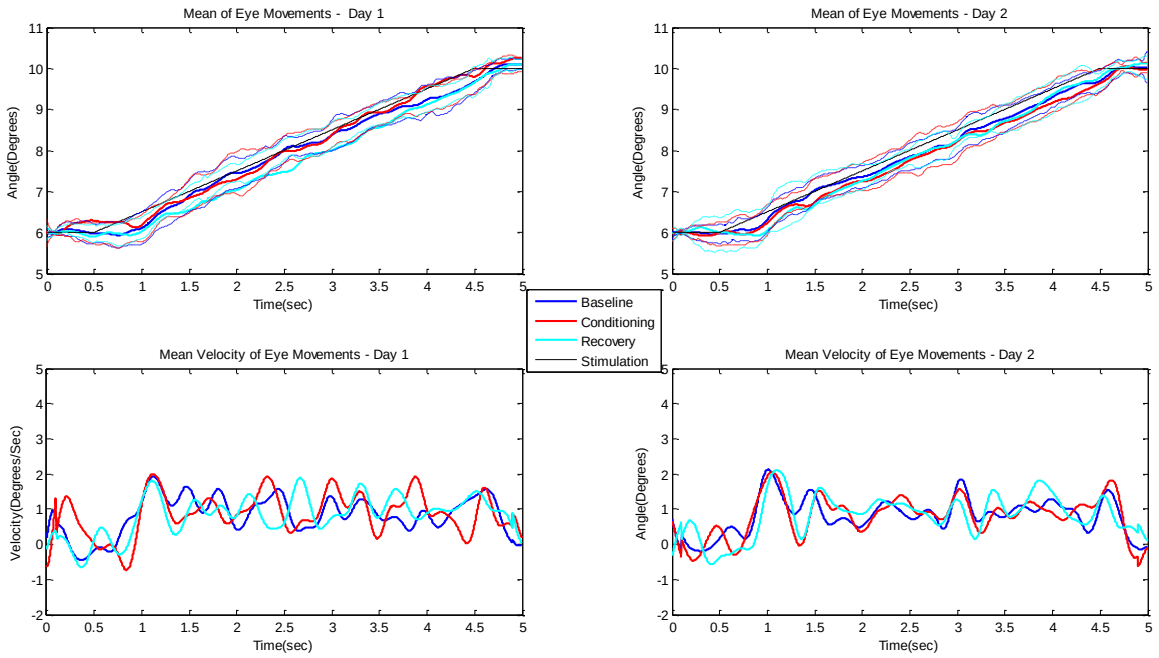

Figure 3.27 Mean disconjugate movements of subject 7 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

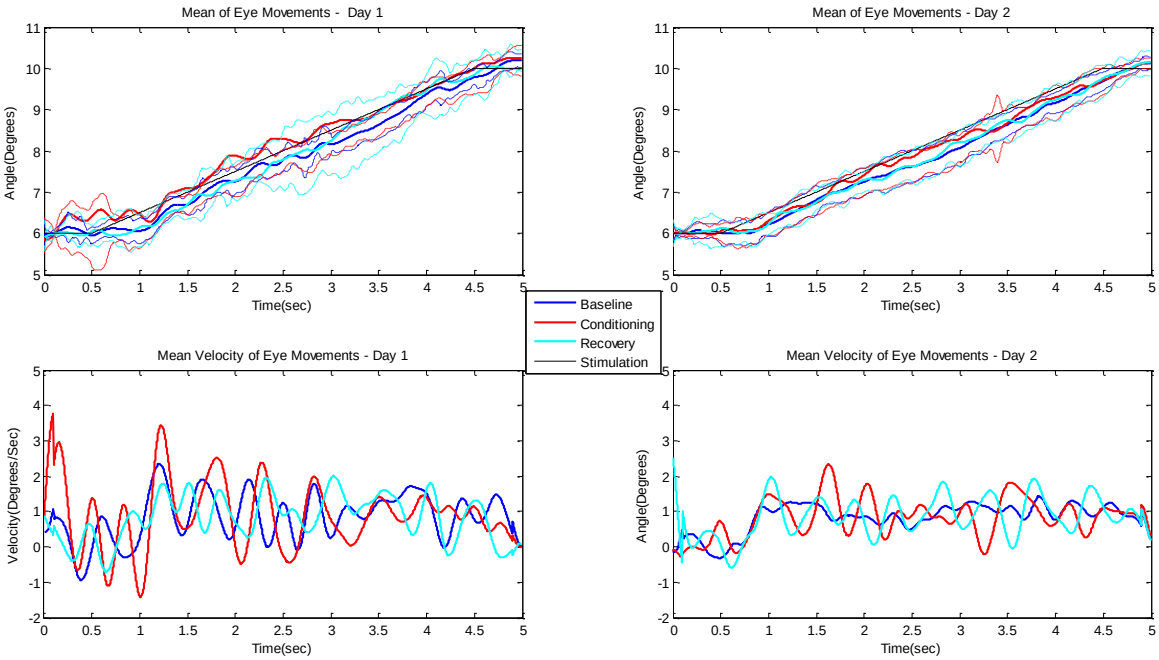

Figure 3.28 Mean disconjugate movements of subject 8 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

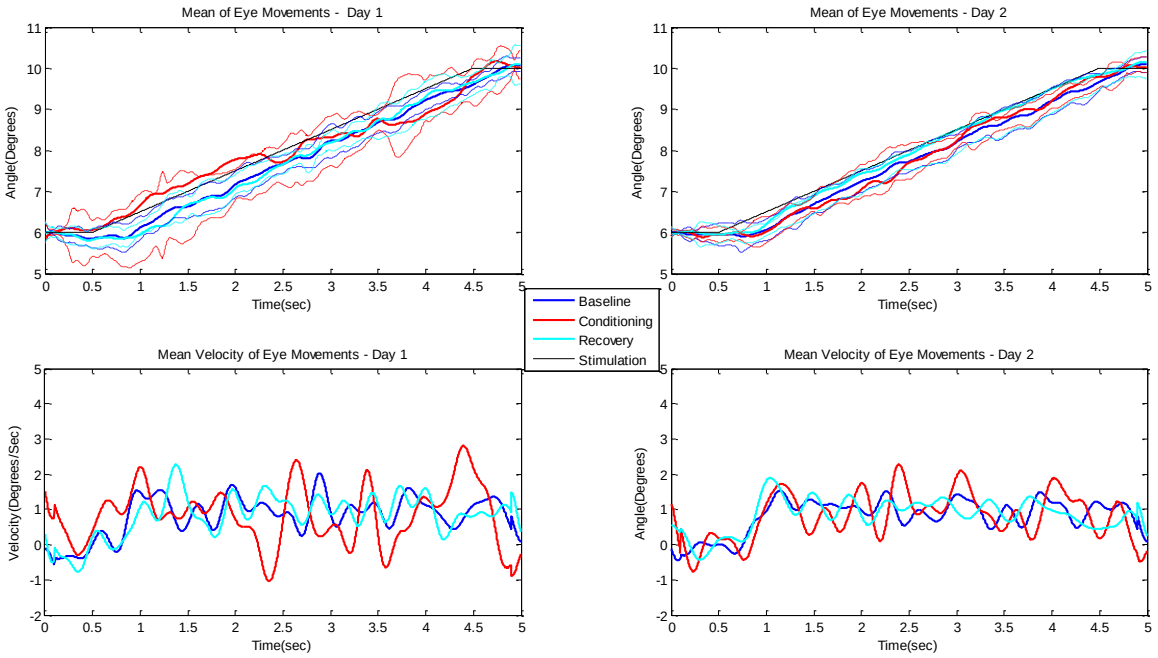

Figure 3.29 Mean disconjugate movements of subject 9 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

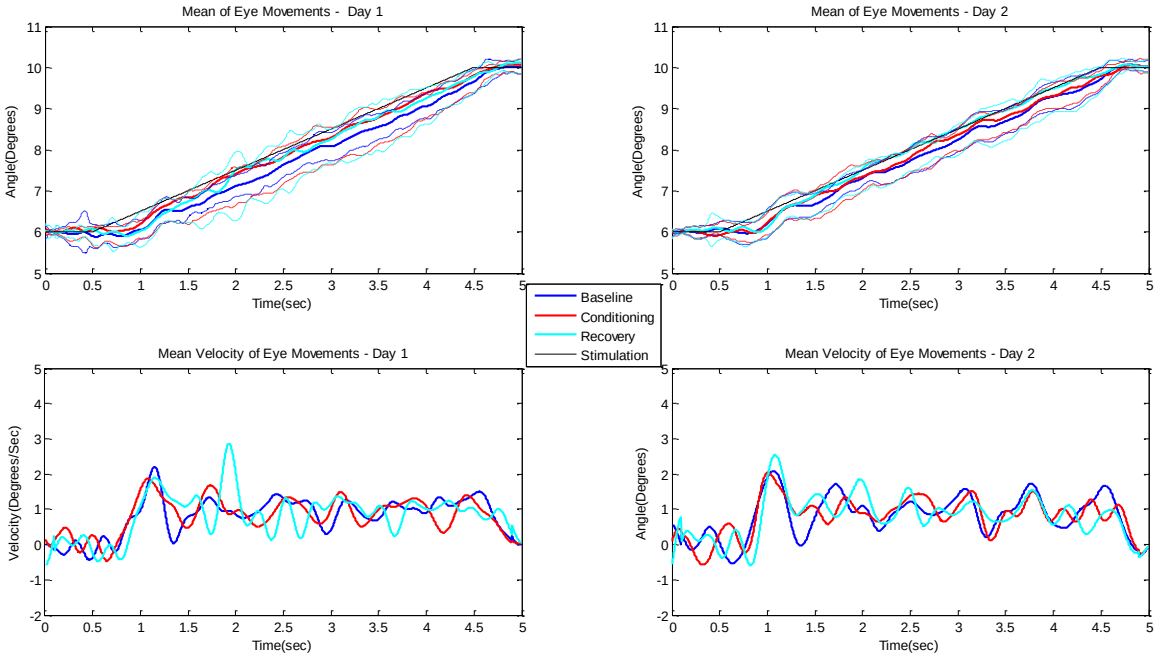

Figure 3.30 Mean disconjugate movements of subject 10 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data.

3.2.2 Left Eye versus Right Eye Movements

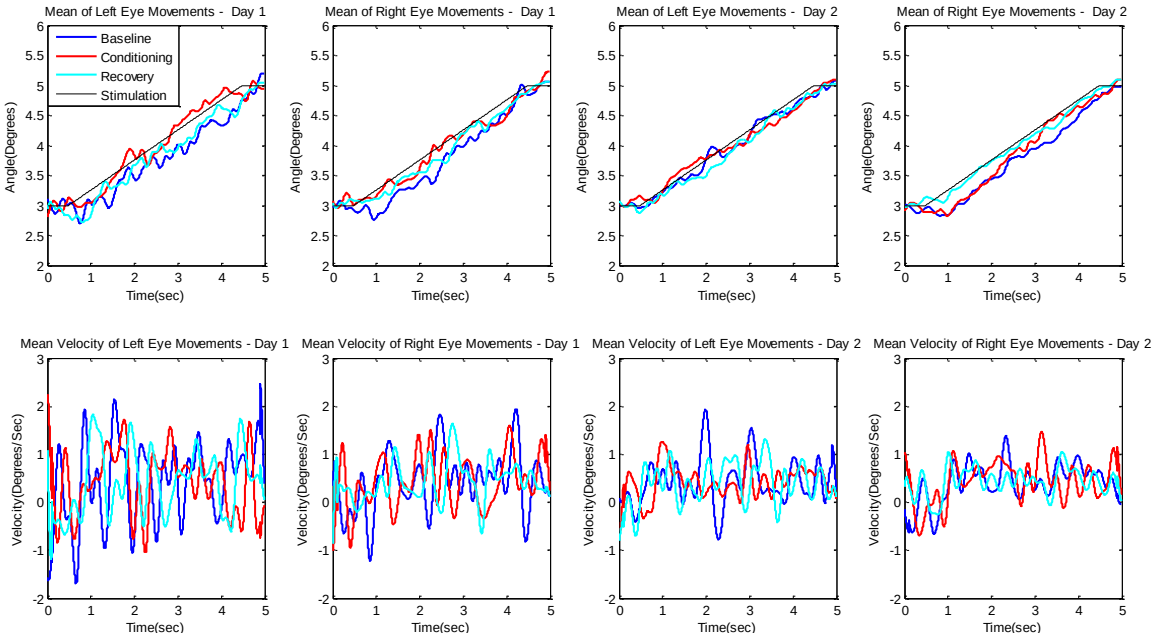

Figure 3.31 Odd columns contain the left eye data and even columns contain right eye data for Subject 1. Top row contains mean position data while the bottom row contains mean velocity data.

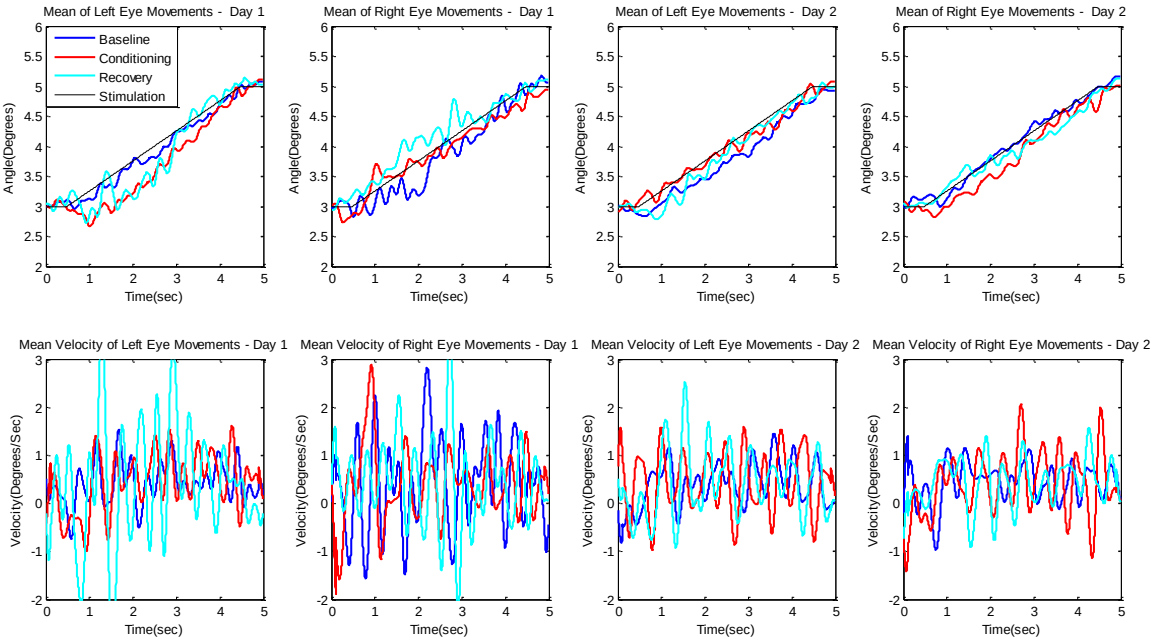

Figure 3.32 Odd columns contain the left eye data and even columns contain right eye data for Subject 2. Top row contains mean position data while the bottom row contains mean velocity data.

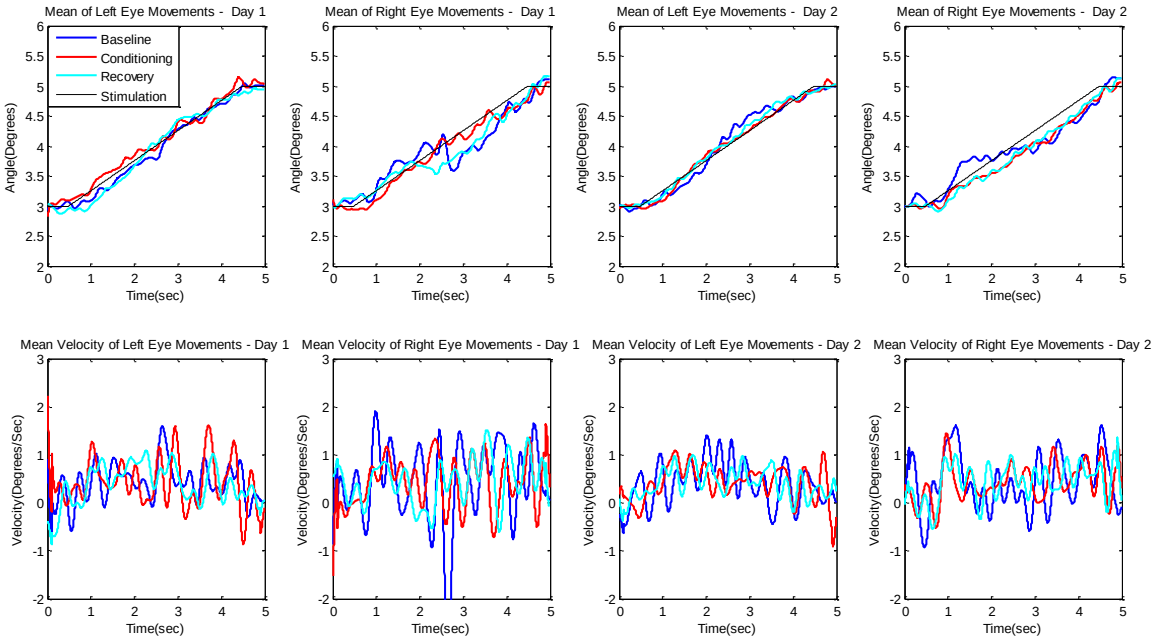

Figure 3.33 Odd columns contain the left eye data and even columns contain right eye data for Subject 3. Top row contains mean position data while the bottom row contains mean velocity data.

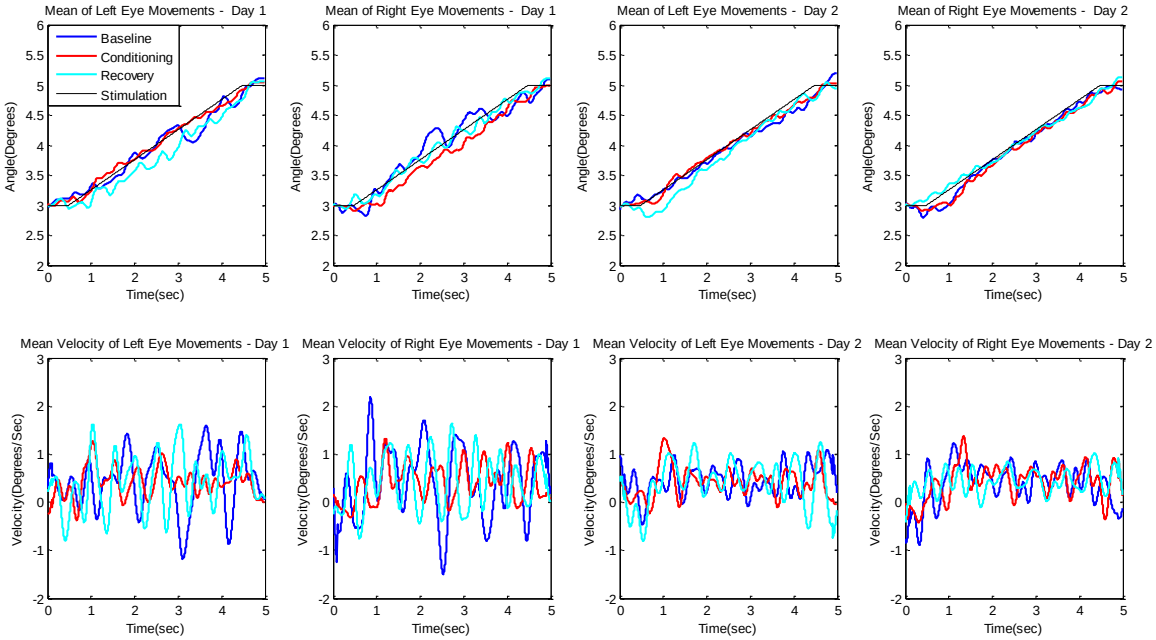

Figure 3.34 Odd columns contain the left eye data and even columns contain right eye data for Subject 4. Top row contains mean position data while the bottom row contains mean velocity data.

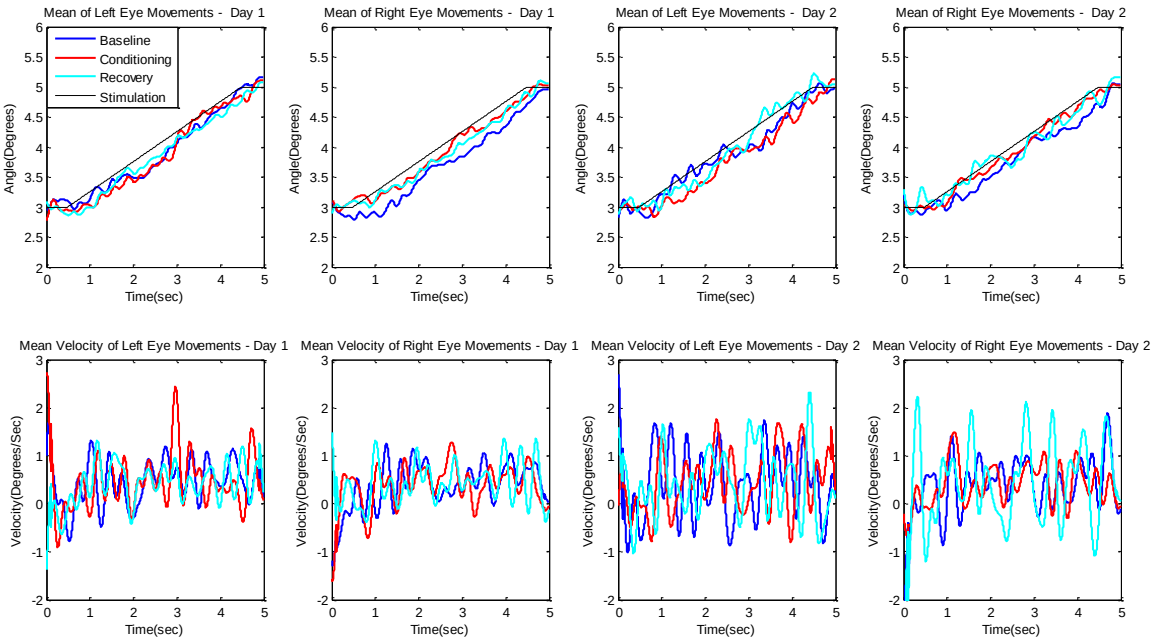

Figure 3.35 Odd columns contain the left eye data and even columns contain right eye data for Subject 5. Top row contains mean position data while the bottom row contains mean velocity data.

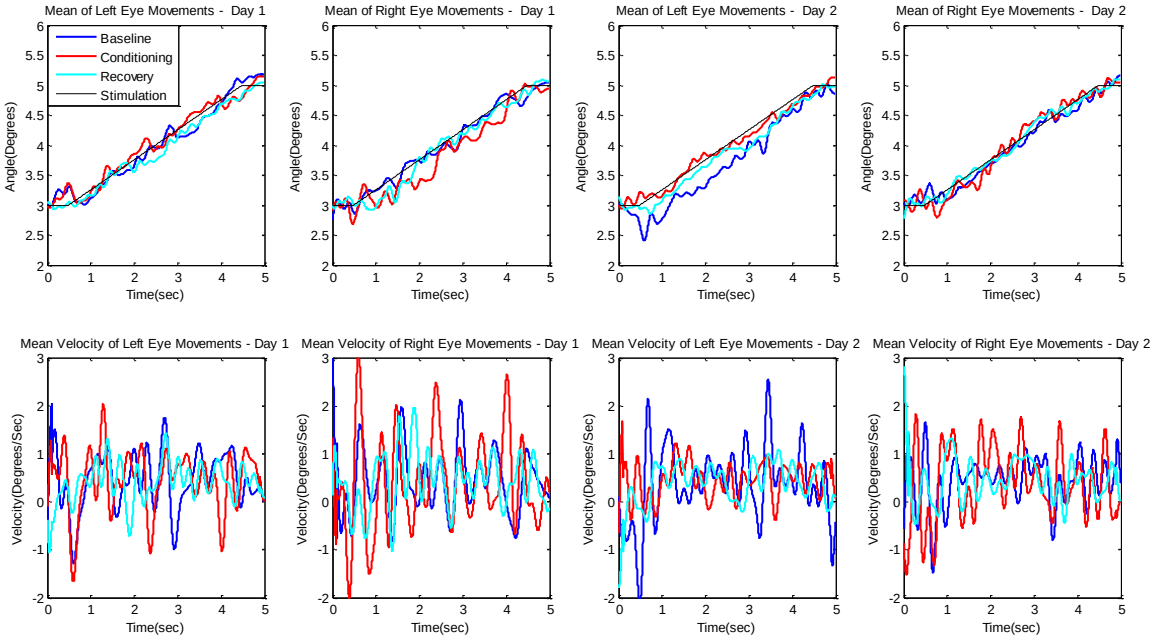

Figure 3.36 Odd columns contain the left eye data and even columns contain right eye data for Subject 6. Top row contains mean position data while the bottom row contains mean velocity data.

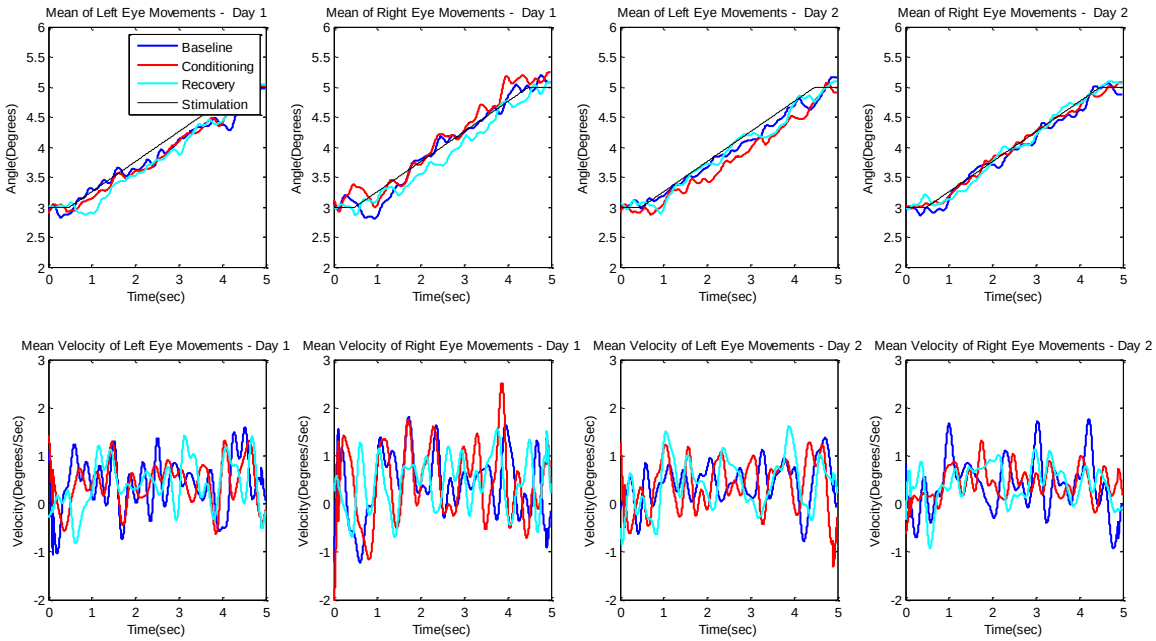

Figure 3.37 Odd columns contain the left eye data and even columns contain right eye data for Subject 7. Top row contains mean position data while the bottom row contains mean velocity data.

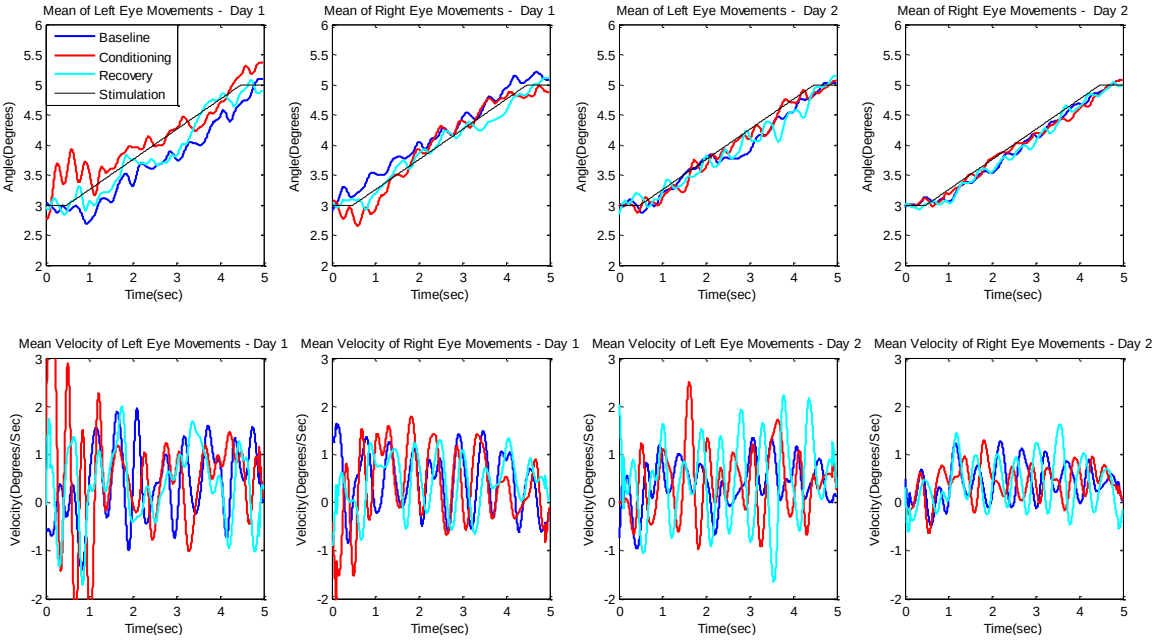

Figure 3.38 Odd columns contain the left eye data and even columns contain right eye data for Subject 8. Top row contains mean position data while the bottom row contains mean velocity data.

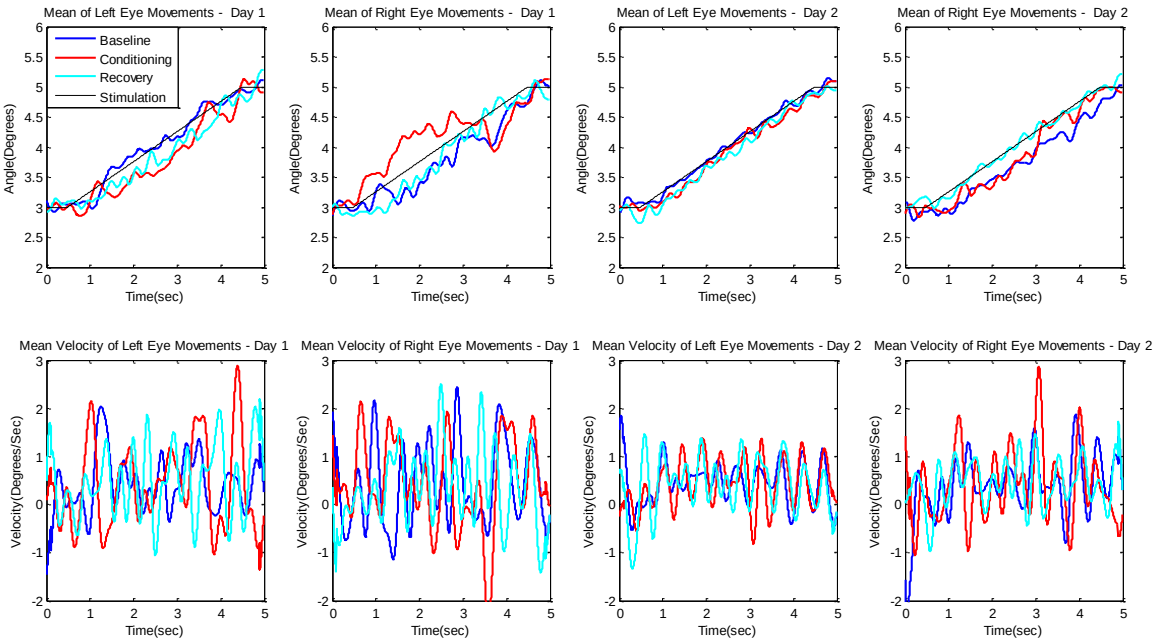

Figure 3.39 Odd columns contain the left eye data and even columns contain right eye data for Subject 9. Top row contains mean position data while the bottom row contains mean velocity data.

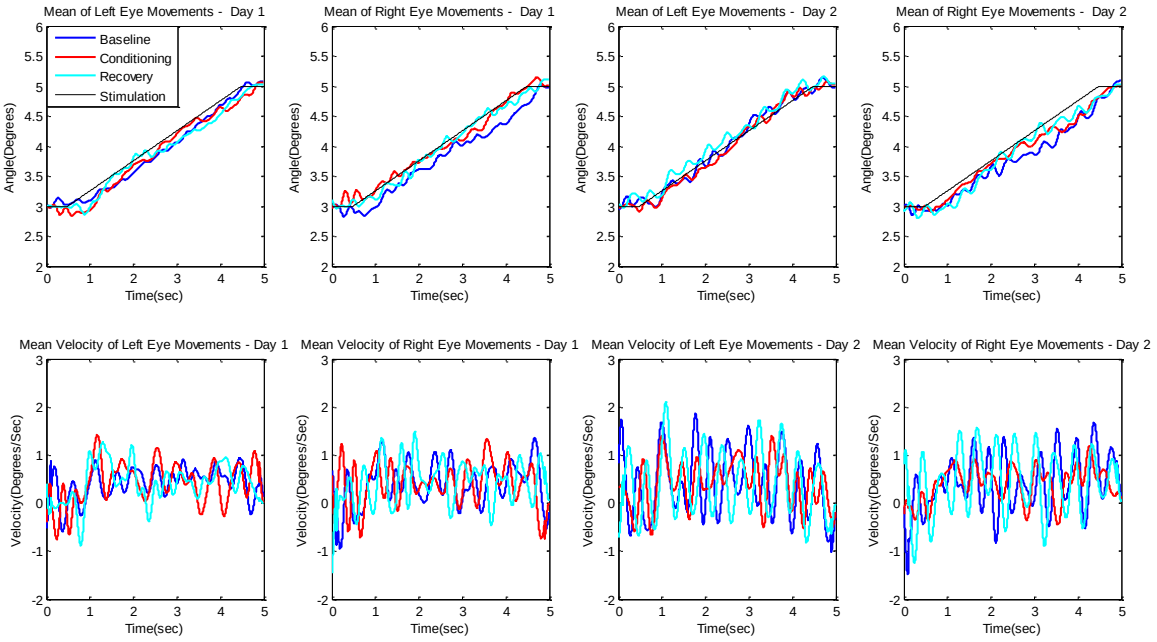

Figure 3.30 Odd columns contain the left eye data and even columns contain right eye data for Subject 10. Top row contains mean position data while the bottom row contains mean velocity data.

3.3 Root Mean Square Error Calculations

This section contains the RMSE data for the various types of analysis performed on the eye movements. The first Subsection 3.3.1 contains the RMSE of the disconjugate mean movements. The second Subsection 3.3.2 compares the RMSE between the left and right eye movements as they are correlated with the presented stimulus. Finally, in Subsection 3.3.3 each disconjugate mean movement of baseline, conditioning, and recovery are divided into three unique components of latency, transient and steady-state that last from start to 0.6 seconds, 0.6 seconds to 1.3 seconds, and 2 seconds to 4.5 seconds respectively.

3.3.1 RMSE of Disconjugate Means Movements

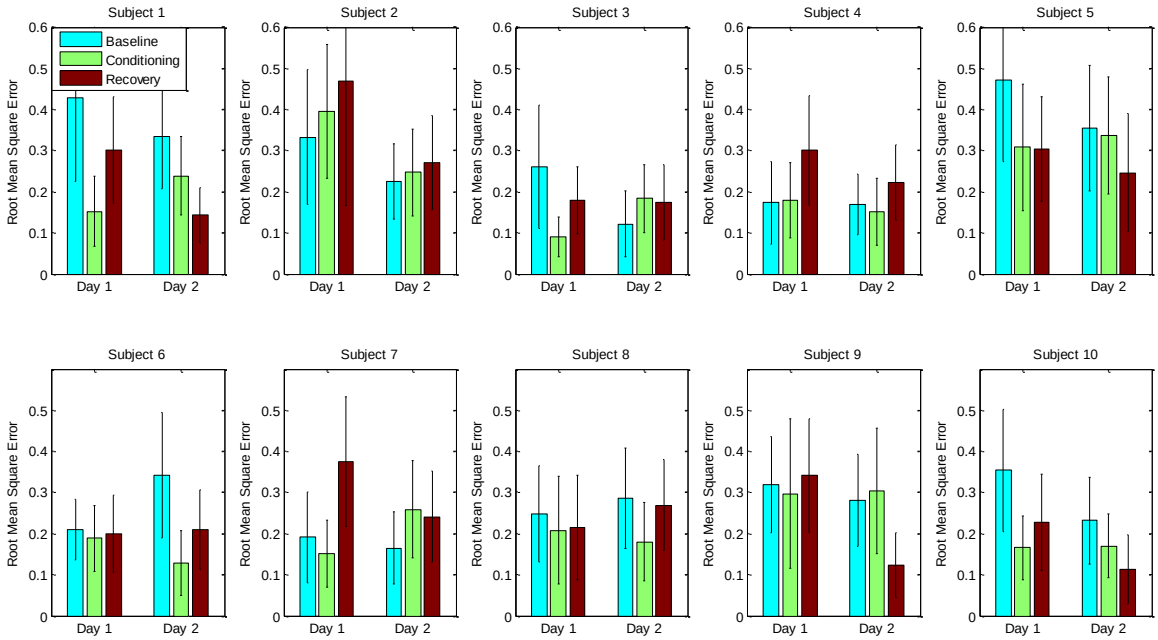

Figure 3.41 RMSE of disconjugate mean movements of all 10 subjects.

Table 3.1 RMSE of Disconjugate Mean Movements

		Baseline	Conditioning	Recovery
Subject 1	Trial 1	0.428±0.203	0.152±0.085	0.302±0.130
	Trial 2	0.335±0.128	0.239±0.095	0.143±0.068
Subject 2	Trial 1	0.333±0.164	0.400±0.162	0.468±0.302
	Trial 2	0.225±0.091	0.247±0.106	0.270±0.115
Subject 3	Trial 1	0.261±0.150	0.090±0.0490	0.180±0.082
	Trial 2	0.122±0.081	0.184±0.083	0.173±0.092
Subject 4	Trial 1	0.173±0.100	0.180±0.092	0.301±0.134
	Trial 2	0.169±0.074	0.150±0.081	0.223±0.091
Subject 5	Trial 1	0.472±0.199	0.308±0.155	0.303±0.127
	Trial 2	0.355±0.153	0.337±0.142	0.246±0.144
Subject 6	Trial 1	0.210±0.074	0.189±0.080	0.200±0.094
	Trial 2	0.341±0.153	0.129±0.078	0.211±0.097
Subject 7	Trial 1	0.191±0.111	0.150±0.081	0.374±0.158
	Trial 2	0.165±0.088	0.259±0.118	0.240±0.111
Subject 8	Trial 1	0.248±0.117	0.208±0.131	0.215±0.128
	Trial 2	0.287±0.122	0.180±0.095	0.269±0.111
Subject 9	Trial 1	0.318±0.117	0.297±0.181	0.342±0.139
	Trial 2	0.282±0.112	0.304±0.153	0.123±0.080
Subject 10	Trial 1	0.354±0.148	0.165±0.076	0.227±0.117
	Trial 2	0.232±0.105	0.170±0.078	0.114±0.085

A repeated-measures ANOVA confirmed a significant effect between the measured groups of baseline, conditioning, and recovery on Day 1 ((F(2,18) = 4.146, p = 0.03) with alpha = 0.05. The same measure on the groups from Day 2 however showed no significance ((F(2,18) = 1.45, p = 0.26). Since a significance is shown between the groups on Day 1, a post-hoc pairwise comparison is performed. The Post-hoc pairwise comparison (paired-samples t-tests, with alpha adjusted to 0.0167 to protect significant) revealed significant differences between Conditioning and Recovery stages (p = 0.008) for Day 1, but there were no difference between Baseline and Conditioning and Baseline and Recovery.

3.3.2 RMSE of Left Eye and Right Eye Movements

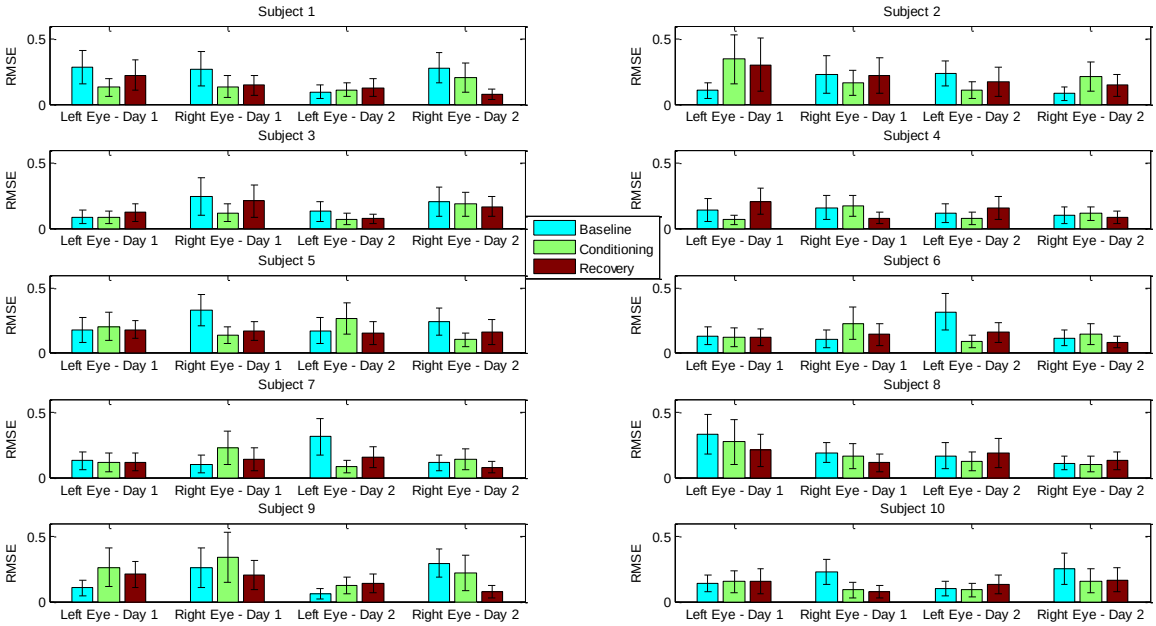

Figure 3.42 RMSE of left and right eye movements.

Table 3.2 RMSE of Left and Right Eye Movements

		Baseline		**Conditioning**		**Recovery**	
		Left	Right	Left	Right	Left	Right
Subject 1	Trial 1	0.284 ± 0.130	0.272 ± 0.135	0.129 ± 0.066	0.135 ± 0.084	0.224 ± 0.118	0.146 ± 0.077
	Trial 2	0.095 ± 0.054	0.278 ± 0.115	0.112 ± 0.055	0.206 ± 0.111	0.127 ± 0.068	0.078 ± 0.042
Subject 2	Trial 1	0.105 ± 0.057	0.229 ± 0.145	0.346 ± 0.188	0.166 ± .094	0.304 ± 0.202	0.217 ± 0.136
	Trial 2	0.233 ± 0.096	0.080 ± 0.050	0.107 ± 0.065	0.209 ± 0.111	0.173 ± 0.114	0.146 ± 0.082
Subject 3	Trial 1	0.087 ± 0.053	0.244 ± 0.143	0.086 ± 0.047	0.120 ± 0.071	0.123 ± 0.069	0.209 ± 0.124
	Trial 2	0.128 ± 0.075	0.205 ± 0.114	0.072 ± 0.043	0.184 ± 0.094	0.073 ± 0.040	0.166 ± 0.074
Subject 4	Trial 1	0.141 ± 0.089	0.160 ± 0.094	0.065 ± 0.036	0.173 ± 0.079	0.207 ± 0.099	0.078 ± 0.042
	Trial 2	0.115 ± 0.074	0.100 ± 0.063	0.073 ± 0.049	0.113 ± 0.054	0.159 ± 0.088	0.084 ± 0.046
Subject 5	Trial 1	0.176 ± 0.095	0.329 ± 0.119	0.202 ± 0.111	0.135 ± 0.063	0.177 ± 0.072	0.166 ± 0.075
	Trial 2	0.168 ± 0.103	0.238 ± 0.106	0.261 ± 0.121	0.099 ± 0.052	0.153 ± 0.090	0.158 ± 0.096
Subject 6	Trial 1	0.129 ± 0.069	0.103 ± 0.068	0.118 ± 0.072	0.226 ± 0.128	0.120 ± 0.066	0.138 ± 0.087
	Trial 2	0.314 ± 0.141	0.112 ± 0.060	0.087 ± 0.047	0.141 ± 0.083	0.156 ± 0.079	0.078 ± 0.044
Subject 7	Trial 1	0.197 ± 0.123	0.147 ± 0.099	0.174 ± 0.089	0.174 ± 0.105	0.214 ± 0.098	0.162 ± 0.090
	Trial 2	0.117 ± 0.070	0.104 ± 0.061	0.238 ± 0.104	0.057 ± 0.037	0.138 ± 0.090	0.085 ± 0.048
Subject 8	Trial 1	0.334 ± 0.150	0.192 ± 0.079	0.272 ± 0.176	0.163 ± 0.095	0.210 ± 0.123	0.114 ± 0.068
	Trial 2	0.166 ± 0.102	0.112 ± 0.056	0.120 ± 0.072	0.102 ± 0.060	0.188 ± 0.112	0.128 ± 0.069
Subject 9	Trial 1	0.104 ± 0.063	0.264 ± 0.152	0.263 ± 0.147	0.338 ± 0.192	0.208 ± 0.103	0.203 ± 0.114
	Trial 2	0.061 ± 0.040	0.296 ± 0.108	0.123 ± 0.066	0.221 ± 0.135	0.143 ± 0.072	0.074 ± 0.047
Subject 10	Trial 1	0.140 ± 0.068	0.227 ± 0.096	0.154 ± 0.084	0.088 ± 0.059	0.156 ± 0.093	0.079 ± 0.047
	Trial 2	0.098 ± 0.057	0.253 ± 0.119	0.090 ± 0.050	0.158 ± 0.092	0.132 ± 0.071	0.168 ± 0.090

Table 3.3 Statistical Analysis of Left and Right Eye Data

Statistical Comparison	df	F/T	p	Comment
ANOVA Day 1 Left Eye 3 stages	2,18	0.554	0.584	No significant effect between Baseline, Conditioning, and Recovery
ANOVA Day 1 Right Eyes 3 stages	2,18	3.310	0.060	No significant effect between Baseline, Conditioning, and Recovery
ANOVA Day 2 Left Eye 3 stages	2,18	0.358	0.704	No significant effect between Baseline, Conditioning, and Recovery
ANOVA Day 2 Right Eyes 3 stages	2,18	3.058	0.072	No significant effect between Baseline, Conditioning, and Recovery
L v R Paired t-test Baseline Day 1	9	-1.42	0.189	No significant effect between Left and Right eyes in Baseline Day 1
L v R Paired t-test Condition Day 1	9	-.036	0.972	No significant effect between Left and Right eyes in Conditioning Day 1
L v R Paired t-test Recovery Day 1	9	2.113	0.064	No significant effect between Left and Right eyes in Recovery Day 1
L v R Paired t-test Baseline Day 2	9	-.627	0.546	No significant effect between Left and Right eyes in Baseline Day 2
L v R Paired t-test Condition Day 2	9	-.604	0.561	No significant effect between Left and Right eyes in Conditioning Day 2
L v R Paired t-test Recovery Day 2	9	1.562	0.153	No significant effect between Left and Right eyes in Recovery Day 2

3.3.3 RMSE of Sectioned Disconjugate Mean Movements

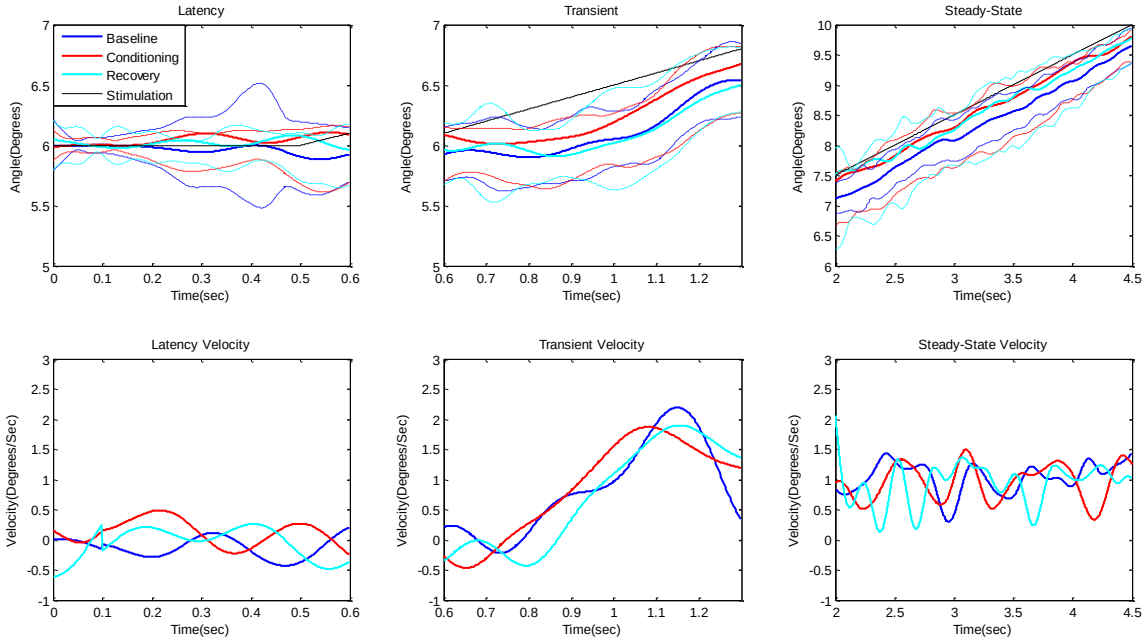

Figure 3.43 Example of mean disconjugate movements and their velocities divided into their respective stages of latency, transient, and steady-state.

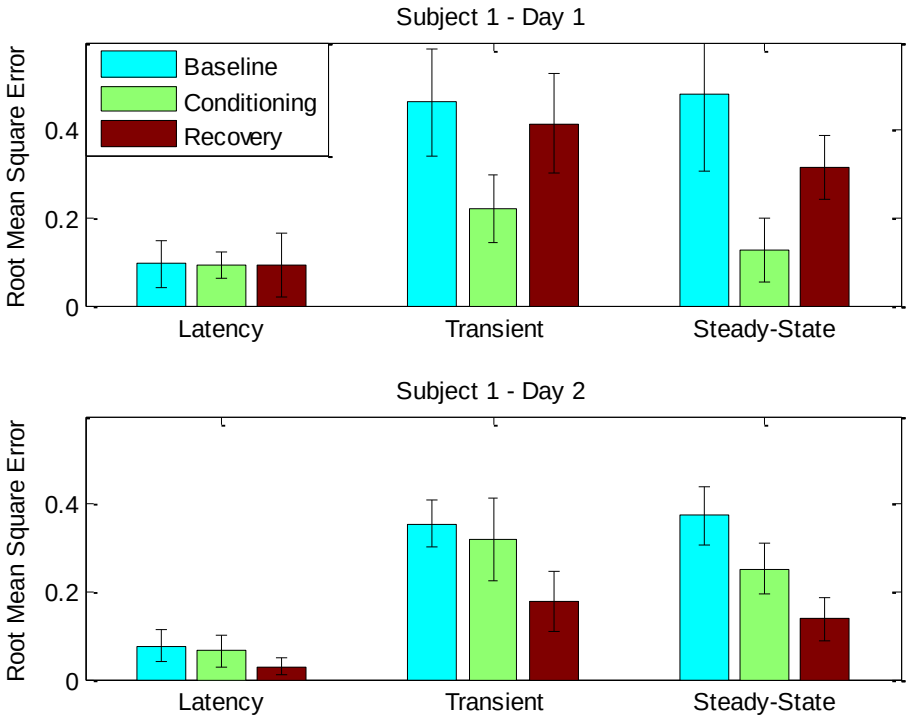

Figure 3.44 RMSE of Subject 1 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

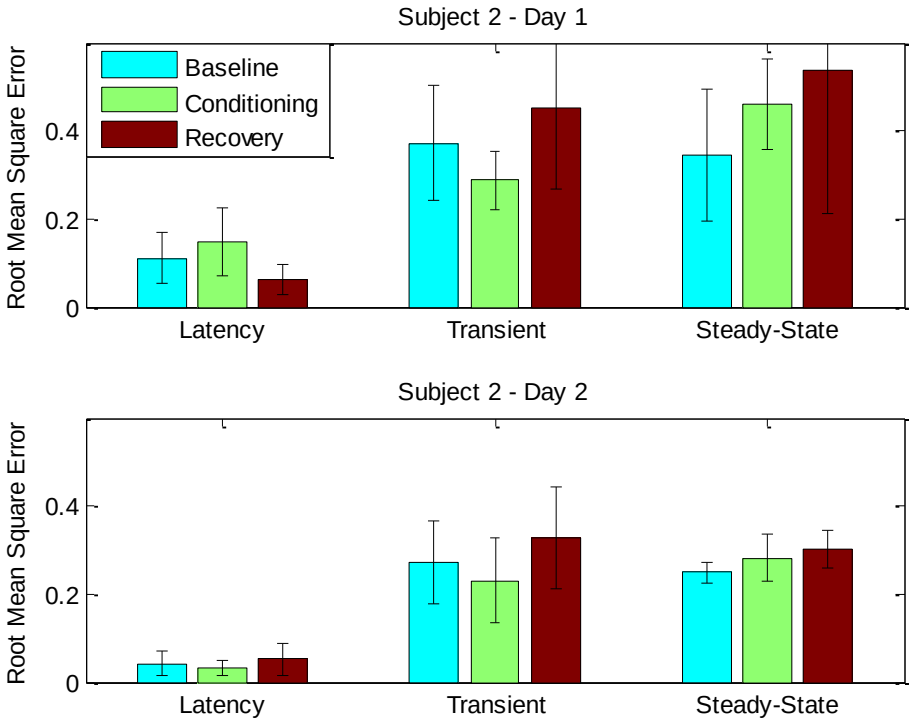

Figure 3.45 RMSE of Subject 2 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

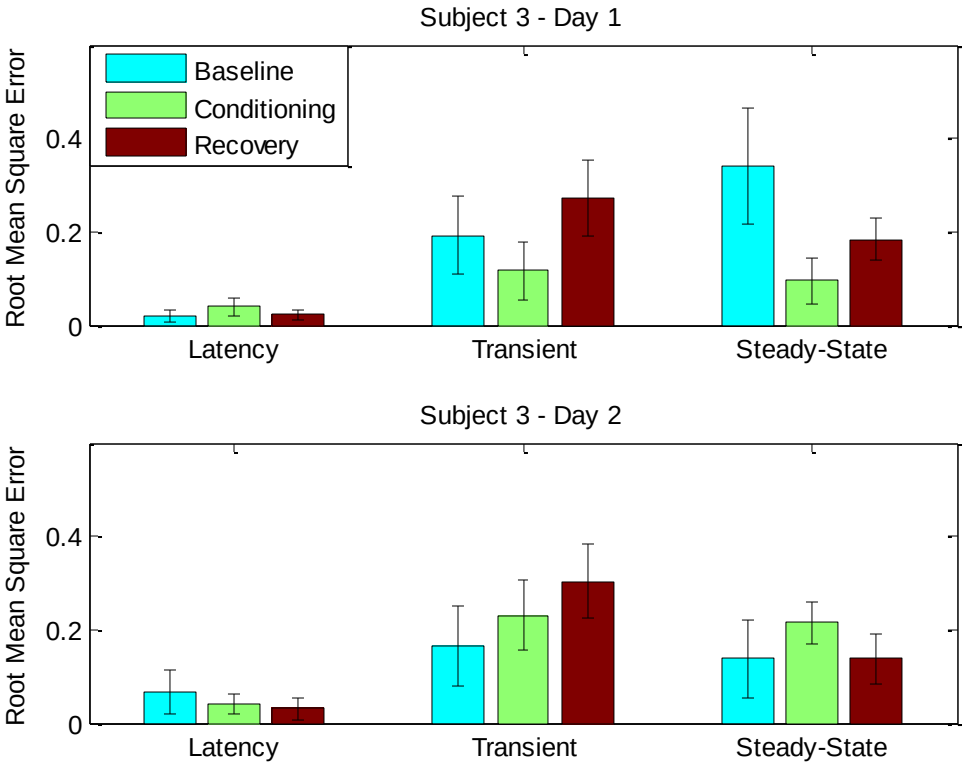

Figure 3.46 RMSE of Subject 3 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

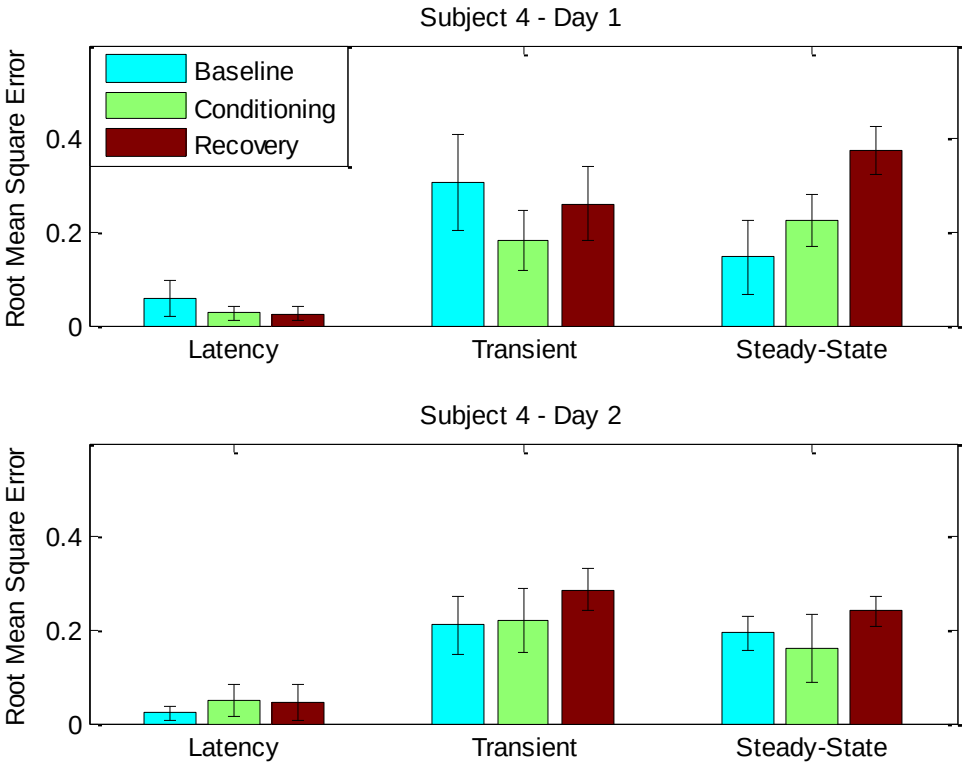

Figure 3.47 RMSE of Subject 4 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

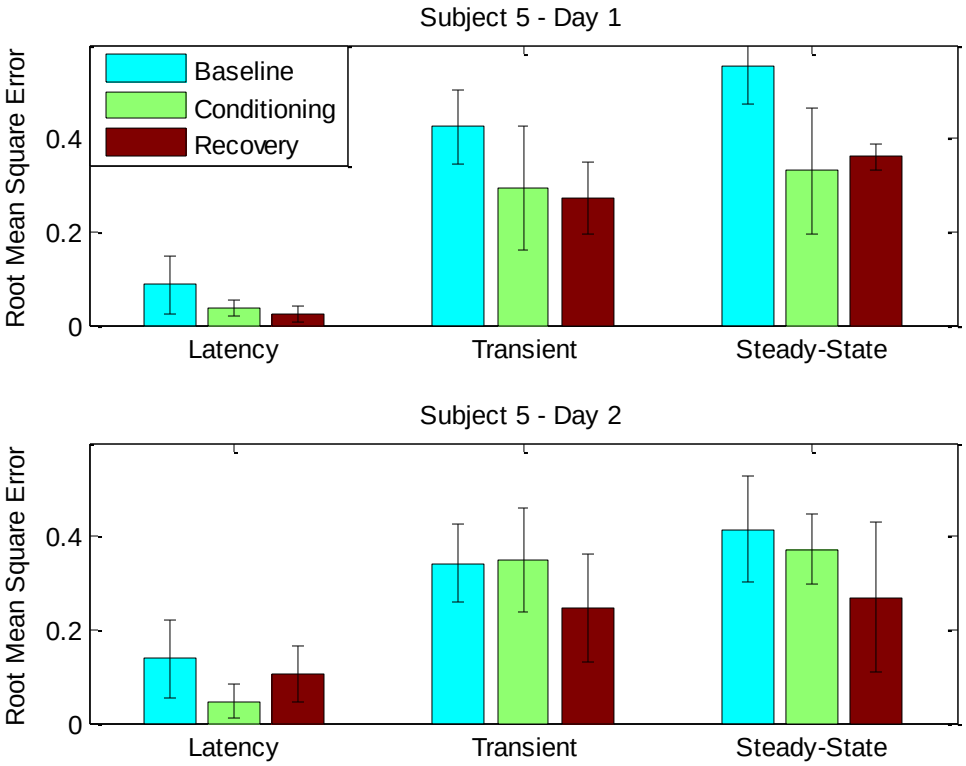

Figure 3.48 RMSE of Subject 5 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

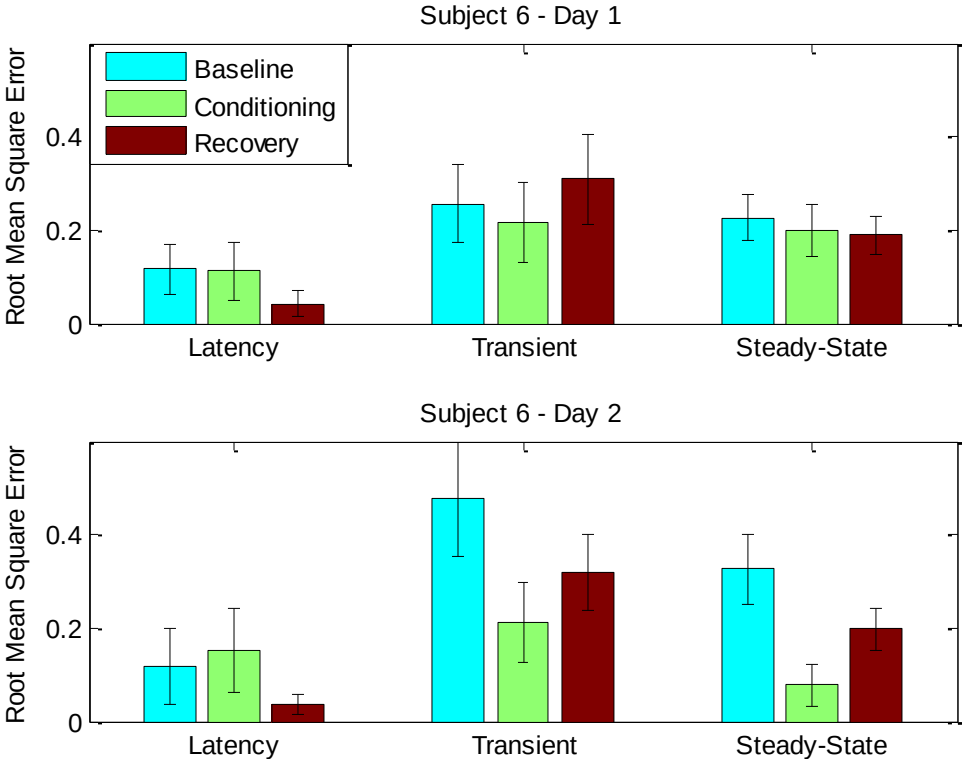

Figure 3.49 RMSE of Subject 6 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

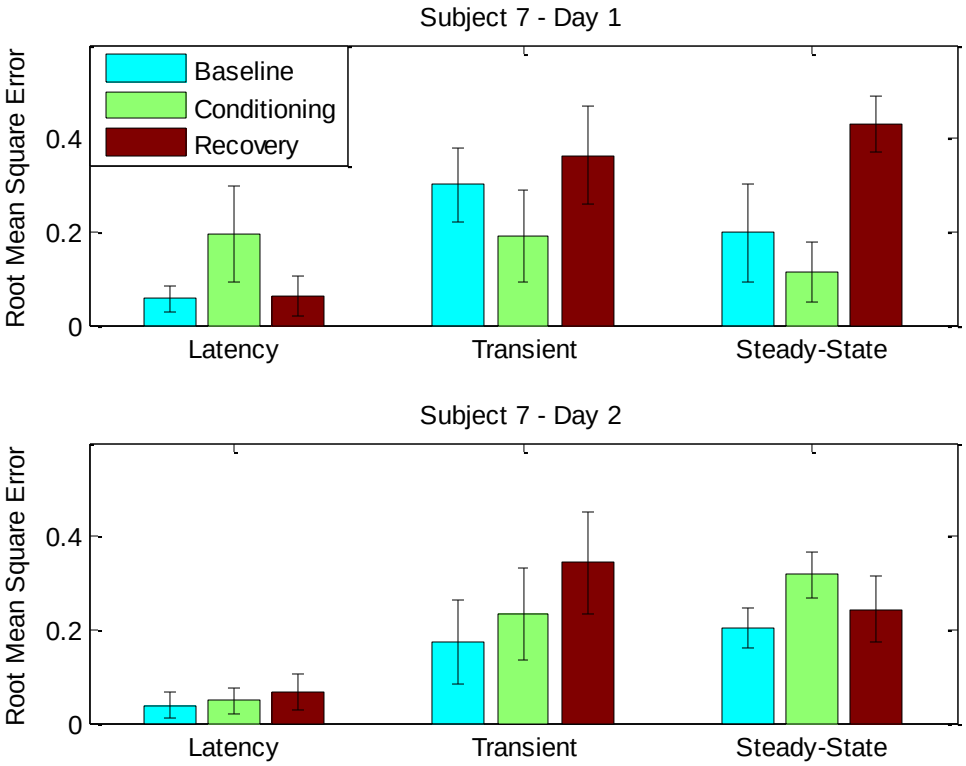

Figure 3.50 RMSE of Subject 7 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

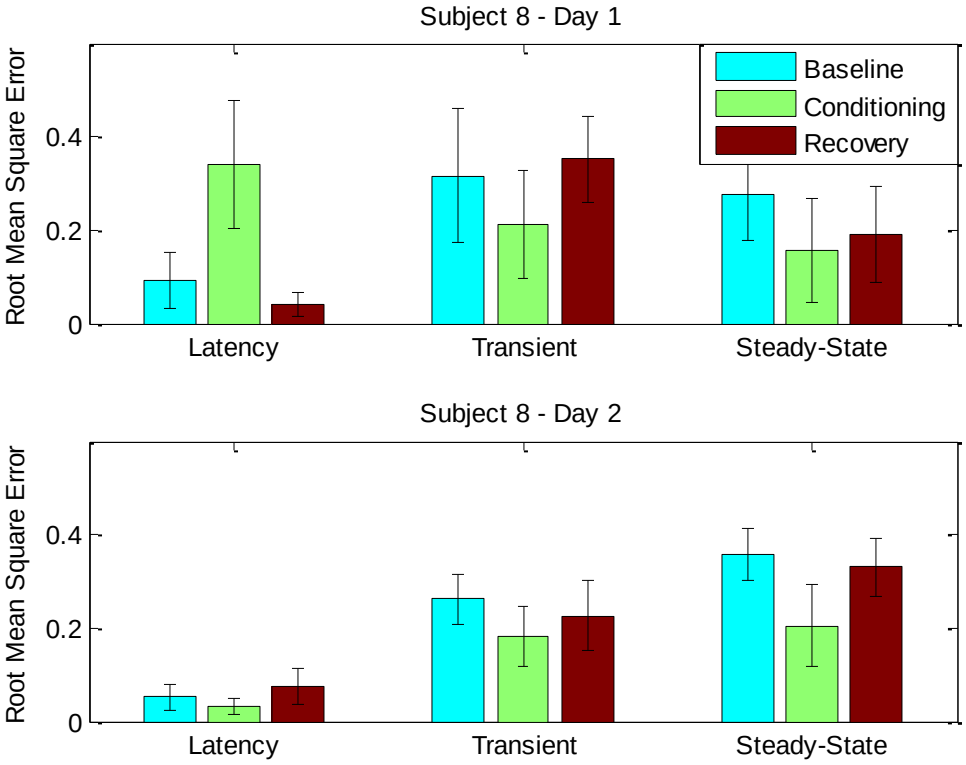

Figure 3.51 RMSE of Subject 8 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

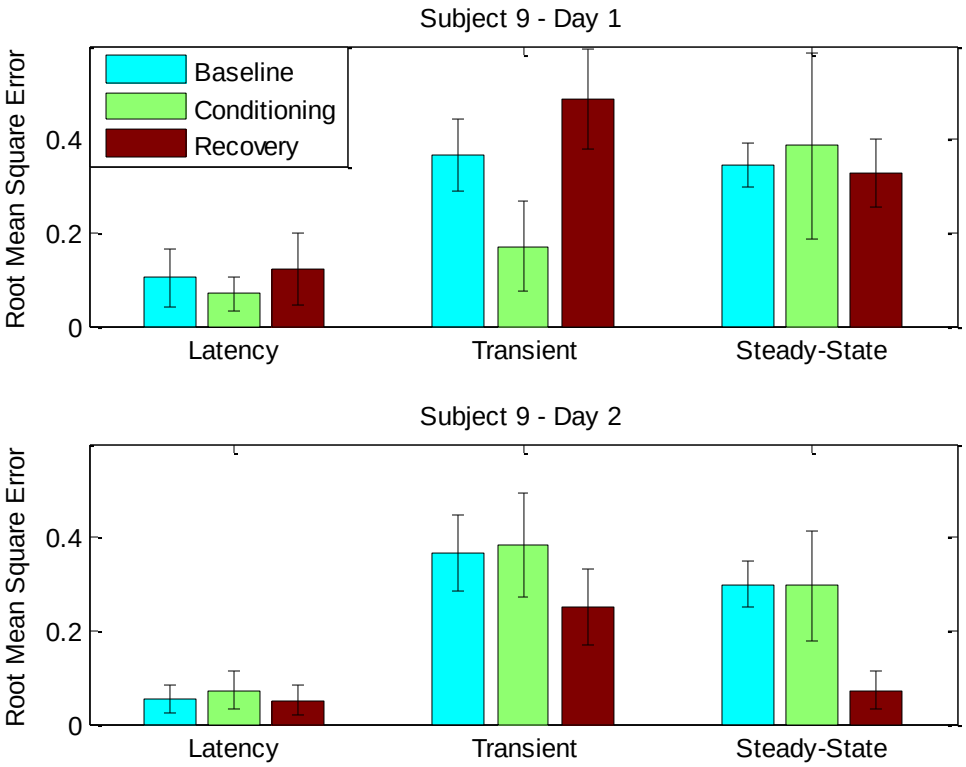

Figure 3.52 RMSE of Subject 9 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

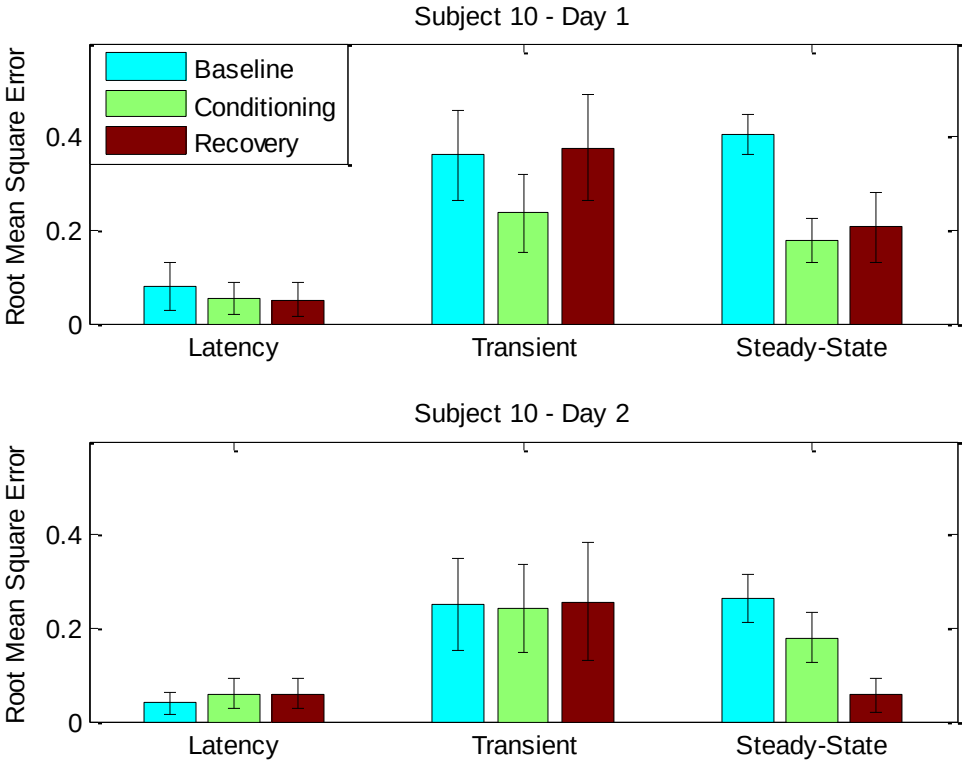

Figure 3.53 RMSE of Subject 10 sectioned disconjugate mean movements. Top plot consists data from day 1 and the bottom is of day 2.

Table 3.4 RMSE of Sectioned Disconjugate Mean Movements

Subject	Day	Stage	Baseline	Conditioning	Recovery
1	1	Latency	0.094±0.055	0.092±0.032	0.093±0.073
		Transient	0.463±0.122	0.222±0.078	0.416±0.113
		Steady-State	0.482±0.175	0.127±0.073	0.315±0.074
	2	Latency	0.076±0.037	0.065±0.035	0.030±0.018
		Transient	0.355±0.055	0.318±0.094	0.178±0.070
		Steady-State	0.373±0.065	0.252±0.059	0.137±0.050
2	1	Latency	0.110±0.058	0.147±0.079	0.062±0.035
		Transient	0.371±0.131	0.287±0.068	0.452±0.186
		Steady-State	0.347±0.150	0.460±0.103	0.537±0.327
	2	Latency	0.042±0.028	0.032±0.017	0.052±0.038
		Transient	0.273±0.095	0.231±0.097	0.329±0.116
		Steady-State	0.249±0.024	0.282±0.055	0.302±0.041
3	1	Latency	0.020±0.012	0.039±0.018	0.022±0.012
		Transient	0.193±0.085	0.116±0.062	0.274±0.082
		Steady-State	0.341±0.124	0.095±0.050	0.184±0.045
	2	Latency	0.067±0.046	0.040±0.022	0.031±0.023
		Transient	0.166±0.086	0.231±0.074	0.303±0.080
		Steady-State	0.138±0.084	0.215±0.043	0.138±0.053
4	1	Latency	0.057±0.038	0.026±0.017	0.025±0.016
		Transient	0.306±0.103	0.182±0.065	0.261±0.080
		Steady-State	0.147±0.079	0.223±0.055	0.374±0.051
	2	Latency	0.022±0.015	0.049±0.034	0.044±0.037
		Transient	0.211±0.062	0.221±0.071	0.286±0.045
		Steady-State	0.193±0.037	0.162±0.074	0.240±0.032
5	1	Latency	0.086±0.062	0.038±0.018	0.023±0.017
		Transient	0.425±0.079	0.294±0.132	0.271±0.078
		Steady-State	0.556±0.084	0.331±0.135	0.361±0.028
	2	Latency	0.138±0.083	0.047±0.035	0.105±0.060
		Transient	0.343±0.082	0.349±0.112	0.247±0.115
		Steady-State	0.415±0.113	0.373±0.075	0.269±0.160
6	1	Latency	0.116±0.055	0.113±0.062	0.043±0.026
		Transient	0.256±0.083	0.218±0.087	0.309±0.097
		Steady-State	0.227±0.050	0.199±0.055	0.190±0.040
	2	Latency	0.117±0.081	0.153±0.090	0.035±0.022
		Transient	0.480±0.127	0.211±0.086	0.319±0.083
		Steady-State	0.326±0.076	0.078±0.047	0.198±0.045
7	1	Latency	0.056±0.027	0.195±0.102	0.062±0.042
		Transient	0.301±0.080	0.192±0.099	0.364±0.106
		Steady-State	0.198±0.104	0.114±0.066	0.432±0.059
	2	Latency	0.038±0.027	0.048±0.028	0.066±0.039
		Transient	0.173±0.090	0.233±0.097	0.344±0.110
		Steady-State	0.205±0.042	0.318±0.050	0.244±0.070

Table 3.4 RMSE of Sectioned Disconjugate Mean Movements continued

Subject	Day	Stage	Baseline	Conditioning	Recovery
8	1	Latency	0.092±0.060	0.341±0.137	0.042±0.026
		Transient	0.316±0.143	0.212±0.116	0.353±0.093
		Steady-State	0.276±0.096	0.157±0.111	0.190±0.104
	2	Latency	0.052±0.028	0.031±0.016	0.076±0.040
		Transient	0.262±0.053	0.182±0.065	0.226±0.074
		Steady-State	0.357±0.054	0.205±0.089	0.330±0.063
9	1	Latency	0.104±0.062	0.070±0.038	0.123±0.076
		Transient	0.367±0.076	0.171±0.095	0.486±0.106
		Steady-State	0.343±0.047	0.387±0.200	0.326±0.073
	2	Latency	0.053±0.110	0.072±0.334	0.085±0.051
		Transient	0.367±0.101	0.385±0.145	0.250±0.071
		Steady-State	0.300±0.148	0.296±0.121	0.072±0.036
10	1	Latency	0.079±0.053	0.053±0.035	0.051±0.037
		Transient	0.360±0.098	0.236±0.084	0.377±0.114
		Steady-State	0.405±0.044	0.178±0.047	0.206±0.074
	2	Latency	0.039±0.024	0.058±0.032	0.058±0.033
		Transient	0.251±0.100	0.241±0.095	0.257±0.126
		Steady-State	0.265±0.051	0.179±0.055	0.057±0.037

A repeated-measures ANOVA confirmed a significant effect between the transient stages of baseline, conditioning, and recovery on Day 1 ((F(2,18) = 20.49, p = 0.00002) with alpha = 0.05. The Post-hoc pairwise comparison (paired-samples t-tests, with alpha adjusted to 0.0167 to protect significant) revealed significant differences between baseline and conditioning stages (p = 0.00009) and between conditioning and recovery stages (p = .00056). A repeated-measures ANOVA performed on day 2 yielded no significant change ((F(2,18) = 0.339, p = 0.717) with alpha = 0.05.

3.4 Effects of Filtering

This section contains the mean disconjugate movements and the RMSE of its three sections of subject 3 when the data was not filtered and when the data was put through a 50Hz low pass filter.

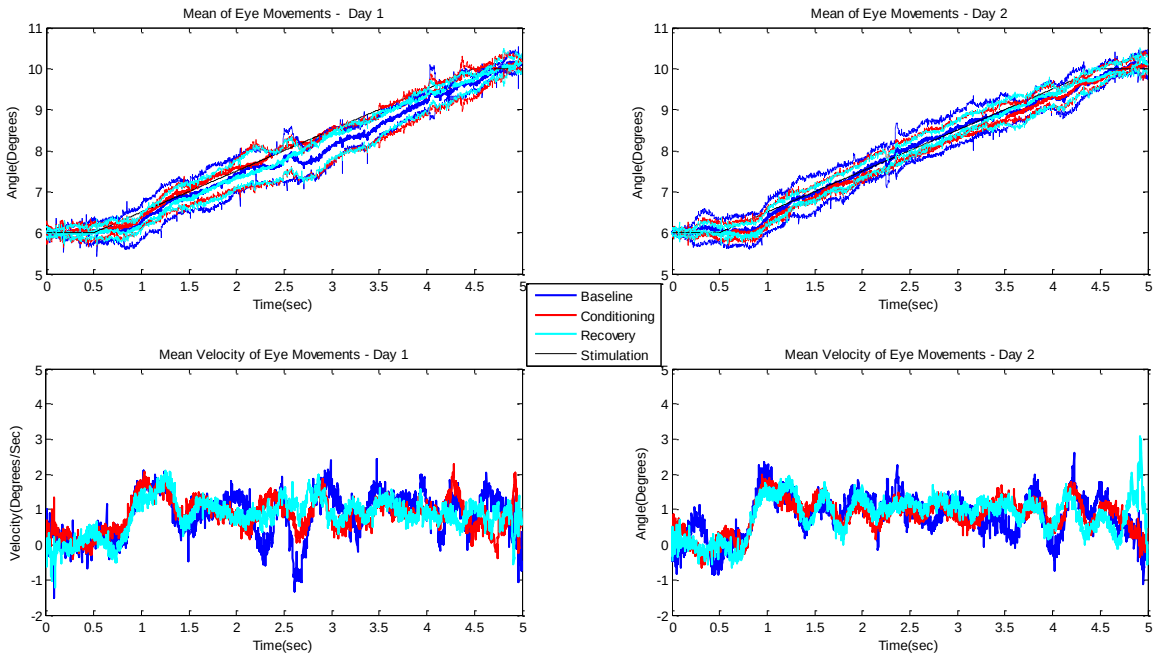

Figure 3.54 Mean disconjugate movements of subject 3 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data. No Filtering was performed on this data set.

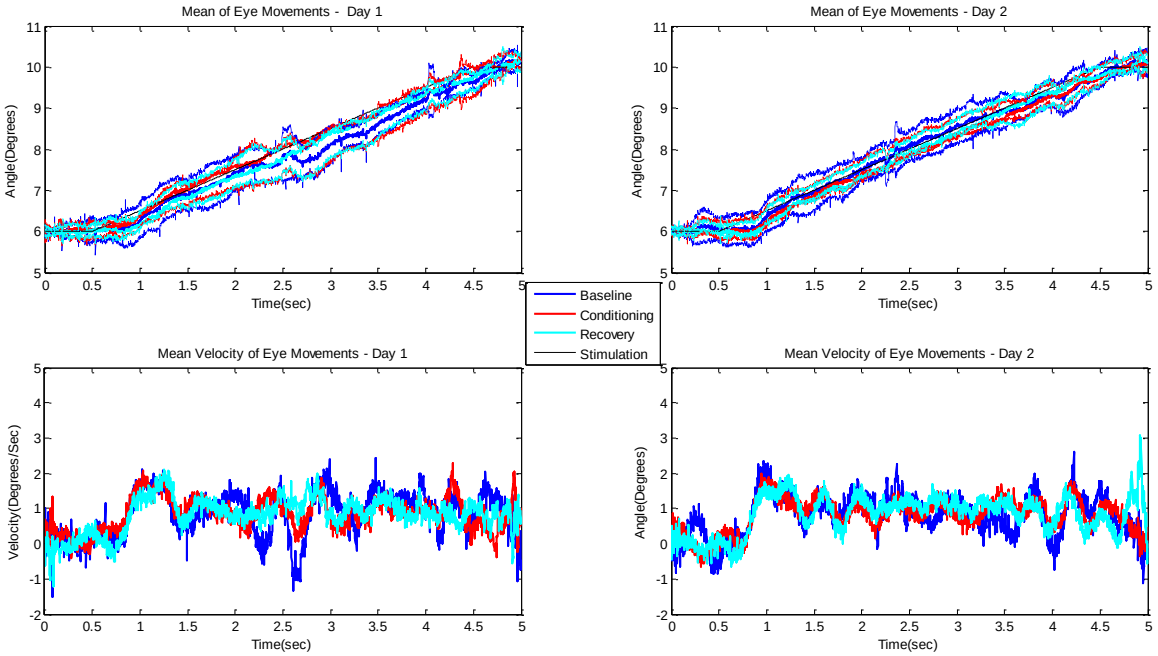

Figure 3.55 Mean disconjugate movements of subject 3 on top row and their velocities on the bottom row. Left column is day 1 data and right column is day 2 data. This dataset was put through a 50Hz low pass filter Butterworth filter.

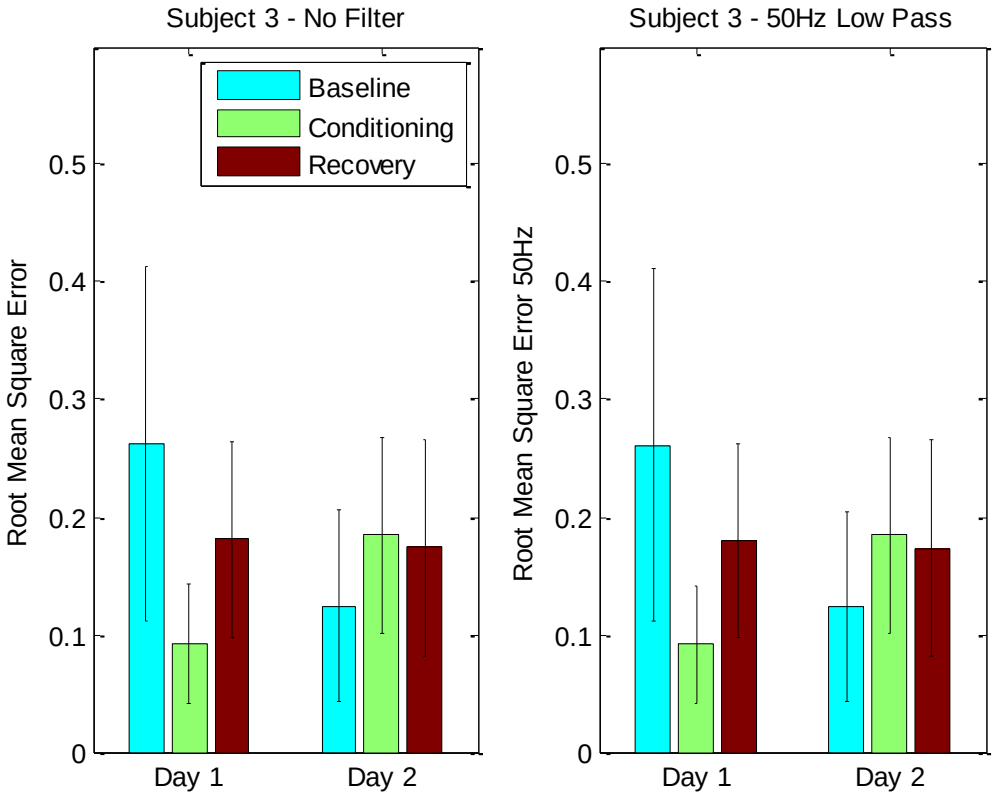

Figure 3.56 RMSE of disconjugate mean movements of Subject 3 data with no filter and a 50Hz low pass filter.

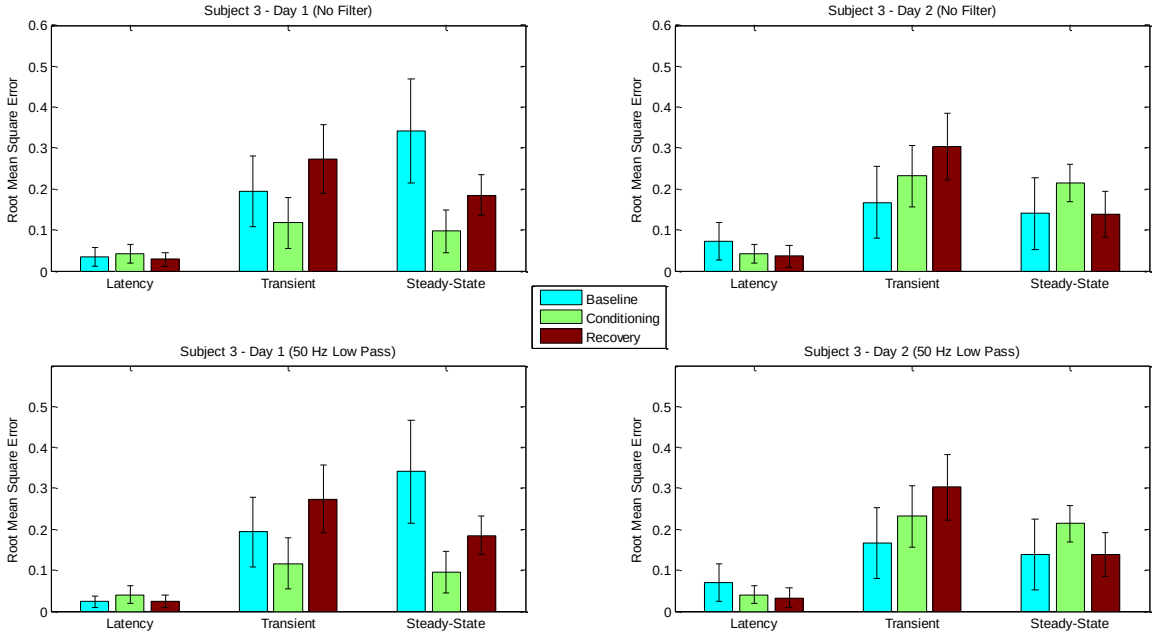

Figure 3.57 RMSE of Subject 3 sectioned disconjugate mean movements. Top row consists data that has not been filtered and the bottom row is of data with a 50Hz low pass. The left column is data from day 1 and the right column is of day 2.

CHAPTER 4

DISCUSSION

The data shows statistical differences on the vergence ramp eye movements due to the short-term modification performed on all ten subject on day 1, however no statistical changes have been observed on day 2. This dissipation of learning effect can be attributed to motor memory form day 1 to day 2. The data additionally indicates no statistical differences between the movements of the left eye and the right eye.

The same modification was performed on day 1 and day 2. Though the same modification was performed on both days there many different factors affecting the subject. The subject on day 1 was a truly naïve subject and more susceptible to modification compared to the same subject on day 2. Day 1 was also unique in nature where an average individual is never required to concentrate hard on a set of targets for an entire hour. This extended task, which may cause fatigue in some of the subjects, would also enable them to learn. These factors of unfamiliarity to the modification would make the day 1 results more reliable than the results from day 2. This is one of the reasons why a statistical difference was noticed between the modification and recovery stages in day 1 for the mean disconjugate movements and in the transient phase. The modification on day 2 yielded no change due to the familiarity to the modification and motor memory.

The left and right eye movement traces are accurate on average, however the precision is greatly diminished when compared to the disconjugate movements as seen in Section 3.1. Regardless of this decrease in precision, the precision of the disconjugate

movements are not affected. These traces show that regardless of the individual movements of the eyes, both eyes work together to form one comprehensive image and work together to precisely follow a given target.

The mean disconjugate movements of the three stages in Section 3.2 provide an in depth view of the movement of the eyes. The first clear observation is the minute fluctuations in the movement as the eyes followed the target in all three stages of movement. These fluctuations show the influence of the feedback system that is constantly trying to adjust the angle of the eyes to precisely follow the target. Another explanation for the fluctuations could be the delay in the neural responses. Looking at the mean disconjugate movements of day 2, it can be seen that the number of oscillations drastically increase following the start of the ramp as compared to when the stimulus is stationary. This trend would be logical since there is no movement initially, hence no adjustment of fixation is needed, and thus the feedback system would not be active. Since the number of oscillations was consistent in all stages for all the subjects, it is clear that the feedback system is not affected due to the short-term modification.

A further analysis was conducted on the mean disconjugate movements by taking the RMSE of the three different stages. The three stages were latency, transient, and steady-state which lasted from 0 sec to 0.6 sec, 0.6 sec to 1.3 sec, and 2 sec to 4.5 sec, respectively. The Latency section contains the data from the start of the movement, where the target is stationary, to the latency component. The transient section contains the initial movement of the eyes after the target begins to move. This section also contains the maximum velocity of the movement. The final stage of stead-state is the ramp portion of the movement.

Nine out of the ten subjects have a consistent trend where the conditioning mean movement is closer in position to the position of the stimulation, as compared to the mean movements of baseline and recovery. The reduction of error present in the conditioning mean disconjugate movement is reflected slightly in the mean disconjugate RMSE and greatly in the RMSE of the transient phase of the mean disconjugate movement. This trend is only present in day 1, and as previously stated, motor learning can be the cause for the dissipation of this effect on day 2. The preprogrammed component of vergence along with the feedback component have shown to be active. Though the feedback component is not effected by modification, the preprogrammed seemed to be effected by the slightest and shortest modification. The data also shows that motor memory plays an important role in vergence ramp movements. Past studies have shown that the preprogrammed and feedback components are effected after long-term modification and this study shows that the preprogrammed component could have been effected by the short-term modification. The results in this study are also consistent with the smooth pursuit literature which shows a modification in the transient phase due to modification (Carl, 1987).

Another trend that is consistent with all subjects is the lag between the position of the target and the fixation of the eyes. This can be best seen in Figure 3.22. The means at all three stages of baseline, conditioning, and recovery are precise for all subjects and in the figures, they are “below” the stimulation (black) line. This clearly shows how the eyes are slightly lagging behind the stimulation.

The general trend in mean conjugate velocities can also be summed up by Figure 3.22. There was an initial velocity of 0 deg/sec, which quickly rose to max velocity of the

movement (transient movement), and later the velocity oscillated around 1 deg/sec. The 1 deg/sec is important because that is the speed of the ramp. The ramp speed is 0.5 deg/sec monocular, which would translate to a 1 deg/sec binocular movement. Disconjugate measurements are binocular measurements in this study.

The isolated left and right eye mean movements have no trends that are uniform on an overall basis, however the trends were subject dependent. The movements of some subjects were precise, some were accurate, and others were precise and accurate. An RMSE analysis on the left and right eye data presented in Figure 3.42 shows disparity of the individual eyes form the position of the target. No differences can be identified between the movements of the left and right eye movements by looking at their RMSEs. Various statistical analysis, including a repeated-measures ANOVA and two-tailed t-test, have been performed to show an absence of statistical significance. The ANOVA analysis yielded no statistical difference between the left eye movements of the three stages on day 1. The same results were produced with the left and right eye movements within themselves on both day 1 and 2. A two-tailed paired t-test comparing the baseline left eye movements with the right eye movements on day 1 also yielded no statistical difference. The same test was repeated five more times, one for each stage of baseline, conditioning, and recovery, and one for each of the two days.

Finally, Section 3.4 shows the importance of the 10Hz low pass filter used to analyze the data for all ten subjects. When compared to data that has not been filtered and data that has been put through a 50HZ low pass filter, there were no difference between the RMSE of the disconjugate mean movements and the RMSE of the sectioned means of Subject 3.

CHAPTER 5

CONCLUSION

This study indicates that there is a significant effect on the vergence system due to a short term modification in the convergent direction in day 1, however no significant effect is seen on day 2. Additionally, no statistical differences have been observed between the individual movements of the left and right eyes.

The influence on the vergence system on day 1 in these ten healthy and binocularly normal subjects indicates that a short term modification is effective. This modification can produce greater results for someone suffering from convergence insufficiency (CI) or other neurological deficiencies that effect eye movements. The motor memory was potentially in effect on day two after the initial modification, however this effect might not be as immediate for those suffering from CI. Further studies can also be performed in the divergent direction in both binocularly normal and CI subjects to observe the effects of this short-term modification. The effects of slow ramps on fast ramps could also be studied in the future.

A stronger study can be formed without the study limitations of time, subjects, and instrumentations. A greater allotted time would allow for a further analysis of the collected data. An analysis of the fast ramp data could yield important information about the actions performed by the individual eyes. A greater number of subject would aid in formulating better statistical analysis of the data. Finally, an increase in the precision of the instrumentation that can measure changes smaller than 0.3 degrees per second could yield more insight into the similarities or differences between the left and right eyes. This

would allow more opportunities to further the studies on left eye versus right eye. An additional analysis on the cross correlation between the movements of both eyes would also yield more information in the future.

APPENDIX

MATLAB SOURCE CODE

This section contains the code of all the preprocessing of data, which formatted collected data to fit MATLAB® arrays, extraction of necessary data, and the analysis of that data.

A.1 Preprocessing Data

```
%
PreprocessData1(NumLoops,NumStimPerLoop,CalibTargetDegrees,CalibOption,
CalibLength)
function
PreprocessData1(NumLoops,NumStimPerLoop,CalibTargetDegrees,CalibOption,
CalibLength)
 %%(# of cals,# of times log fiel is called,[3 5 7],'Type' or 'Click',
cal
 %%time (sec))
close all
PreprocessOption = 0;

while PreprocessOption ~= 4

    PreprocessOption = menu('What would you like to do?','Import
Data','Preprocess Data','Make Variables','Done');

    if PreprocessOption == 1
        [FileName,PathName] = ...
            uigetfile('*.*','Select the raw data file');
        [data] = txt2mat([PathName,FileName],0);
        save variables1.mat
    end

    if PreprocessOption == 2
        load variables1.mat;
        NanInd = isnan(data(:,1));
        x = find(NanInd == 1);
        nanbeg2 = [];
        nanend2 = [];
        for i =2:length(x)
            if x(i) - x(i-1) > 2
                nanbeg = x(i-1)+1;
                nanbeg2 = [nanbeg2;nanbeg];
                nanend = x(i)-1;
                nanend2 = [nanend2;nanend];
            end
        end
        tmp = [nanbeg2,nanend2];
```

```
CalibInd = 1;
ExpInd = 1;
for i = 1:length(nanbeg2)
    tmp = data(nanbeg2(i):nanend2(i));
    if length(tmp) < CalibLength*500+2
        data2calib(CalibInd,:) = tmp;
        CalibInd = CalibInd+1;
    else
        data2exp(ExpInd,:) = tmp;
        ExpInd = ExpInd+1;
    end
end

Data2calibSize = size(data2calib); j=1;
for i = 1:6:Data2calibSize(1)
    data3calib{j} = (data2calib(i:i+5,:)')';
    j=j+1;
end
Data2expSize = size(data2exp); j=1;
for i = 1:6:Data2expSize(1)
    data3exp{j} = (data2exp(i:i+5,:)')';
    j=j+1;
end

NumCal = length(data3calib)./(numel(CalibTargetDegrees)*2);
for i=1:NumCal
    if numel(CalibTargetDegrees) == 2
        LeftEyeCalib{1,NumCal} = data3calib{1+((i-1)*6)};
        LeftEyeCalib{2,NumCal} = data3calib{2+((i-1)*6)};

        RightEyeCalib{1,NumCal} = data3calib{3+((i-1)*6)};
        RightEyeCalib{2,NumCal} = data3calib{4+((i-1)*6)};
    end

    if numel(CalibTargetDegrees) == 3
        LeftEyeCalib{1,i} = data3calib{1+((i-1)*6)}(2,:);
        LeftEyeCalib{2,i} = data3calib{2+((i-1)*6)}(2,:);
        LeftEyeCalib{3,i} = data3calib{3+((i-1)*6)}(2,:);

        RightEyeCalib{1,i} = data3calib{4+((i-1)*6)}(1,:);
        RightEyeCalib{2,i} = data3calib{5+((i-1)*6)}(1,:);
        RightEyeCalib{3,i} = data3calib{6+((i-1)*6)}(1,:);
    end

    if numel(CalibTargetDegrees) == 4

        LeftEyeCalib{1}(NumCal) = data3calib{1+((i-1)*6)};
        LeftEyeCalib{2}(NumCal) = data3calib{2+((i-1)*6)};
        LeftEyeCalib{3}(NumCal) = data3calib{3+((i-1)*6)};
        LeftEyeCalib{4}(NumCal) = data3calib{4+((i-1)*6)};

        RightEyeCalib{1}(NumCal) = data3calib{5+((i-1)*6)};
        RightEyeCalib{2}(NumCal) = data3calib{6+((i-1)*6)};
        RightEyeCalib{3}(NumCal) = data3calib{7+((i-1)*6)};
        RightEyeCalib{4}(NumCal) = data3calib{8+((i-1)*6)};
```

```
            end

            if numel(CalibTargetDegrees) == 5
                LeftEyeCalib{1}(NumCal) = data3calib{1+((i-1)*6)};
                LeftEyeCalib{2}(NumCal) = data3calib{2+((i-1)*6)};
                LeftEyeCalib{3}(NumCal) = data3calib{3+((i-1)*6)};
                LeftEyeCalib{4}(NumCal) = data3calib{4+((i-1)*6)};
                LeftEyeCalib{5}(NumCal) = data3calib{5+((i-1)*6)};

                RightEyeCalib{1}(NumCal) = data3calib{6+((i-1)*6)};
                RightEyeCalib{2}(NumCal) = data3calib{7+((i-1)*6)};
                RightEyeCalib{3}(NumCal) = data3calib{8+((i-1)*6)};
                RightEyeCalib{4}(NumCal) = data3calib{9+((i-1)*6)};
                RightEyeCalib{5}(NumCal) = data3calib{10+((i-1)*6)};

            end
        end
        LEpt1 = zeros(1,NumCal)+1;
        LEpt2 = zeros(1,NumCal)+500;
        REpt1 = zeros(1,NumCal)+1;
        REpt2 = zeros(1,NumCal)+500;
        for Cal=1:NumCal
            ReselectCalibMean = 0;
            hFig=figure(Cal);
            screen_size = get(0, 'ScreenSize');
            set(hFig,'Position',[0 0 screen_size(3) screen_size(4)])
            while ReselectCalibMean ~= 3
                CalibSP1 = subplot(2,2,1);
                plot([(LeftEyeCalib{1,Cal});(LeftEyeCalib{2,Cal});...
                    (LeftEyeCalib{3,Cal})]');
                title('Left Eye')
                ylim([-5 5])
                ylabel('Volts')
                xlabel('Samples')

                CalibSP2 = subplot(2,2,2);
                plot([(RightEyeCalib{1,Cal});(RightEyeCalib{2,Cal});...
                    (RightEyeCalib{3,Cal})]');
                title('Right Eye')
                ylim([-5 5])
                ylabel('Volts')
                xlabel('Samples')

                subplot(2,2,3)
                cla
                LE_MeanCalibVolts(Cal,:) =
[mean(LeftEyeCalib{1,Cal}(LEpt1(Cal)...

:LEpt2(Cal))),mean(LeftEyeCalib{2,Cal}(LEpt1(Cal)...

:LEpt2(Cal))),mean(LeftEyeCalib{3,Cal}(LEpt1(Cal)...
                    :LEpt2(Cal)))];

                [LE_R,LE_PVAL] =
corrcoef(LE_MeanCalibVolts(Cal,:),CalibTargetDegrees);
                LE_CorrCoef(Cal,:) = LE_R(1,2);
```

```
                LE_CalibFit(Cal,:) =
polyfit(LE_MeanCalibVolts(Cal,:),CalibTargetDegrees,1);
                f =
polyval(LE_CalibFit(Cal,:),LE_MeanCalibVolts(Cal,:));

                plot(LE_MeanCalibVolts(Cal,:),CalibTargetDegrees);
                hold on
                plot(LE_MeanCalibVolts(Cal,:),f,'r');
                R = mat2str(LE_CorrCoef(Cal));
                m = mat2str(LE_CalibFit(Cal,1));
                b = mat2str(LE_CalibFit(Cal,2));
                text(LE_MeanCalibVolts(Cal,3),2,{['R^2 = ',R(1:5)];['y
= ',...
                    m(1:4),'x + ',b(1:4)]})
                ylabel('Degrees')
                xlabel('Volts')

                subplot(2,2,4)
                cla
                RE_MeanCalibVolts(Cal,:) =
[mean(RightEyeCalib{1,Cal}(REpt1(Cal):...

REpt2(Cal))),mean(RightEyeCalib{2,Cal}(REpt1(Cal):...

REpt2(Cal))),mean(RightEyeCalib{3,Cal}(REpt1(Cal):...
                    REpt2(Cal)))];
                [RE_R,RE_PVAL] =
corrcoef(RE_MeanCalibVolts(Cal,:),CalibTargetDegrees);
                RE_CorrCoef(Cal,:) = RE_R(1,2);
                RE_CalibFit(Cal,:) =
polyfit(RE_MeanCalibVolts(Cal,:),CalibTargetDegrees,1);
                f =
polyval(RE_CalibFit(Cal,:),RE_MeanCalibVolts(Cal,:));

                plot(RE_MeanCalibVolts(Cal,:),CalibTargetDegrees);
                hold on
                plot(RE_MeanCalibVolts(Cal,:),f,'r');
                R = mat2str(RE_CorrCoef(Cal));
                m = mat2str(RE_CalibFit(Cal,1));
                b = mat2str(RE_CalibFit(Cal,2));
                text(RE_MeanCalibVolts(Cal,3),2,{['R^2 = ',R(1:5)];...
                    ['y = ',m(1:4),'x + ',b(1:4)]})
                ylabel('Degrees')
                xlabel('Volts')


                ReselectCalibMean = menu('Reselect data to calculate
calibration mean','Left Eye','Right Eye','Done');
                if strcmp(CalibOption,'Click') == 1
                    if ReselectCalibMean ==1
                        gcf(CalibSP1)
                        [TmpX,TmpY] = ginput(2);
                        LEpt1(Cal) = round(TmpX(1));
                        LEpt2(Cal) = round(TmpX(2));
                        LE_CalibBegEnd(Cal,:) = [LEpt1,LEpt2];
                    end
```

```
                    if ReselectCalibMean ==2
                        gcf(CalibSP2)
                        [TmpX,TmpY] = ginput(2);
                        REpt1(Cal) = round(TmpX(1));
                        REpt2(Cal) = round(TmpX(2));
                        RE_CalibBegEnd(Cal,:) = [REpt1,REpt2];
                    end
                end
                if strcmp(CalibOption,'Type') == 1
                    if ReselectCalibMean == 1
                        TmpX = inputdlg('Input new range for
calibration','New Calibration Range');
                        TmpX = str2num(TmpX{:});
                        LEpt1(Cal) = TmpX(1);
                        LEpt2(Cal) = TmpX(2);
                        LE_CalibBegEnd(Cal,:) = [LEpt1,LEpt2];
                    end

                    if ReselectCalibMean ==2
                        TmpX = inputdlg('Input new range for
calibration','New Calibration Range');
                        TmpX = str2num(TmpX{:});
                        REpt1(Cal) = TmpX(1);
                        REpt2(Cal) = TmpX(2);
                        RE_CalibBegEnd(Cal,:) = [REpt1,REpt2];
                    end
                end
            end
        end
        SPnum = 1;
        figure
        for Cal = 1:NumCal
            subplot(NumCal,2,SPnum)
            f = polyval(LE_CalibFit(Cal,:),LE_MeanCalibVolts(Cal,:));
            plot(LE_MeanCalibVolts(Cal,:),CalibTargetDegrees);
            hold on
            plot(LE_MeanCalibVolts(Cal,:),f,'r');
            R = mat2str(LE_CorrCoef(Cal));
            m = mat2str(LE_CalibFit(Cal,1));
            b = mat2str(LE_CalibFit(Cal,2));
            text(LE_MeanCalibVolts(Cal,3),2,{['R^2 = ',R(1:5)];['y =
',m(1:4),'x + ',b(1:4)]})
            ylabel('Degrees')
            xlabel('Volts')

            SPnum = SPnum+1;

            subplot(NumCal,2,SPnum)
            f = polyval(RE_CalibFit(Cal,:),RE_MeanCalibVolts(Cal,:));
            plot(RE_MeanCalibVolts(Cal,:),CalibTargetDegrees);
            hold on
            plot(RE_MeanCalibVolts(Cal,:),f,'r');
            R = mat2str(RE_CorrCoef(Cal));
            m = mat2str(RE_CalibFit(Cal,1));
            b = mat2str(RE_CalibFit(Cal,2));
```

```
            text(RE_MeanCalibVolts(Cal,3),2,{['R^2 = ',R(1:5)];['y =
',m(1:4),'x + ',b(1:4)]})
            ylabel('Degrees')
            xlabel('Volts')

            SPnum = SPnum+1;
        end

        CalSelectOptions = menu('Select option for calibration
use','Use Loop Cal','Use Best Cal','Manually Select Cal','Keep All');
        if CalSelectOptions == 2
            BestCal_LE = menu('Which loop provided the best LEFT EYE
calibration','1','2','3','4','5','6');
            BestCal_RE = menu('Which loop provided the best RIGHT EYE
calibration','1','2','3','4','5','6');
        end

        if CalSelectOptions == 4
                RE_Fit = RE_CalibFit;
                LE_Fit = LE_CalibFit;
        end

        for Cal = 1:NumLoops

            if CalSelectOptions == 1
                RE_Fit(Cal,:) = RE_CalibFit(Cal,:);
                LE_Fit(Cal,:) = LE_CalibFit(Cal,:);
            end

            if CalSelectOptions == 2
                RE_Fit(Cal,:) = RE_CalibFit(BestCal_LE,:);
                LE_Fit(Cal,:) = LE_CalibFit(BestCal_RE,:);
            end

            if CalSelectOptions == 3
                UseWhichCal_LE = menu(['Use which RE loop cal from loop
',num2str(Cal)],'1','2','3','4','5','6');
                UseWhichCal_RE = menu(['Use which LE loop cal from loop
',num2str(Cal)],'1','2','3','4','5','6');

                RE_Fit(Cal,:) = RE_CalibFit(UseWhichCal_RE,:);
                LE_Fit(Cal,:) = LE_CalibFit(UseWhichCal_LE,:);
            end
        end
        save variables2.mat
    end

    if PreprocessOption == 3
        load variables1.mat
        load variables2.mat

[stringdata]=textread([PathName,FileName],'%s','delimiter','\n','whites
pace','');
        Index = 1;
        for i = 1:6:length(nanbeg2)
```

```
            for j=1:6
                tmp = data(nanbeg2(j+i-1):nanend2(j+i-1));
                data3total{Index}(j,:) = tmp;
            end
            Index = Index+1;
        end
        j=1;
        k=1;
        for i=1:(length(nanbeg2))
            tmp = data(nanbeg2(i):nanend2(i));
                if rem(i,6) == 1
                    if
rem(i,(NumStimPerLoop+numel(CalibTargetDegrees)*2))==0
                        k=k+1;
                    end
                    eval(sprintf('%s%s%i =
data3total{%d};',FileName(1:3),stringdata{nanbeg2(i)-6},k,(i-1)/6+1));
                    Vars{j}=stringdata{nanbeg2(i)-6};
                    j=j+1;
                end
        end
        save('StimData', '-regexp', ['^' FileName(1:3)])
        StimVars = load('StimData.mat');
        save variables.mat
        for m=1:length(data3total)
                StimVarsRE.(Vars{m}) = StimVars.([FileName(1:3)
Vars{m}]); %* RE_Fit(m,1) + RE_Fit(m,2);
                StimVarsLE.(Vars{m}) = StimVars.([FileName(1:3)
Vars{m}]);%* LE_Fit(m,1) + LE_Fit(m,2);
        end
%         for i=1:length(data3total)
%             if length(StimVarsRE.(Vars{i})) > 501
%                 for k=1:NumLoops
%                     for j=1:NumStimPerLoop
%                         StimVarsRE.(Vars{j+NumStimPerLoop*(1-
k)})=StimVarsRE.(Vars{j+NumStimPerLoop*(1-k)})*RE_Fit(k,1)+RE_Fit(k,2);
%                     end
%                 end
%             end
%         end

    end
    save([FileName(1:3) 'variables.mat']);
    close all
end
end
```

A.2 Extracting Data and Producing Mean Movements

```
% This code will process the raw data in a 2Point Calibration method
% and produces means of L, R, and Total eye movements
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%% Loading of RAW Data and inputting subject's innitials
Innitials = input('Enter Subject"s innitials (ex:"CXY"):');
Date = input('Enter date of experiment (ex:"mmddyyyy"):');
load(['C:\Dropbox\Data\RawExtractedData\' Innitials '_Raw_' Date
'.mat'])
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%% Calibration of Data with 2 point binocular calibration
% This method will calibrate each stimulation WRT the beginning and
final
% sets of data
% The data will begin at 3deg and end at 5deg
eval(['T1 = ' Innitials '_Raw_' Date '_T1;'])
eval(['T2 = ' Innitials '_Raw_' Date '_T2;'])
eval(['C = ' Innitials '_Raw_' Date '_C;'])
% Obtaining the mean of first and last 250 points for T1
for i=1:20
mean3ofR = mean(T1{i}(2,1:200));
mean3ofL = mean(T1{i}(1,1:251));
mean5ofR = mean(T1{i}(2,2300:2500));
mean5ofL = mean(T1{i}(1,2300:2500));

%m = (y2-y1)/(x2=x1)
mR = (mean5ofR - mean3ofR)/(5-3);
mL = (mean5ofL - mean3ofL)/(5-3);

for j= 1:2500
T1_2Cal{i}(1,j)=((T1{i}(1,j)- mean3ofL)/mL)+3;
T1_2Cal{i}(2,j)=((T1{i}(2,j)- mean3ofR)/mR)+3;
end
end

% Obtaining the mean of first and last 250 points for T2
for i=1:20
mean3ofL = mean(T2{i}(1,1:200));
mean3ofR = mean(T2{i}(2,1:251));
mean5ofL = mean(T2{i}(1,2300:2500));
mean5ofR = mean(T2{i}(2,2300:2500));

%m = (y2-y1)/(x2=x1)
mR = (mean5ofR - mean3ofR)/(5-3);
mL = (mean5ofL - mean3ofL)/(5-3);

for j= 1:2500
%y-y1 = m(x-x1)
%((y-y1)/m)+x1 = x
T2_2Cal{i}(1,j)=((T2{i}(1,j)- mean3ofL)/mL)+3;
T2_2Cal{i}(2,j)=((T2{i}(2,j)- mean3ofR)/mR)+3;
end
end
```

```
% Obtaining the mean of first and last 250 points for C
for i=1:120
mean3ofL = mean(C{i}(1,1:200));
mean3ofR = mean(C{i}(2,1:251));
mean5ofL = mean(C{i}(1,2300:2500));
mean5ofR = mean(C{i}(2,2300:2500));

%m = (y2-y1)/(x2=x1)
mR = (mean5ofR - mean3ofR)/(5-3);
mL = (mean5ofL - mean3ofL)/(5-3);

for j= 1:2500
%y-y1 = m(x-x1)
%((y-y1)/m)+x1 = x
C_2Cal{i}(1,j)=((C{i}(1,j)- mean3ofL)/mL)+3;
C_2Cal{i}(2,j)=((C{i}(2,j)- mean3ofR)/mR)+3;
end
end

%% The Following reajdustes the data and provides options for removing
data

%Adjusting the data to go from 3 degrees to 5 degrees

for i = 1:20
T1_2Cal{i}(1,:) = T1_2Cal{i}(1,:) - mean(T1_2Cal{i}(1,1:75)) + 3;
T1_2Cal{i}(2,:) = T1_2Cal{i}(2,:) - mean(T1_2Cal{i}(2,1:75)) + 3;
end

for i = 1:20
T2_2Cal{i}(1,:) = T2_2Cal{i}(1,:) - mean(T2_2Cal{i}(1,1:100)) + 3;
T2_2Cal{i}(2,:) = T2_2Cal{i}(2,:) - mean(T2_2Cal{i}(2,1:100)) + 3;
end

for i = 1:120
C_2Cal{i}(1,:) = C_2Cal{i}(1,:) - mean(C_2Cal{i}(1,1:100)) + 3;
C_2Cal{i}(2,:) = C_2Cal{i}(2,:) - mean(C_2Cal{i}(2,1:100)) + 3;
end

%% Plotting Code for finding bad data and removing bad data
for i=1:20
    hold on
    subplot(3,1,1)
    plot(T1_2Cal{i}(1,:))
    axis([0 2500 0 8])
    title(i); xlabel('Baseline Left');
    subplot(3,1,2)
    plot(T1_2Cal{i}(2,:))
    axis([0 2500 0 8])
    title(i); xlabel('Baseline Right');
    subplot(3,1,3)
    plot(T1_2Cal{i}(1,:)+T1_2Cal{i}(2,:))
    axis([0 2500 3 13])
    title(i); xlabel('Baseline Sum');
```

```
    pause
    clf
end

for i=1:20
    hold on
    subplot(3,1,1)
    plot(T2_2Cal{i}(1,:))
    axis([0 2500 0 8])
    title(i); xlabel('Recovery Left');
    subplot(3,1,2)
    plot(T2_2Cal{i}(2,:))
    axis([0 2500 0 8])
    title(i); xlabel('Recovery Right');
    subplot(3,1,3)
    plot(T2_2Cal{i}(1,:)+T2_2Cal{i}(2,:))
    axis([0 2500 3 13])
    title(i); xlabel('Recovery Sum');
    pause
    clf
end

for i=1:120
    hold on
    subplot(3,1,1)
    plot(C_2Cal{i}(1,:))
    axis([0 2500 0 8])
    title(i); xlabel('Conditioning Left');
    subplot(3,1,2)
    plot(C_2Cal{i}(2,:))
    axis([0 2500 0 8])
    title(i); xlabel('Conditioning Right');
    subplot(3,1,3)
    plot(C_2Cal{i}(1,:)+C_2Cal{i}(2,:))
    axis([0 2500 3 13])
    title(i); xlabel('Conditioning Sum');
    pause
    clf
end

T1removeLeft = input('Left eye data to remove from Baseline (ex:
[1,2,3]):');
T1removeRight = input('Right eye data to remove from Baseline (ex:
[1,2,3]):');
T2removeLeft = input('Left eye data to remove from Recovery (ex:
[1,2,3]):');
T2removeRight = input('Right eye data to remove from Recovery (ex:
[1,2,3]):');
CremoveLeft = input('Left eye data to remove from Conditioning (ex:
[1,2,3]):');
CremoveRight = input('Right eye data to remove from Conditioning (ex:
[1,2,3]):');

Remove = [T1removeLeft];
for i = Remove
    T1_2Cal{1,i}(1,:) = NaN;
```

```
   end
Remove = [T1removeRight];
for i = Remove
    T1_2Cal{1,i}(2,:) = NaN;
end

Remove = [T2removeLeft];
for i = Remove
    T2_2Cal{1,i}(1,:) = NaN;
end
Remove = [T2removeRight];
for i = Remove
    T2_2Cal{1,i}(2,:) = NaN;
end

Remove = [CremoveLeft];
for i = Remove
    C_2Cal{1,i}(1,:) = NaN;
end
Remove = [CremoveRight];
for i = Remove
    C_2Cal{1,i}(2,:) = NaN;
end

%% Filtring of Data

% Initial Test
for j = 1:2
for i = 1:20
sampling_rate = 500;
T = 1/sampling_rate;
duration = (length(T1_2Cal{1}))/sampling_rate;
% figure(1)
% f_axis = 0:1/duration:sampling_rate-1/duration;
% plot(f_axis,abs(fft(AJB_2Cal_11112013_T1{i}(j,:))))
fc = 10;
fn = 2*fc/(sampling_rate);
[b,a] = butter(6,fn,'low');
filtered = filtfilt(b,a,T1_2Cal{i}(j,:)')';
fill1{i}(j,:) = filtered;
end
end

% Final Test
for j = 1:2
for i = 1:20
sampling_rate = 500;
T = 1/sampling_rate;
duration = (length(T2_2Cal{1}))/sampling_rate;
fc = 10;
fn = 2*fc/(sampling_rate);
[b,a] = butter(6,fn,'low');
filtered = filtfilt(b,a,T2_2Cal{i}(j,:)')';
fill2{i}(j,:) = filtered;
end
end
```

```
% Conditioning
for j = 1:2
for i = 1:120
sampling_rate = 500;
T = 1/sampling_rate;
duration = (length(C_2Cal{1}))/sampling_rate;
fc = 10;
fn = 2*fc/(sampling_rate);
[b,a] = butter(6,fn,'low');
filtered = filtfilt(b,a,C_2Cal{i}(j,:)')';
fill3{i}(j,:) = filtered;
end
end

%% Calculation of the Means
% Initial Test

for j = 1:20
    fillL(j,:) = fill1{j}(1,:);
    fillR(j,:) = fill1{j}(2,:);
end

fillL(logical(sum(fillL~=fillL,2)),:)=[];
fillR(logical(sum(fillR~=fillR,2)),:)=[];

for i = 1:2500
    fillLmeanB(1,i) = mean(fillL(:,i));
    fillRmeanB(1,i) = mean(fillR(:,i));
end

BLstd = std(fillL);
BRstd = std(fillR);

% Plotting L vs R data for Baseline seperate plot
for i = 1:20
figure(2)
subplot(2,1,1)
plot(0:T:4.998,fill1{i}(1,:))
hold on
plot(0:T:4.998,fillLmeanB,'r','LineWidth',2)
title('Initial Left Eye Movements')
axis([0 5 2 7])
subplot(2,1,2)
plot(0:T:4.998,fill1{i}(2,:))
hold on
plot(0:T:4.998,fillRmeanB,'r','LineWidth',2)
title('Initial Right Eye Movements')
axis([0 5 2 7])
end

% Plotting L vs R data for Baseline combined plot
for i = 1:20
figure(1)
subplot(3,3,1)
```

```
plot(0:T:4.998,fill1{i}(1,:))
hold on
plot(0:T:4.998,fillLmeanB,'r','LineWidth',2)
title('Baseline Left Eye Movements')
axis([0 5 2 7])
subplot(3,3,2)
plot(0:T:4.998,fill1{i}(2,:))
hold on
plot(0:T:4.998,fillRmeanB,'r','LineWidth',2)
title('Baseline Right Eye Movements')
axis([0 5 2 7])
end

for j = 1:20
    filltotal(j,:) = fill1{j}(1,:) + fill1{j}(2,:);
end

filltotal(logical(sum(filltotal~=filltotal,2)),:)=[];

for i = 1:2500
    fillmean1(1,i) = mean(filltotal(:,i));
end

BTstd = std(filltotal);

% Baseline Total individual plot
for i = 1:20
figure(3)
plot(0:T:4.998,fill1{i}(1,:)+fill1{i}(2,:))
hold on
plot(0:T:4.998,fillmean1,'r','LineWidth',2)
axis([0 5 4 12])
title('Total Baseline Vergence Movement')
end

% Baseline Total Combined Plot
for i = 1:20
figure(1)
subplot(3,3,3)
plot(0:T:4.998,fill1{i}(1,:)+fill1{i}(2,:))
hold on
plot(0:T:4.998,fillmean1,'r','LineWidth',2)
axis([0 5 4 12])
title('Total Baseline Vergence Movement')
end

% Final Test

for j = 1:20
    fillL(j,:) = fill2{j}(1,:);
    fillR(j,:) = fill2{j}(2,:);
end

fillL(logical(sum(fillL~=fillL,2)),:)=[];
fillR(logical(sum(fillR~=fillR,2)),:)=[];
```

```
for i = 1:2500
    fillLmeanR(1,i) = mean(fillL(:,i));
    fillRmeanR(1,i) = mean(fillR(:,i));
end

RLstd = std(fillL);
RRstd = std(fillR);

% Plotting L vs R data for Recovery seperate plot
for i = 1:20
figure(5)
subplot(2,1,1)
plot(0:T:4.998,fill2{i}(1,:))
hold on
plot(0:T:4.998,fillLmeanR,'r','LineWidth',2)
title('Recovery Left Eye Movements')
axis([0 5 2 7])
subplot(2,1,2)
plot(0:T:4.998,fill2{i}(2,:))
hold on
plot(0:T:4.998,fillRmeanR,'r','LineWidth',2)
title('Recovery Right Eye Movements')
axis([0 5 2 7])
end

% Plotting L vs R data for Recovery seperate plot
for i = 1:20
figure(1)
subplot(3,3,7)
plot(0:T:4.998,fill2{i}(1,:))
hold on
plot(0:T:4.998,fillLmeanR,'r','LineWidth',2)
title('Recovery Left Eye Movements')
axis([0 5 2 7])
subplot(3,3,8)
plot(0:T:4.998,fill2{i}(2,:))
hold on
plot(0:T:4.998,fillRmeanR,'r','LineWidth',2)
title('Recovery Right Eye Movements')
axis([0 5 2 7])
end

for j = 1:20
    filltotal(j,:) = fill2{j}(1,:) + fill2{j}(2,:);
end

filltotal(logical(sum(filltotal~=filltotal,2)),:)=[];

for i = 1:2500
    fillmean2(1,i) = mean(filltotal(:,i));
end

RTstd = std(filltotal);
```

```
% Recovery Total Seperate Plot
for i = 1:20
figure(6)
plot(0:T:4.998,fill2{i}(1,:) + fill2{i}(2,:))
hold on
plot(0:T:4.998,fillmean2,'r','LineWidth',2)
title('Total Recovery Vergence Movement')
end

% Recovery Total Combined Plot
for i = 1:20
figure(1)
subplot(3,3,9)
plot(0:T:4.998,fill2{i}(1,:) + fill2{i}(2,:))
hold on
plot(0:T:4.998,fillmean2,'r','LineWidth',2)
axis([0 5 4 12])
title('Total Recovery Vergence Movement')
end

% Conditioning
Slow = [2,10,14,23,29];
Slow = [2,10,14,23,29,Slow+30,Slow+60,Slow+90];
Fast =
[1,3,4,5,6,7,8,9,11,12,13,15,16,17,18,19,20,21,22,24,25,26,27,28,30];
Fast =
[1,3,4,5,6,7,8,9,11,12,13,15,16,17,18,19,20,21,22,24,25,26,27,28,...
    30,Fast+30,Fast+60,Fast+90];
i = 1;
for j = Slow
    fillLslow(i,:) = fill3{j}(1,:);
    fillRslow(i,:) = fill3{j}(2,:);
    i = i+1;
end
i = 1;
for j = Fast
    fillLfast(i,:) = fill3{j}(1,:);
    fillRfast(i,:) = fill3{j}(2,:);
    i = i+1;
end

fillStotal = fillLslow + fillRslow;

fillStotal(logical(sum(fillStotal~=fillStotal,2)),:)=[];

fillLslow(logical(sum(fillLslow~=fillLslow,2)),:)=[];
fillRslow(logical(sum(fillRslow~=fillRslow,2)),:)=[];
fillLfast(logical(sum(fillLfast~=fillLfast,2)),:)=[];
fillRfast(logical(sum(fillRfast~=fillRfast,2)),:)=[];

for i = 1:2500
    fillLmeanslowC(1,i) = mean(fillLslow(:,i));
    fillLmeanfast(1,i) = mean(fillLfast(:,i));
    fillRmeanslowC(1,i) = mean(fillRslow(:,i));
    fillRmeanfast(1,i) = mean(fillRfast(:,i));
end
```

```
CSLstd = std(fillLslow);
CFLstd = std(fillLfast);
CSRstd = std(fillRslow);
CFRstd = std(fillRfast);
CSTstd = std(fillStotal);

fillmean3 = fillLmeanslowC+fillRmeanslowC;

figure(8)
for i = Slow
    subplot(2,2,1)
    plot(0:T:4.998,fill3{i}(1,:))
    axis([0 5 2 7])
    hold on
    plot(0:T:4.998,fillLmeanslowC,'r','LineWidth',2)
    title('Slow Stimulus Left Eye')
end

for i = Slow
    subplot(2,2,2)
    plot(0:T:4.998,fill3{i}(2,:))
    axis([0 5 2 7])
    hold on
    plot(0:T:4.998,fillRmeanslowC,'r','LineWidth',2)
    title('Slow Stimulus Right Eye')
end

for i = Fast
    subplot(2,2,3)
    plot(0:T:4.998,fill3{i}(1,:))
    axis([0 5 2 7])
    hold on
    plot(0:T:4.998,fillLmeanfast,'r','LineWidth',2)
    title('Fast Stimulus Left Eye')
end

for i = Fast
    subplot(2,2,4)
    plot(0:T:4.998,fill3{i}(2,:))
    axis([0 5 2 7])
    hold on
    plot(0:T:4.998,fillRmeanfast,'r','LineWidth',2)
    title('Fast Stimulus Right Eye')
end

for i = Slow
    figure(1)
    subplot(3,3,4)
    plot(0:T:4.998,fill3{i}(1,:))
    axis([0 5 2 7])
    hold on
    plot(0:T:4.998,fillLmeanslowC,'r','LineWidth',2)
    title('Conditioning Left Eye Movements')
end
```

```
for i = Slow
    figure(1)
    subplot(3,3,5)
    plot(0:T:4.998,fill3{i}(2,:))
    axis([0 5 2 7])
    hold on
    plot(0:T:4.998,fillRmeanslowC,'r','LineWidth',2)
    title('Conditioning Right Eye Movements')
end

for i = Slow
    figure(1)
    subplot(3,3,6)
    plot(0:T:4.998,fill3{i}(1,:) + fill3{i}(2,:))
    axis([0 5 4 12])
    hold on
    plot(0:T:4.998,fillmean3,'r','LineWidth',2)
    axis([0 5 4 12])
    title('Total Conditioning Vergence Movement')
end


% Reasign Variables
eval([Innitials,'2_',Date,'_2Base = fillmean1;'])
eval([Innitials,'2_',Date,'_2Recover = fillmean2;'])
eval([Innitials,'2_',Date,'_2Cond = fillmean3;'])
eval([Innitials,'2_',Date,'_2BaseL = fillLmeanB;'])
eval([Innitials,'2_',Date,'_2BaseR = fillRmeanB;'])
eval([Innitials,'2_',Date,'_2RecoverL = fillLmeanR;'])
eval([Innitials,'2_',Date,'_2RecoverR = fillRmeanR;'])
eval([Innitials,'2_',Date,'_2CondL = fillLmeanslowC;'])
eval([Innitials,'2_',Date,'_2CondR = fillRmeanslowC;'])

eval([Innitials,'2_',Date,'_2BLstd = BLstd;'])
eval([Innitials,'2_',Date,'_2BRstd = BRstd;'])
eval([Innitials,'2_',Date,'_2BTstd = BTstd;'])
eval([Innitials,'2_',Date,'_2RLstd = RLstd;'])
eval([Innitials,'2_',Date,'_2RRstd = RRstd;'])
eval([Innitials,'2_',Date,'_2RTstd = RTstd;'])
eval([Innitials,'2_',Date,'_2CFLstd = CFLstd;'])
eval([Innitials,'2_',Date,'_2CFRstd = CFRstd;'])
eval([Innitials,'2_',Date,'_2CSLstd = CSLstd;'])
eval([Innitials,'2_',Date,'_2CSRstd = CSRstd;'])
eval([Innitials,'2_',Date,'_2CSTstd = CSTstd;'])

%% Clear Variables
clear a C C_2Cal CremoveLeft CremoveRight Date Fast Innitials Remove
Slow T
clear T1* T2* b duration fc fill* filtered fn i j mL mR mean*
sampling_rate
clear slow BLstd BRstd BTstd RLstd RRstd RTstd CFLstd CFRstd CSLstd
CSRstd
clear CSTstd
```

A.3 Root Mean Square Error

```
% This function calculates the Variance and the Standard Deviation of a
% function stuff idkkk

function [E,S] = RMSError(Original_Signal, Reference_Signal)

    for i = 1:length(Original_Signal)
%         step(i) = (abs(Original_Signal(i) - Reference_Signal(i)))^2;
        step(i) = (Original_Signal(i) - Reference_Signal(i))^2;
    end

    step2 = sum(step)/i;

    %S = std(abs(Original_Signal(i) - Reference_Signal(i)),0);
    S = std(step.^.5);
    E = (step2)^.5;
end
```

A.4 Position to Velocity

```
function [data_velocity] =
PositionToVelocity_SRC(data_position,sample_rate,range)

for i = 1:length(data_position)
    if i < range + 1
    data_velocity(:,i) = (data_position(i+range) -
data_position(i))./(range/sample_rate);

    elseif i > length(data_position) - range
    data_velocity(:,i) = (data_position(i) - data_position(i-
range))./(range/sample_rate);

    else
    data_velocity(:,i) = (data_position(i+range) - data_position(i-
range))./(2*range/sample_rate);
    end
end
```

A.5 Bar Plots with Error Bars

```
%**********************************************************************
****
%
%   This is a simple extension of the bar plot to include error bars.
It
%   is called in exactly the same way as bar but with an extra input
%   parameter "errors" passed first.
%
%   Parameters:
```

```
%   errors - the errors to be plotted (extra dimension used if
assymetric)
%   varargin - parameters as passed to conventional bar plot
%   See bar and errorbar documentation for more details.
%
%   Output:
%   [hBar hErrorbar] = barwitherr(..) returns a vector of handles to
the
%                      barseries (hBar) and error bar (hErrorbar)
objects
%
%   Symmetric Example:
%   y = randn(3,4);         % random y values (3 groups of 4
parameters)
%   errY = 0.1.*y;          % 10% error
%   h = barwitherr(errY, y);% Plot with errorbars
%
%   set(gca,'XTickLabel',{'Group A','Group B','Group C'})
%   legend('Parameter 1','Parameter 2','Parameter 3','Parameter 4')
%   ylabel('Y Value')
%   set(h(1),'FaceColor','k');
%
%
%   Asymmetric Example:
%   y = randn(3,4);         % random y values (3 groups of 4
parameters)
%   errY = zeros(3,4,2);
%   errY(:,:,1) = 0.1.*y;   % 10% lower error
%   errY(:,:,2) = 0.2.*y;   % 20% upper error
%   barwitherr(errY, y);    % Plot with errorbars
%
%   set(gca,'XTickLabel',{'Group A','Group B','Group C'})
%   legend('Parameter 1','Parameter 2','Parameter 3','Parameter 4')
%   ylabel('Y Value')
%
%
%   Notes:
%   Ideally used for group plots with non-overlapping bars because it
%   will always plot in bar centre (so can look odd for over-lapping
bars)
%   and for stacked plots the errorbars will be at the original y value
is
%   not the stacked value so again odd appearance as is.
%
%   The data may not be in ascending order.  Only an issue if x-values
are
%   passed to the fn in which case their order must be determined to
%   correctly position the errorbars.
%
%
%   24/02/2011  Martina F. Callaghan    Created
%   12/08/2011  Martina F. Callaghan    Updated for random x-values
%   24/10/2011  Martina F. Callaghan    Updated for asymmetric errors
%   15/11/2011  Martina F. Callaghan    Fixed bug for assymetric errors
&
%                                       vector plots
```

```
%   14/06/2013  Martina F. Callaghan    Returning handle as recommended
by
%                                       Eric (see submission comments)
%   08/07/2013  Martina F. Callaghan    Only return handle if
requested.
%   18/07/2013  Martina F. Callaghan    Bug fix for single group data
that
%                                       allows assymetric errors.
%                                       Also removed dot from display
as
%                                       per Charles Colin comment. The
%                                       handle can be returned to
control
%                                       appearance.
%   27/08/2013  Martina F. Callaghan    Ensuring errors are always
stored
%                                       as lowerErrors and upperErrors
even
%                                       if symmetric.
%
%**************************************************************************
****

function varargout = barwitherr(errors,varargin)

% Check how the function has been called based on requirements for
"bar"
if nargin < 3
    % This is the same as calling bar(y)
    values = varargin{1};
    xOrder = 1:size(values,1);
else
    % This means extra parameters have been specified
    if isscalar(varargin{2}) || ischar(varargin{2})
        % It is a width / property so the y values are still
varargin{1}
        values = varargin{1};
        xOrder = 1:size(values,1);
    else
        % x-values have been specified so the y values are varargin{2}
        % If x-values have been specified, they could be in a random
order,
        % get their indices in ascending order for use with the bar
        % locations which will be in ascending order:
        values = varargin{2};
        [tmp xOrder] = sort(varargin{1});
    end
end

% If an extra dimension is supplied for the errors then they are
% assymetric split out into upper and lower:
if ndims(errors) == ndims(values)+1
    lowerErrors = errors(:,:,1);
    upperErrors = errors(:,:,2);
elseif isvector(values)~=isvector(errors)
    lowerErrors = errors(:,1);
```

```
    upperErrors = errors(:,2);
else
    lowerErrors = errors;
    upperErrors = errors;
end


% Check that the size of "errors" corresponsds to the size of the y-
values.
% Arbitrarily using lower errors as indicative.
if any(size(values) ~= size(lowerErrors))
    error('The values and errors have to be the same length')
end

[nRows nCols] = size(values);
handles.bar = bar(varargin{:}); % standard implementation of bar fn
hold on
hBar = handles.bar;

if nRows > 1
    hErrorbar = zeros(1,nCols);
    for col = 1:nCols
        % Extract the x location data needed for the errorbar plots:
        x = get(get(handles.bar(col),'children'),'xdata');
        % Use the mean x values to call the standard errorbar fn; the
        % errorbars will now be centred on each bar; these are in
ascending
        % order so use xOrder to ensure y values and errors are too:
        hErrorbar(col) = errorbar(mean(x,1), values(xOrder,col),
lowerErrors(xOrder,col), upperErrors(xOrder, col), '.k');
        set(hErrorbar(col), 'marker', 'none')
    end
else
    x = get(get(handles.bar,'children'),'xdata');
    hErrorbar = errorbar(mean(x,1), values, lowerErrors, upperErrors,
'.k');
    set(hErrorbar, 'marker', 'none')
end

hold off

switch nargout
    case 1
        varargout{1} = hBar;
    case 2
        varargout{1} = hBar;
        varargout{2} = hErrorbar;
end
```

A.6 Section Analysis

```
%% Loading of All Data
load('C:\Dropbox\Data\MeanMovements\AJB2_11112013_3Slows.mat')
load('C:\Dropbox\Data\MeanMovements\AJB2_11202013_3Slows.mat')
```

```
load('Basic.mat')
% The Data in this analysis is going to be divided into 3 different
% sections: Latency, Transient, and Steady-State and their RMSE is
taken

%% Isolting and Plotting Latency Period
T = 0:.002:.598; %0.6
Start = T(1)*500+1;
Finish = Start + length(T)-1;

%Subject 1 - Day 1
[Error1(1,1),Deviation1(1,1)] = RMSError...
    (AJB2_11112013_2Base(Start:Finish),Slow2(Start:Finish));
[Error1(1,2),Deviation1(1,2)] = RMSError...
    (AJB2_11112013_2Cond(Start:Finish),Slow2(Start:Finish));
[Error1(1,3),Deviation1(1,3)] = RMSError...
    (AJB2_11112013_2Recover(Start:Finish),Slow2(Start:Finish));
%Subject 1 - Day 2
[Error2(1,1),Deviation2(1,1)] = RMSError...
    (AJB2_11202013_2Base(Start:Finish),Slow2(Start:Finish));
[Error2(1,2),Deviation2(1,2)] = RMSError...
    (AJB2_11202013_2Cond(Start:Finish),Slow2(Start:Finish));
[Error2(1,3),Deviation2(1,3)] = RMSError...
    (AJB2_11202013_2Recover(Start:Finish),Slow2(Start:Finish));

%% Isolting and Plotting Transient Period
T = .6:.002:1.3; %.6 1.3
Start = T(1)*500+1;
Finish = Start + length(T)-1;

%Subject 1 - Day 1
[Error1(2,1),Deviation1(2,1)] = RMSError...
    (AJB2_11112013_2Base(Start:Finish),Slow2(Start:Finish));
[Error1(2,2),Deviation1(2,2)] = RMSError...
    (AJB2_11112013_2Cond(Start:Finish),Slow2(Start:Finish));
[Error1(2,3),Deviation1(2,3)] = RMSError...
    (AJB2_11112013_2Recover(Start:Finish),Slow2(Start:Finish));
%Subject 1 - Day 2
[Error2(2,1),Deviation2(2,1)] = RMSError...
    (AJB2_11202013_2Base(Start:Finish),Slow2(Start:Finish));
[Error2(2,2),Deviation2(2,2)] = RMSError...
    (AJB2_11202013_2Cond(Start:Finish),Slow2(Start:Finish));
[Error2(2,3),Deviation2(2,3)] = RMSError...
    (AJB2_11202013_2Recover(Start:Finish),Slow2(Start:Finish));

%% Isolting and Plotting Steady-State Period
T = 2:.002:4.498; %2 4.5
Start = T(1)*500+1;
Finish = Start + length(T)-1;

%Subject 1 - Day 1
[Error1(3,1),Deviation1(3,1)] = RMSError...
    (AJB2_11112013_2Base(Start:Finish),Slow2(Start:Finish));
[Error1(3,2),Deviation1(3,2)] = RMSError...
    (AJB2_11112013_2Cond(Start:Finish),Slow2(Start:Finish));
[Error1(3,3),Deviation1(3,3)] = RMSError...
```

```
    (AJB2_11112013_2Recover(Start:Finish),Slow2(Start:Finish));
%Subject 1 - Day 2
[Error2(3,1),Deviation2(3,1)] = RMSError...
    (AJB2_11202013_2Base(Start:Finish),Slow2(Start:Finish));
[Error2(3,2),Deviation2(3,2)] = RMSError...
    (AJB2_11202013_2Cond(Start:Finish),Slow2(Start:Finish));
[Error2(3,3),Deviation2(3,3)] = RMSError...
    (AJB2_11202013_2Recover(Start:Finish),Slow2(Start:Finish));

figure(1)
subplot(2,1,1)
h = barwitherr(Deviation1,Error1);
set(h(1),'FaceColor','c');
set(gca,'XTickLabel',{'Latency','Transient','Steady-State'})
ylabel('Root Mean Square Error')
legend('Baseline','Conditioning','Recovery')
title('Subject 1 - Day 1')
axis([.5 3.5 0 0.6])

subplot(2,1,2)
h = barwitherr(Deviation2,Error2);
set(h(1),'FaceColor','c');
set(gca,'XTickLabel',{'Latency','Transient','Steady-State'})
ylabel('Root Mean Square Error')
title('Subject 1 - Day 2')
axis([.5 3.5 0 0.6])

clear Fast* Slow* Time Finish Start T h AJB*
```

REFERENCES

Alvarez, T.L., Bhavsar, M., Semmlow, J.L., Bergen, M.T., Pedrono, C. (2005). Short-term predictive changes in the dynamics of disparity vergence eye movements. *Journal of Vision*. 5. 640-649.

Alvarez, T.L., Semmlow, J.L., Yuan W. (1998). Closely Spaced, Fast Dynamic Movements in Disparity Vergence. *Journal of Neurophysiology*. 79(1). 37-44.

Alvarez, T.L., Semmlow, J.L., Yuan, W., Munzo, P. (May, 1999). Dynamic details of disparity convergence eye movements. *Annals of Biomedical Engineering.* 27(3). 380-90.

Alvarez, T.L., Vicci, V.R., Alkan, Y., Kim, E.H., Gohel, S., Barrett, A.M., Chiaravalloti, N., Biswal, B.B. (Dec 2010). Vision therapy in adults with convergence insufficiency: Clinical and functional magnetic resonance imaging measures. *Optometry & Vision Science*. 87. E985-1002.

Carl, J.R., Gellman, R.S. (May, 1987). Human Smooth Pursuit: Stimulus-Dependent Responses. *Journal of Neurophysiology.* 57(5). 1446-63.

Guo, Y., Kim, E., Alvarez, T. (2011). VisualEyes: A Modular Software System System for Oculomotor Experimentation. *Journal of Visualized Experiments*.

Horwood, A. M., Riddell, P.M. (2009). A Novel Experimental Method for Measuring Vergence and Accommodation Responses to the Main Near Visual Cues in Typical and Atypical Groups. *Strabismus.* 17. 9-15.

Hung, George K., Semmlow, J.L., Ciufferda, K.J. (1986) A Dual-Mode Dynamic Model of the Vergence Eye Movement System. *IEEE Transactions of Biomedical Engineering.* 33(11). 1021-1028.

Hung, George K., Semmlow, John L., Sun, L., Ciuffreda, Kenneth J. (1991). Vergence Control of Central and Peripheral Disparities. *Experimental Neurology*. 113. 202-211.

Janchinski, W. (Jul, 2010). Objective vs subjective measures or fixation disparity for short and long fixation periods. *Ophthalmic Physiological Optics*. 30. 379-90.

Judge, S.J. (1991). *Vergence.* 8. London, UK: MacMillan Press.

Khojasteh, E., Galiana H.L. (2007) Modulation of vergence during the vestibule-ocular reflex. *IEEE EMBC 29th annual international Conference Proceedings.* Lyon, France. 5377-80.

Kim, Eun H., Vicci, Vincent R., Granger-Donetti, B., Alvarez, Tara L. (Jul 2011) Short-term adaptations of the dynamic disparity vergence and phoria systems. *Experimental Brain Research*. 121(2). 267-278.

Lee, Y.Y., Granger-Donetti, B., Chang, C., Alvarez, T.L. (2009). Sustained convergence induced changes in phoria and divergence dynamics. *Vision Research.* 49(24). 2960-72.

Lisberger, S.G., Fuchs, A.F. (1978). Role of Primate Flocculus During Rapid Behavioral Modification of Vestibuloocular Reflex. I. Purkinje Cell Activity During Visually Guided Horizontal Smooth-Pursuit Eye Movements and Passive Head Rotation. *Journal of Neurophysiology*. 41(3). 733-763.

Maddox, E. E. (1893). *The Clinical Use of Prisms and the Decentering of Lenses.* Bristol, England: John Wright & Sons.

Marieb, Elaine M., Hoehn, K. (2010). "The Special Senses." *Human Anatomy & Physiology*. 8th Ed. New York, NY: Pearson. 547-67.

Munoz, Paula, Semmlow, John L., Yuan, Weihong, Alvarez, Tara L. (May 1999). Short term modification of disparity vergence eye movements. *Vision Research*. 39(9). 1695-1705.

Rambold, H., Neumann, G., Sander, T., Helmchen, C. (2006). Age-related changes of vergence under natural viewing conditions. *Neurobiology of Aging*. 27. 163-72

Semmlow, J.L., Hung, G.K., Ciuffreda, K.J. (1986). Quantitative Assessment of Disparity Vergence Components. *Investigative Ophthalmology & Visual Science*. 27(4). 558-564.

Semmlow, J.L., Yuan W. (2002). Adaptive Modification of Disparity Vergence Components: An Independent Component Analysis Study. *Investigative Ophthalmology & Visual Science.*

Yuan, W., Semmlow, J.L. (2000). The influence of repetitive eye movements on vergence performance. *Vision Research*. 40(22). 3089-98.

Yuan, W., Semmlow, J.L., Alvarez, T.L., Munoz, P. (Oct 1999). Dynamics of the disparity vergence step response: A model-based analysis. *IEEE Transactions on Biomedical Engineering*. 46. 1191-1198.