The Effects of Motion Experience on Reflexive Eye Movements and Dynamic Tilt Perception

by

Robert P. Ocampo

B.A. Biology, Psychology
Haverford College, 2004

SUBMITTED TO THE DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN AERONAUTICS AND ASTRONAUTICS
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
[February 2008]
JANUARY 2008

Signature of Author: ______________________
Department of Aeronautics and Astronautics
January 12, 2008

Certified by: ______________________
Daniel M. Merfeld
Associate Professor of Otology and Laryngology
Thesis Supervisor

Certified by: ______________________
Laurence R. Young
Department of Aeronautics and Astronautics
Apollo Program Professor of Astronautics
Professor of Health Sciences and Technology
Thesis Supervisor

Accepted by: ______________________
David Darmofal
Professor of Aeronautics and Astronautics
Chairman, Committee for Graduate Students

THE EFFECTS OF MOTION EXPERIENCE ON REFLEXIVE EYE MOVEMENTS AND DYNAMIC TILT PERCEPTION

by

ROBERT P. OCAMPO

Submitted to the Department of Aeronautics and Astronautics on January 12, 2008 in partial fulfillment of the requirements for the Degree of Master of Science in Aeronautics and Astronautics

Abstract:

Measured differences between human and primate VOR may derive from disparate experiences with psychophysical motion stimuli. To test this hypothesis, horizontal and torsional eye movements, as well as roll tilt perception, were measured in both "naïve" and "experienced" human subjects during 0.5 Hz sinusoidal roll motion. Between these measurement sessions, subjects experienced pseudo-random roll tilt adaptation sessions. For half of the subjects, these adaptation sessions occurred in the light. Subjects participated in 12 measurement sessions and 9 adaptation sessions over the course of three days. (1) The presence of visual cues did not significantly affect horizontal or torsional VOR, but did adversely influence perception of roll tilt as measured using a somatosensory bar. (2) Torsional VOR followed a pattern of habituation and/or adaptation for naïve and dark-adapted subjects. (3) Experienced subjects made horizontal eye movements of significantly larger amplitude than those made by naïve subjects. (4) The motion paradigm yielded a significant decrease in torsional amplitude without a corresponding decrease in horizontal amplitude. These last two findings suggest experience does not affect the horizontal VOR. Consequently, the hypothesis that human/primate horizontal VOR differences stem from dissimilar motion experience is not supported. Primate and human VOR must continue to be studied separately.

Thesis Supervisor: Daniel Merfeld
Associate Professor of Otology and Laryngology

Thesis Supervisor: Laurence Young
Apollo Program Professor of Astronautics
Professor of Health Sciences and Technology

Acknowledgements

The author would like to acknowledge his friends and family for their support:

To those who provided scientific and academic guidance, both at JVPL and MIT: thank you. A special thanks goes to Rich, Mike, and Faisal; you saved my hide on many occasions. Dan and Lionel, this thesis could not have been accomplished without your help.

Quenton Cassidy, Alice Cashin, Superman, Michael Jordan, Calvin and Hobbes, and the Smashing Pumpkins: Some of you are real, some of you are imaginary; but all of you inspired.

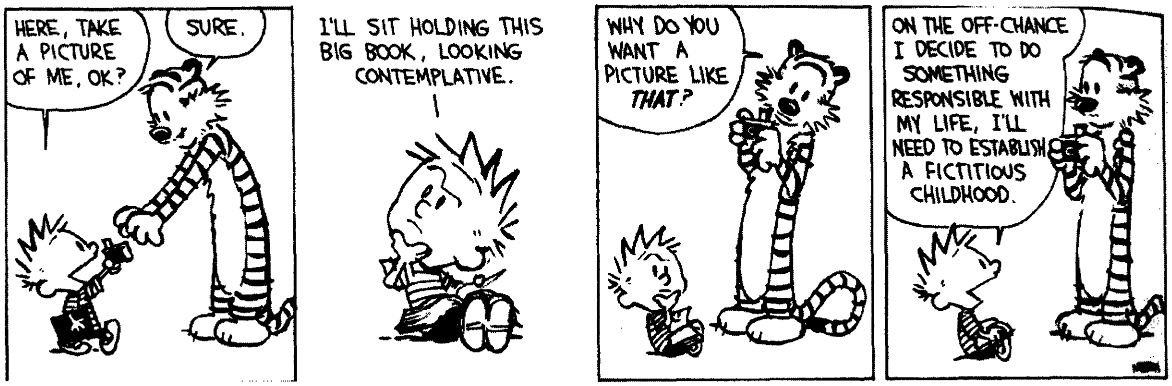
Armin, Adam, and Mandy: You were friendly to me in an especially unfriendly town. Boston would have been a far more difficult experience without you. Becky and Priti, you fed and housed me when I was without shelter, and I cannot thank you enough. Julie, you made sure I stayed sane. I'm fairly certain that without you, I would have broken the spines of far more textbooks.

Alan, Julian, Dennis, Jenna, and Goose: You constantly remind me of what is really important: not academics, not work, but friendship and family.

Special thanks to the towns of Lake Placid, NY, Cocoa Beach, FL, and Lincoln, NH. The process of writing was made far more enjoyable in your friendly confines.

Lastly, and most importantly, thank you Mom and Dad. You believed in me when no one else did. I could not have done this without you.

Biographical Sketch

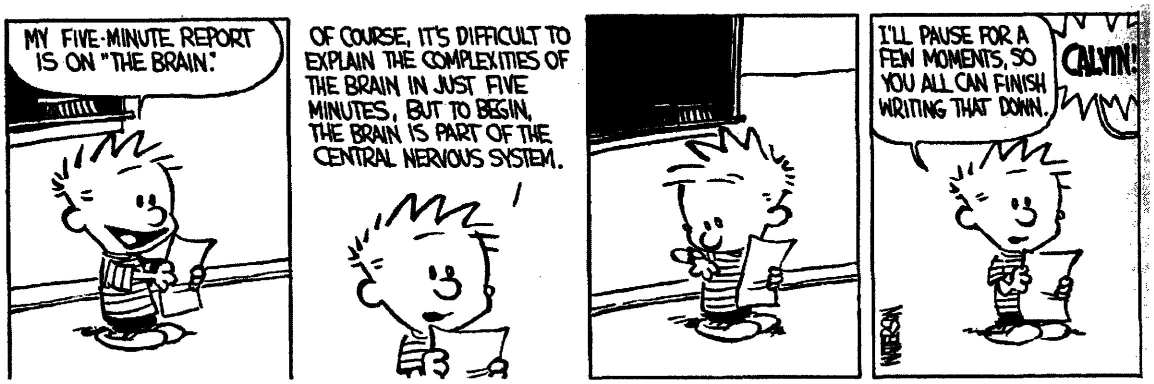


Robert Ocampo was born to two very loving and supportive parents, and considers Hillsborough, CA his home. He received his primary basketball education at South Hillsborough School, where he dominated the courts for nearly a decade. Seduced by a nearby 24-hour IHOP, Robert chose to attend Haverford College, where he obtained his bachelor's degree in Biology and Psychology.

In 2004, Robert thru-hiked the Appalachian Trail.

He looks forward to leaving Boston, disappearing into the mountains, and never looking back.

Introduction



Purpose:

Living and working in space is a challenge for human beings. Having evolved in a constant 1-g environment, the human body does not readily adapt to novel gravitational conditions. Physiological responses that rely on feedback from gravity-sensing organs—particularly spatial perception and reflexive eye movements—are fallible during and immediately following spaceflight. These shortcomings add danger to an already risky endeavor.

Nevertheless, NASA plans to return to the moon in the following decade, a trip that requires no less than 6 gravitational transitions[1]. Certainly, the voyage is feasible from a biological standpoint; indeed, it has been achieved safely in the past. Tomorrow's astronauts, however, face cumulatively greater risks: they will return in greater numbers, and for longer time intervals. For this reason, countermeasures that enhance the body's ability to adapt to microgravity must be developed. The task starts here on earth, as we

---

[1] Astronauts on a round trip voyage to the moon face 4 discrete gravitational environments. In addition to the 1-g environment of earth, moonbound astronauts face microgravity en-route, $1/6^{th}$ gravity on the lunar surface, and multiple "g's" during terrestrial launch and landing. Together, these equate to the 6 gravitational transformations mentioned here.

work to accurately characterize the sensory network responsible for interpreting gravity and motion: the vestibular system.

Background:

Humans use information derived from both visual and vestibular cues to estimate both body movement and spatial orientation (Young 1984). The vestibular system, located in the inner ear, is comprised of two distinct organs: the otolith organs and semicircular canals. Specifically, the semicircular canals act as integrating angular accelerometers, and accurately measure head angular velocity for frequencies above 0.04 Hz. The otolith organs act as linear accelerometers, measuring gravito-inertial force (GIF, f), the vector sum of inertial force (translation, a) and gravity (tilt, g).

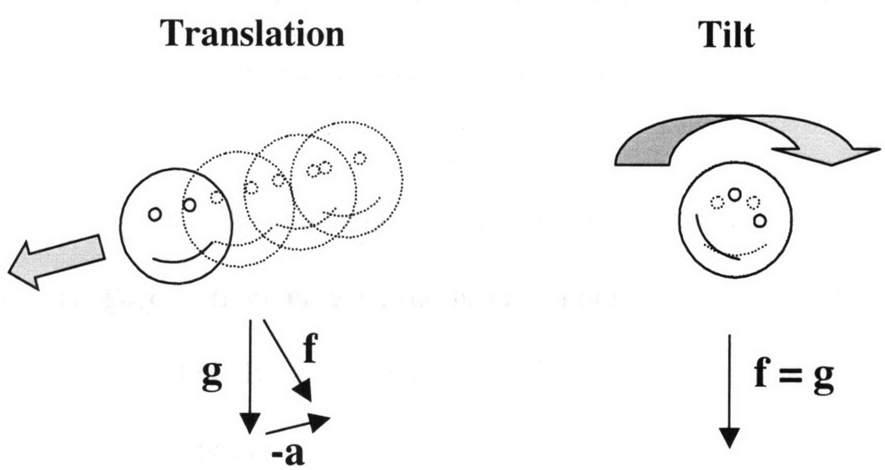


Figure 1: Tilt and translation can produce an equivalent GIF vector with respect to the idiotropic vector. Adapted from Zupan, Peterka, and Merfeld (Zupan et al. 2000)

Tilt/Translation Ambiguity:

Afferent information relayed from the otolith organs is inherently ambiguous. According to Einstein's equivalence principle (Einstein 1908), no linear accelerometer can alone resolve gravito-inertial force into its constituent components (see figure 1). Indeed, neurons in the utricular otolith fire at the same rate during tilt and translation of equal frequency and amplitude, suggesting that inertial and gravitational forces are indistinguishable at the peripheral level (Fernandez and Goldberg 1976a; b; c).

Eye movement response characteristics epitomize this ambiguity. During angular and linear acceleration, the vestibular system works to stabilize the eye in space so as to minimize retinal slip (vestibulo-ocular reflexes or VORs). However, low frequency sinusoidal *translation* generates small torsional eye movements in both primates and humans (Angelaki et al. 2000; Lichtenberg et al. 1982), a response compensatory for image stabilization during *tilt*, but anti-compensatory for *translation*.

Resolution of the tilt/translation ambiguity:

A number of crucial physiological tasks, such as walking and running, require accurate resolution of the measured GIF. Fortunately, despite the GIF's inherent ambiguity, vestibular responses to motion are generally compensatory (Green et al. 2005). Humans have little difficulty accurately perceiving ego-motion during either active or passive movement. Furthermore, VORs adequately minimize retinal slip for motions in the behaviorally relevant frequency range ((Angelaki et al. 1999; Merfeld and Young 1995) as cited by (Zupan et al. 2000)). Even during low-frequency translation (as described above) anticompensatory torsional eye movements are accompanied by compensatory horizontal VORs (Angelaki et al. 2000; Lichtenberg et al. 1982). These results suggest that the nervous system has identified a reliable, if imperfect, means of distinguishing tilt from translation.

How the brain resolves this ambiguity remains undetermined. Three distinct, though not necessarily exclusive hypotheses have been proposed: peripheral processing, frequency segregation and internal models.

Peripheral Processing:

The peripheral processing hypothesis suggests that translation is signaled by activation of phasic irregular otoliths, and tilt is marked by stimulation of tonic regular afferents (Mayne 1974; Young and Meiry 1967).

Frequency Segregation:

Alternatively, the frequency segregation hypothesis postulates that motion is interpreted as either tilt or translation based on the frequency of the stimulus (Mayne 1974; Paige and Tomko 1991). Motion below 0.2 Hz is interpreted as tilt; motion above 0.2 Hz is perceived as translation. Theoretical and experimental evidence suggests that simple filtering may partially explain the response characteristics of VOR and motion perception.

Theoretical Evidence for Frequency Segregation:

Humans rarely experience low frequency inertial motion; to do so requires an uncharacteristically long translation. Paige and Tomko state that to meet the perceptual threshold of 0.01g at 0.05 Hz requires a translation of 1m (Paige and Tomko 1991). Such lengthy translations rarely occur in nature. Thus, interpreting low frequency motion as tilt is ordinarily the correct assumption.

High frequency stimulation of the otoliths, in the absence of other cues, is generally attributed to translation. While high frequency tilt is not uncommon, afferent signals from the semicircular canals (which are reliable in this frequency range), provide a secondary cue to help correctly distinguish the motion. Indeed, torsional eye responses in the squirrel monkey decrease with increasing frequency, a result that is consistent with low-pass filtering (Telford et al. 1997).

Perceptual Evidence for Frequency Segregation:

During eccentric centrifugation at constant velocity, humans typically experience a tilting sensation despite body alignment with the gravity vector (Graybiel and Brown 1951; Merfeld et al. 2001). Here, low frequency centripetal acceleration is interpreted as tilt (albeit, incorrectly), a finding that suggests frequency segregation may also help to regulate the perception of motion.

Even the limitations of filtering as a means of motion interpretation provide support for the frequency segregation hypothesis. In the frequency range where filtering is ineffective (mid-level, "crossover frequencies'), motion sickness, a response generally attributed to sensory conflict (Oman 1990), is most pronounced (Wood 2002). Conversely, where filtering is most effective, sensory conflict is at a minimum (as indicated by a decrease in motion sickness susceptibility). Though correlative, these associations strongly suggest that motion perception is at least partially influenced by frequency segregation.

Internal Models:

The third hypothesis presumes that the brain uses an "internal model" to resolve linear acceleration from gravity (Merfeld et al. 1999; Merfeld and Zupan 2002). Internal models are defined as neural representations of physical relationships; in this specific instance, the internal model represents the relationship between gravito-inertial force (GIF, $\hat{f}$), gravity ($\hat{g}$), and linear acceleration ($\hat{a}$) (see equation 1b).

Physical Representation: $f = g - a$ (eqn. 1a)

Neural Estimates: $\hat{f} = \hat{g} - \hat{a}$ (eqn. 1b)

Mathematically, gravito-inertial force ($\hat{f}$) can result from countless combinations of gravitational and linear forces. To adequately define all three variables in an internal model, gravity or linear acceleration must be estimated. The internal model hypothesis postulates that the semicircular canals provide a cue that helps estimate gravity with the following equation (see equation 2b)[2]:

Physical Representation: $\frac{dg}{dt} = \omega \times g$ (eqn. 2a)

Neural Estimates: $\frac{d\hat{g}}{dt} = \hat{\omega} \times \hat{g}$ (eqn. 2b)

[2] In this example the semicircular canals are used to derive the estimate of gravity; however any sensory, cognitive, or efferent cue can also contribute to the gravitational estimate.

By integrating both sides of the equation, $\hat{g}$ can be determined:

$$\text{Physical representation: } g = \int (w \times g)dt \text{ (eqn. 3a)}$$

$$\text{Neural representation: } \hat{g} = \int (\hat{w} \times \hat{g})dt \text{ (eqn. 3b)}$$

In these equations, $\hat{\omega}$ refers to angular head velocity as measured by the semicircular canals. From this gravitational estimate, an estimate of linear acceleration can be derived from the measured GIF, as shown by combining equation 1b with 3b.

$$\hat{a} = \int (\hat{w} \times \hat{g})dt - \hat{f} \text{ (eqn. 3)}$$

Neuronal Evidence for Internal Models:

The firing patterns of cells in the vestibular and fastigal nuclei provides evidence, at the neuronal level, for an internal model (Angelaki and Dickman 2003; Green et al. 2005). So-called "Otolith+Canal" cells (Angelaki and Dickman 2003) receive afferent input from both the semicircular canals and otolith organs, but fire only in response to translation. This discovery indicates that a centralized neural network exists to resolve the ambiguous vestibular signals[3].

---

[3] For a neural network to give rise to an internal model, there must exist a cellular mechanism capable of temporally integrating angular velocity (Green and Angelaki, 2004). (While Angular *position* is required to resolve the GIF vector, the semicircular canals afferents encode angular *velocity*) Though this neural integrator is not well understood, the discovery of Otolith+Canal cells affirms the existence of such a mechanism.

Caloric Evidence for Internal Models:

Several lines of psychophysical evidence support the internal model hypothesis. When humans (Coats and Smith 1967; Peterka et al. 2004) and primates (Minor and Goldberg 1990; Paige 1985) are exposed to caloric stimuli, the magnitude of the VOR response is dependent on position; when supine, the response is larger than when prone. This asymmetry may be due to stimulation of the semicircular canals without simultaneous activation of the otoliths. In this instance, the gravito-inertial force measured by the static otolith organs does not equal the shifting estimate of gravity provided by the semicircular canals. According to the internal model hypothesis, this disparity is interpreted as linear acceleration, and linear VORs are induced to compensate for the perceived motion. Based on geometric considerations, the linear VOR should sum with the canal-driven angular VOR while supine, but subtract from the angular VOR while prone, a prediction experimentally verified by Peterka and colleagues (Peterka et al. 2004).

"Dumping" Protocol:

Rapid tilt after sudden angular deceleration (post-rotatory tilt) induces linear VORs which are consistent with an internal model (Merfeld et al. 1999; Zupan et al. 2000). The physical dynamics of the semicircular canals prohibit accurate assessment of motion during rapid deceleration. In fact, during deceleration, the canals actually measure angular motion in the direction *opposite* the initial rotation (Wilson and Melvill Jones 1979). This illusory movement consequently shifts the estimate of gravity. As with caloric stimulation, this new gravitational vector does not match the GIF measured by the static otolith organs, thereby generating a non-zero estimate of linear acceleration. The internal model predicts that this induced linear acceleration should generate linear VORs that are dependent on the direction of the preceding rotation and the final position post-tilt. Indeed, nose-up rotations *do* generate larger VORs than nose-down rotations (Zupan et al. 2000).

Off-Vertical Axis Rotation (OVAR):

Subjects rotated about an axis not aligned with the gravitational vertical receive dynamic signals from both the semicircular canals and the otolith organs. Once rotation ceases, the two vestibular organs convey conflicting information—the semicircular canals indicate rotation in the direction opposite the preceding motion, whereas the otoliths correctly indicate motion termination. Thus, the otolith-measured GIF is unequal to the estimate of gravity provided by the semicircular canals. As discussed in the "Dumping" (p. 12) and "Caloric" (p. 11) sections, the internal model interprets this inequality as linear motion, inducing horizontal eye movements that are dependent on the subject's final orientation. Indeed, after yaw rotation along an earth-horizontal axis ("barbeque spit"), horizontal VOR is greater when the final orientation is nose up than nose down (Zupan et al. 2000).

Combined Tilt and Translation:

Primates exposed to a novel tilt-translation paradigm also exhibit eye movement responses that reflect an internal model of motion (Angelaki et al. 1999; Angelaki et al. 2001). During inter-aural, sinusoidal translation, the shear force on the otoliths can be eliminated by carefully incorporating sinusoidal tilt ("Tilt-Translation" paradigm). In labyrinthine-intact primates, robust torsional and horizontal eye movements are appropriately compensatory (Angelaki et al. 1999; Angelaki et al. 2001). Upon inactivation of the semicircular canals, however, eye movements follow the resultant interaural acceleration (Angelaki et al. 1999). In this example, canal cues appear to regulate the interpretation of otolith afferents, a finding consistent with an internal model.

Perception and Action Use Qualitatively Different Mechanisms:

The frequency segregation and internal model hypotheses may not be mutually exclusive. Merfeld and colleagues postulate that a combination of both internal models and frequency segregation are used by the human central nervous system to interpret motion. Frequency segregation may be principally utilized to drive reflexive eye movements, and internal models may be primarily responsible for spatial perception (Merfeld et al. 2005a; b; Zupan and Merfeld 2005).

The frequency segregation hypothesis presumes that motion is categorized based on the frequency of the stimulus. Low frequency motion is perceived as tilt; high frequency motion is identified as translation. While the frequency segregation hypothesis predicts a number of experimental findings (Paige and Tomko 1991; Telford et al. 1997; Wood 2002), it fails to explain how "non-categorical" motion is interpreted. Humans accurately perceive low frequency translation and high frequency tilt (Merfeld et al. 2005a), a finding that suggests that human perception is *not* dictated by frequency segregation.

Instead, human perception appears to be governed by an internal model. Increasing the vertical distance from the axis of rotation during roll tilt increases the shear force on the otoliths while maintaining uniform canal cues. Despite the changing otolith signal, tilt continues to be correctly perceived (Merfeld et al. 2005b). This indicates canal cues are required to interpret the otolith signal during motion perception, a finding in line with an internal model of motion.

In contrast, radial position relative to the axis of rotation *does* strongly affect the magnitude of the vestibulo-ocular reflex in humans (Merfeld et al. 2005b). Horizontal VOR increases with distance (and, by association, inter-aural acceleration), despite uniform canal cues. Though consistent with simple filtering, these results do not preclude the possibility that internal models contribute to the VOR response (Merfeld et al. 2005a; b).

Monkeys vs. Humans:

The human VOR results just described stand in direct contrast to Angelaki's VOR measurements in non-human primates. As mentioned previously, normal rhesus monkeys exposed to a "Tilt-Translation" motion paradigm exhibit compensatory eye movements predicted by an internal model, but antithetical to simple filtering (Angelaki et al. 1999; Angelaki et al. 2001). Human VORs characteristics, on the contrary, appear consistent with filtering (Merfeld et al. 2005a; b).

These contradictory findings may be due to differences between species. Indeed, as reported by Merfeld (Merfeld et al. 2005b; Merfeld and Zupan 2002), the ocular response to motion differs in non-human primates. While the eye axis of rotation in humans remains constant following post-rotational tilt and fixed-radius centrifugation, the monkey eye axis tends to align with the resultant gravito-inertial force. In addition, horizontal VOR in humans immediately following both centrifugation and post-rotatory tilt shows a strong orientation dependency. Such dependence on orientation is not found in monkeys. Lastly, human VOR appears less influenced by static otolith cues than monkey VOR (Haslwanter et al. 2000).

Motion Experience and VOR:

Another possible explanation, explored in this thesis, is that motion experience enhances the brain's ability to accurately interpret vestibular information. Because of ethical and financial considerations, most laboratories test the same monkey in a number of different experiments [Merfeld, personal communications—May 2006]. This repeated testing may improve the VOR response. Human subjects do not typically participate in more than a few psychophysical studies; thus, reported inter-species VOR differences may at least partially be attributed to variations in motion experience.

Indeed, a number of studies have shown an effect of motion exposure on reflexive eye movements. Fighter pilots, who experience high G-loads in flight, typically show different angular VOR characteristics across a range of rotation frequencies (Ahn 2003). The horizontal VOR in pilots also better compensates for slow harmonic accelerations (Lee et al. 2004). Moreover, even a small amount of exposure can improve the vestibulo-ocular response to motion, as student pilots have been show to exhibit a more appropriate horizontal VOR gain than non-pilots (Lee et al. 2004).

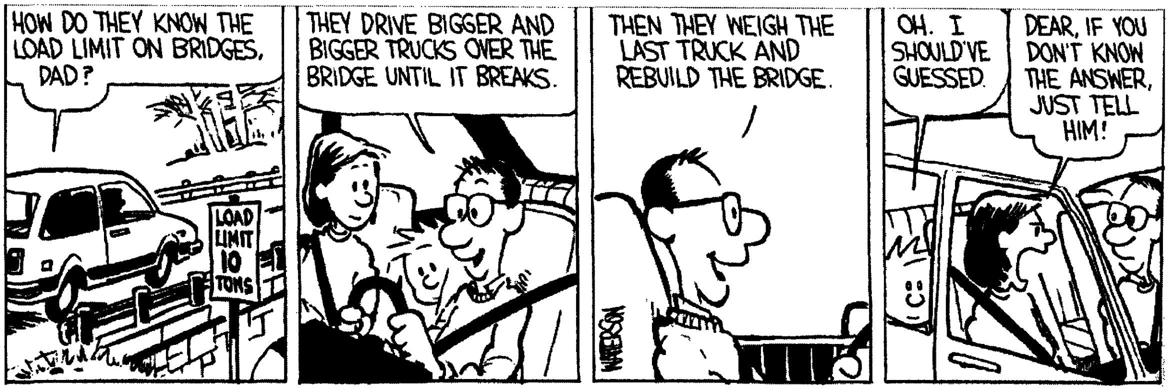
Motion Experience and Spatial Perception:

Motion exposure has also been shown to influence spatial perception. In space, perception of tilt during radial centrifugation changes over the course of a mission (Clement et al. 2001), as astronauts gain experience with the novel gravitational environment. Gymnasts, who are well acquainted with rolling and tumbling motions that dynamically stimulate both the canals and otolith organs, have been shown to better approximate their subjective postural vertical (SPV) (Bringoux et al. 2000) than non-gymnasts. Indeed, experience is thought to play such a strong role in human adaptation that it has been suggested as a means of preparing astronauts for spaceflight. By pairing tilt motion with visual translation information, a conflicting interaction on earth, but an applicable relationship in space, astronauts can gain experience with novel spaceflight sensations while still on earth (Woodard et al. 1987).

Hypothesis:

Together, these findings suggest that VOR *and* spatial orientation, though governed by seemingly different mechanisms, may both be influenced by prior motion experience.

Methodology:



Approval:

This experimental protocol was approved by the Internal Review Board at the Massachusetts Eye and Ear Infirmary (MEEI) and the Committee on the Use of Humans as Experimental Subjects (COUHES) at the Massachusetts Institute of Technology (MIT).

Subject recruitment:

8 subjects (3 females, 5 males, mean age 30.5 years, range 25-47) were recruited for this experiment. Only individuals who were in good physical health and free of back, neck, joint, and dental problems were eligible to participate. In addition, potential subjects who exhibited a high proclivity for motion sickness were excluded from the experiment. Prior to experimental testing, all subjects underwent a series of standardized tests to ensure normal vestibular function. (Jenks Vestibular Diagnostic Lab, Massachusetts Eye and Ear Infirmary—see Appendix A, p. 67).

Each subject signed an Informed Consent form and an Authorization to Release Protected Health Information form. Subjects who participated in the experimental portion of this study were financially compensated for their time.

Naïve vs. Experienced:

A questionnaire was utilized in this study to investigate level of experience with atypical motion (see Appendix B). Subjects were classified either as "naïve" or "experienced" based on their responses. Four of the five male subjects were classified as experienced; all three female subjects were categorized as naïve[4].

The four experienced subjects had either considerable experience with passive motion testing devices (such as centrifuges, linear sleds, tilt devices, and motion simulators) and/or extensive flight experience (≥100 hours pilot in command). The remaining four subjects (all non-pilots with little passive motion testing experience) were classified as naïve.

The fact that most experienced subjects were pilots is significant given the strong correlation between flight training and reflexive eye movements. Compared to non-pilots, compensatory eye movements in pilots tend to be more appropriate during both earth-vertical yaw rotation (Ahn 2003) and slow harmonic acceleration (Lee et al. 2004). Together, this suggests that our classification system was well suited to uncover even subtle differences in spatial perception between naïve and experienced subjects.

---

[4] While no studies have yet identified gender differences in compensatory eye movements, a number of experiments have shown a robust gender difference in spatial perception (for detailed literature review, see Halpern, 2000). Typically, men perform better on tests of spatial perception (most notably in Piaget's water level task (Thomas et al. 1973; Witting and Allen, 1984). Thus, analysis of spatial perception in naïve (mostly female) and experienced subjects (all male) was initially approached with caution. However, as will be shown, spatial perception did not differ between naïve and experienced subjects; thus it can safely be assumed that gender did not act as a confound in this experiment.

Experimental Testing:

All eight subjects participated in three days of testing. Each of the three days of testing consisted of four "measurement' sessions alternating with three "adaptation" sessions (see figure 2). The majority of subjects were tested over a span of three consecutive days. No subject participated in more than one testing session a day, and all subjects completed the three testing sessions within a one-week time period.

Both the measurement and adaptation sessions were performed on a swinging motor-driven chair known as the "tilt device" (Neurokinetics, Pittsburgh, PA) that was specifically configured for roll tilt motion.

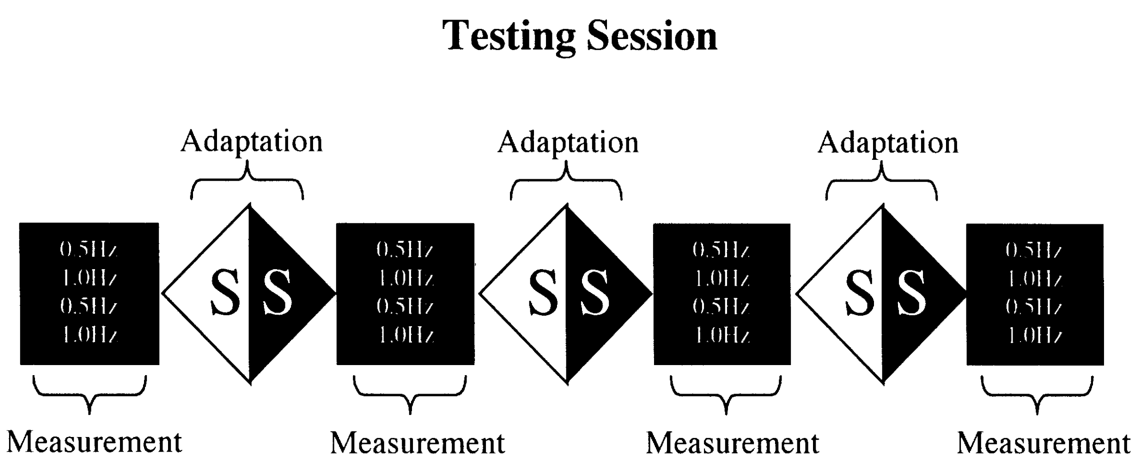


Figure 2: For each testing session, four measurement sessions were interspersed with three adaptation sessions. Measurement sessions consisted of four movements (2 x 0.5 Hz and 2 x 1.0 Hz motions) performed in the dark. Adaptation sessions consisted of pseudo-random sum of sines (SS) motions conducted either in the light or dark. Each subject experienced a total of three testing sessions over the course of three separate days.

Head Center:

Prior to the start of each testing session, the external auditory meatus for each subject was aligned with a horizontal reference line located on the tilt device chair. The subject was then vertically translated so this reference line was level with the axis of rotation (within 1cm). In this manner, tangential and centripetal accelerations were kept at a sub-threshold level during both the measurement and adaptation sessions ((Benson et al. 1986; Jones and Young 1978), as cited by (Merfeld et al. 2005a)).

Fixation Material:

All subjects wore a five-point safety harness, in combination with leg, torso, and foot restraints. Foam padding was also employed when necessary to further minimize body motion with respect to the device. Lastly, a moldable, thermoplastic mask (WFR-Aquaplast, Wyckoff, New Jersey) was used to fix the subjects head to the device headrest. This mask was custom-fitted for each subject, and allowed for rapid removal.

Measurement Sessions:

Eye movements and spatial perception were measured during the measurement sessions. With the ear aligned at the center of rotation, subjects were tilted in roll (± 20°) at two alternating sinusoidal frequencies (the exact sequence: 0.5 Hz, 1.0 Hz, 0.5 Hz, 1.0 Hz). The1.0 Hz trial consisted of 20 steady-state cycles; the 0.5 Hz trial consisted of 15 steady-state cycles. For both 0.5 Hz and 1.0 Hz trials, the angular velocity was linearly increased (and decreased) over 3 and 4 cycles, respectively, so as to limit rapid and uncomfortable acceleration. A brief (~10s) delay occurred between each of the four profiles. All measurement sessions were performed in the dark.

Adaptation Sessions:

During the adaptation sessions, subjects experienced pseudo-random sum of sines roll tilt ("SS motion"). This type of motion was designed to expose subjects to a variety of roll tilt frequencies (0.1, 0.2, 0.5, 0.7, and 1.0 Hz, with the tilt amplitude for each frequency equal to 6.1°). All subjects experienced three variations ("profiles") of pseudo-random sum of sines tilt. The characteristics of each profile were the same except for the amount of phase lead between frequencies. The three phase separations were 0, 144, and 216 degrees. Subjects were positioned so their ears were aligned with the center of rotation, and tilted no more than ± 20°. Each adaptation session lasted 15 minutes.

Light vs. Dark:

Half of the naïve and half of the experienced subjects experienced the adaptation sessions in the dark; the remaining four subjects experienced adaptation in the light. For each subject, light level was consistent from session to session and day to day.

Visual cues were provided to some of the subjects in an attempt to improve the appropriateness of the VOR and perceptual response. Past research has shown that visual cues influence the human VOR. While the vestibulo-ocular reflex operates primarily in an open-loop manner, a mechanism for plasticity exists to maintain the VOR in a manner that keeps retinal images stable during novel visual-vestibular interactions[5] (Miles and Lisberger 1981). Humans exposed to binocular lenses (Paige and Sargent 1991) or field-reversing prisms (Jones 1977) show an adapted VOR which compensates for the novel

[5] Anecdotally, VOR plasticity is easily demonstrated with a pair of prescription glasses. Putting on the glasses requires a decrease in VOR gain to compensate for the magnified visual scene.

visual-vestibular environment. Additionally, exposure to optokinetic stimuli in conflict with vestibular cues has been shown to influence VOR gain in a number of species, including mice (Faulstich et al. 2004), primates (Angelaki and Hess 1998; Wei and Angelaki 2001), and humans (Watanabe et al. 2003). In this experiment, visual cues faithfully supplemented vestibular afferent signals, and as such were predicted to improve both spatial perception and the VOR response.

Visual Cues:

A horizontal line painted on the wall facing the subject served to enhance the subject's perception of earth horizontal. In addition, three earth stationary posters helped orient the subject to the gravitational vertical. Two of the posters were identical and depicted human figures walking across a crosswalk (taken from the Beatles' "Abbey Road" album cover). The remaining poster depicted a number of vertical pencils aligned in parallel.

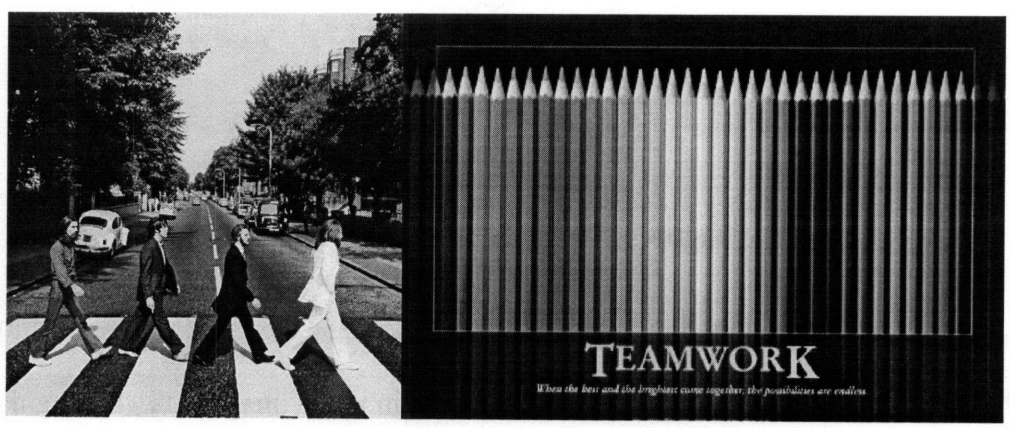


Figure 3: The Beatles "Abbey Road" album cover, and a poster with vertical pencils, were used as visual cues for light adapted subjects. Note the strong, vertical cues.

Subjects were also exposed to several visual cues that rotated synchronously with the tilt device. The most notable of these cues was the tilt device arm, which continuously took up nearly 40 percent of the visual field[6]

Eye Movement Measurements:

Eye position was measured using a clinically certified video-oculography (VOG) mask (SensoriMotor Instruments (SMI), Teltow, Germany). Infrared LEDs invisible to the subject were used to illuminate the eye.

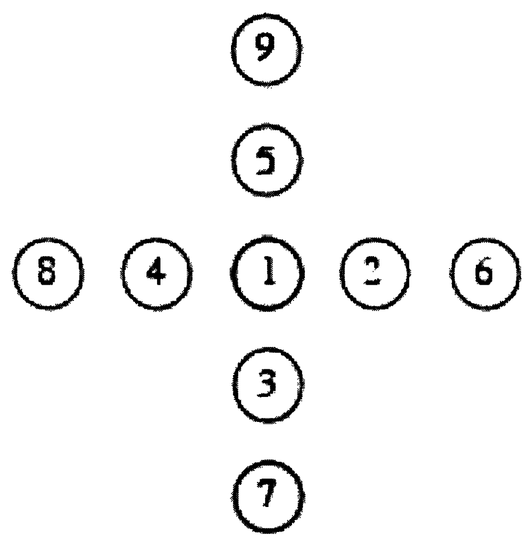


Figure 4: 9 point calibration bar

A standard 9-point calibration performed prior to the first testing session allowed horizontal and vertical eye position to be inferred from video image data. During these calibration sessions, each subject, sitting a measured distance from the "calibration bar" (9 LEDs evenly distributed in a cross-like pattern, see figure 4) fixated on individual

[6] Only subjects adapted in the light were exposed to the visual cues, and then only during the adaptation session.

LED targets as they were illuminated in sequence. By comparing the actual location of the eye with the location predicted by eye-LED geometry, each eye was separately calibrated.

Torsional eye movements were calculated with the help of a reference image taken immediately after the calibration session. Commercially developed software (SensoriMotor Instruments (SMI), Teltow, Germany) calculated the grey levels of each reference image to produce an ellipse concentric with the subject's pupil. Subsequent angular changes in the ellipse with respect to the reference image were recorded as torsion. Reference images were taken for both eyes.

Bite Bar:

A moldable bite bar was made for each subject, and was worn throughout each day of testing. The bite bar attached to the VOG mask and helped minimize movement of the head with respect to the cameras.

Calibration Bar:

A 9-point LED panel ("Calibration Bar"—see figure 4) was built specifically for this experiment. In both the horizontal and vertical planes, LEDs were spaced 2 inches apart, generating an angle from ocular centerline of roughly 10° for the inner ring of LEDs, and 20° for the outer ring (The actual angles differed for each subject, directly depending on their horizontal distance from the calibration bar). During the calibration session, each LED was illuminated for approximately 2 seconds.

Somatosensory Bar:

A non-visual somatosensory task was chosen as the perceptual measurement in this experiment, as past research has shown that aligning a visual stimulus interferes with eye responses and may provide an ambiguous measurement of spatial perception (Curthoys 1996). Subjects were asked to continuously align a metal "somatosensory bar" with their perception of earth-horizontal (Merfeld et al. 2001; Park et al. 2006; Zupan and Merfeld 2003).

The somatosensory bar was attached to an analog potentiometer connected to a 5V power supply, such that rotational changes in bar position generated a commensurate change in voltage. Somatosensory bar voltage was continuously measured and recorded throughout each measurement session.

The mathematical relationship between bar angle and voltage was obtained by measuring voltage at 17 discrete bar angles (-80°, -70°, -60°, -50°, -40°, -30°, -20°, -10°, 0°, 10°, 20°, 30°, 40°, 50°, 60°, 70°, 80°) prior to the start of the experiment. A digital inclinometer accurate to within 0.01° degrees was used to assure that the bar was accurately positioned in each of the 17 positions. Voltage was measured on a digital voltmeter, accurate to within 0.0001V.

A linear regression was performed on this preliminary somatosensory bar data, and a line of best fit was calculated using the method of least squares (Angle = 75.5 x Voltage – 100.64). The relationship between bar angle and voltage was highly correlated, with an R^2 value equal to 0.999.

Somatosensory Bar Bias:

Prior to each measurement session, each subject was asked to offset the bar and then align it with his or her perception of earth horizontal. Once this alignment had been performed, the subject pressed a button on the bar. Each subject performed this task six times per measurement session. Bar position immediately preceding button push was averaged for each measurement session, and this value was used as the perceptual bias for the session.

Data Acquisition:

Eye movement data was collected at 60 Hz on a commercially built computer (SensoriMotor Instruments (SMI), Teltow, Germany). Somatosensory bar data, as well as analog translation and digital tilt signals, were filtered and collected on a separate "Data Acquisition Computer" at 120 Hz. Motion start initiated a digital trigger that was recorded by both the SMI and Data Acquisition computers. This trigger signal provided a means of temporally syncing the data acquired by the two computers.

Practice Session:

Subjects participated in a practice session prior to the first testing session to ensure they understood the somatosensory task, as well as to familiarize them with the motion of the tilt device. All subjects experienced 20° static roll tilts to the left and right, as well as 3 cycles of a 20°, 0.02 Hz sinusoidal roll tilt[7]. After the practice session, subjects were asked if they felt comfortable with the task; no subject requested additional

[7] A relatively low tilt frequency was chosen for the practice session so as to minimize adaptation.

practice.

Monitoring the subjects:

All subjects were monitored for alertness; those subjects who appeared tired or drowsy were given trivia questions. For dark-adapted subjects, lights were briefly turned on prior to each measurement session. This helped ensure vergence remained relatively constant during measurement sessions (see the "Vergence" section (p. 35) for more on vergence).

Motion Sickness:

Movement during both the measurement and adaptation sessions was relatively benign and elicited little sensory conflict. Nevertheless, we remained alert for evidence of motion sickness throughout the experiment. At the end of each testing and training session, subjects were asked to rate their motion sickness on a scale of 0-10, with 0 indicating they felt fine and 10 indicating they were about to throw up. To ensure subject safety, the testing protocol dictated that all motion would cease once a response higher than 4 was given; however, at no point did any subject indicate a motion sickness value greater than 2.

As an additional precaution, the duration of motion was increased incrementally during the adaptation session to minimize the possibility of motion sickness (which tends to build rapidly). If a subject felt fine after one minute of SS motion, he or she underwent two minutes of SS motion; if he or she felt fine after two minutes of SS motion, he or she would experience three minutes of SS motion. This buildup continued until the subject showed he or she could safely experience 5 minutes of uninterrupted SS motion. In this piecemeal manner, subjects experienced 15 minutes of motion during each adaptation session. After the first day's adaptation session, each subject experienced 5-minute SS intervals during subsequent adaptation sessions.

Fatigue:

Subjects were tested on three separate days. Eye movements and perceptual responses were compared from day to day to ensure that any changes were due to psychophysical modifications, and not fatigue.

Kinematic Corrections:

The video-oculography mask utilized in this experiment measured eye movements in an eye-fixed coordinate system. A kinematic correction was made to rotate the acquired VOG data from an eye fixed to a head fixed reference frame.

$$\vec{\lambda} = \begin{bmatrix} \lambda_x \\ \lambda_y \\ \lambda_z \end{bmatrix} = S(\theta_1, \theta_2)\,\vec{\dot{\theta}} = S(\theta_1, \theta_2) \begin{bmatrix} \dot{\theta}_3 \\ \dot{\theta}_2 \\ \dot{\theta}_1 \end{bmatrix} \quad \text{(eqn. 4)}$$

Where,

$$S(\theta_1, \theta_2) = \begin{bmatrix} 0 & -sin(\theta_1)cos(\theta_1) & cos(\theta_2) \\ 0 & cos(\theta_1)sin(\theta_1) & cos(\theta_2) \\ 1 & 0 & -sin(\theta_2) \end{bmatrix} \quad \text{(eqn. 5)}$$

In these equations, λ_x, λ_y, and λ_z refer to torsional, vertical, and horizontal eye velocity in a head fixed reference frame. θ_1, θ_2, and θ_3 refer to horizontal, vertical, and torsional eye position in an eye fixed reference frame. Both uncorrected and corrected eye velocity values were analyzed in this study. While corrected values have the advantage of being largely unaffected by head motion, they are easily contaminated by small calibration errors, particularly when there is a predominant response (in this case, a large, predicted torsional response). For further details, see Merfeld (Merfeld 1990).

Fitting the data:

Eye position was differentiated and fast phase saccades were manually removed to generate measurements of torsional and horizontal slow phase velocity. For each trial[8], both torsional and horizontal SPV were fitted using a least-mean-square linear regression to the equation:

$$x(t) = B + A_c \cos(2\pi f t) + A_s \sin(2\pi f t) \text{ (eqn.6).}$$

Here, A_c represents cosine amplitude, A_s represents sine amplitude, and B is the DC bias. SPV was fitted on a cycle-by-cycle basis. In most cases, several cycles were manually excluded from further analysis, as their fit did not accurately characterize the

[8] In this experiment, a "trial" was defined as either a 0.5 Hz or 1.0 Hz steady state motion. Each measurement consisted of two trials of the same frequency.

measured data. The remaining fits were then averaged to produce eye amplitude (A_{trial}) for each trial. In this example, A_{trial} is a complex value, where A_s equals the real component, and A_c equals the imaginary component.

$$A_{trial} = \overline{A}_s + \overline{A}_c i \quad \text{(eqn. 7)}$$

In the same manner amplitude for somatosensory bar position, device position, and device velocity were calculated.

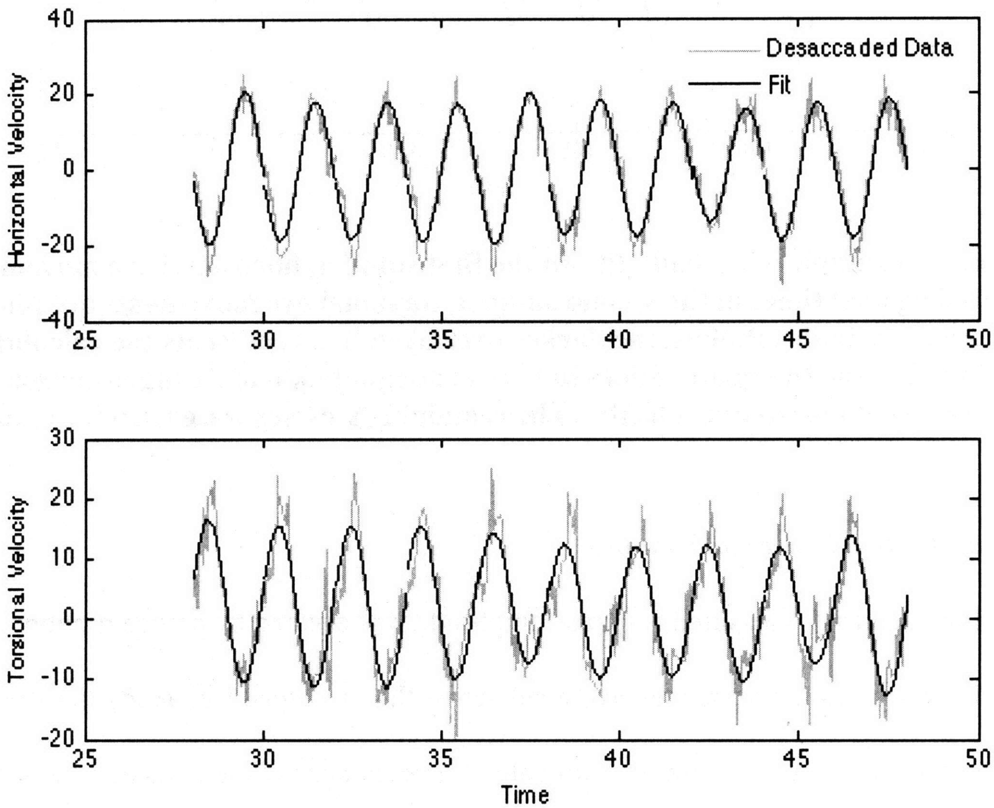


Figure 5: An example of a "good" horizontal and torsional fit. In the first subplot, horizontal eye movements are plotted against time; in the second subplot, torsional eye movements are plotted against time. In both subplots, the darker overlayed line represents the calculated "fit." In this example, all 10 cycle fits were included for analysis.

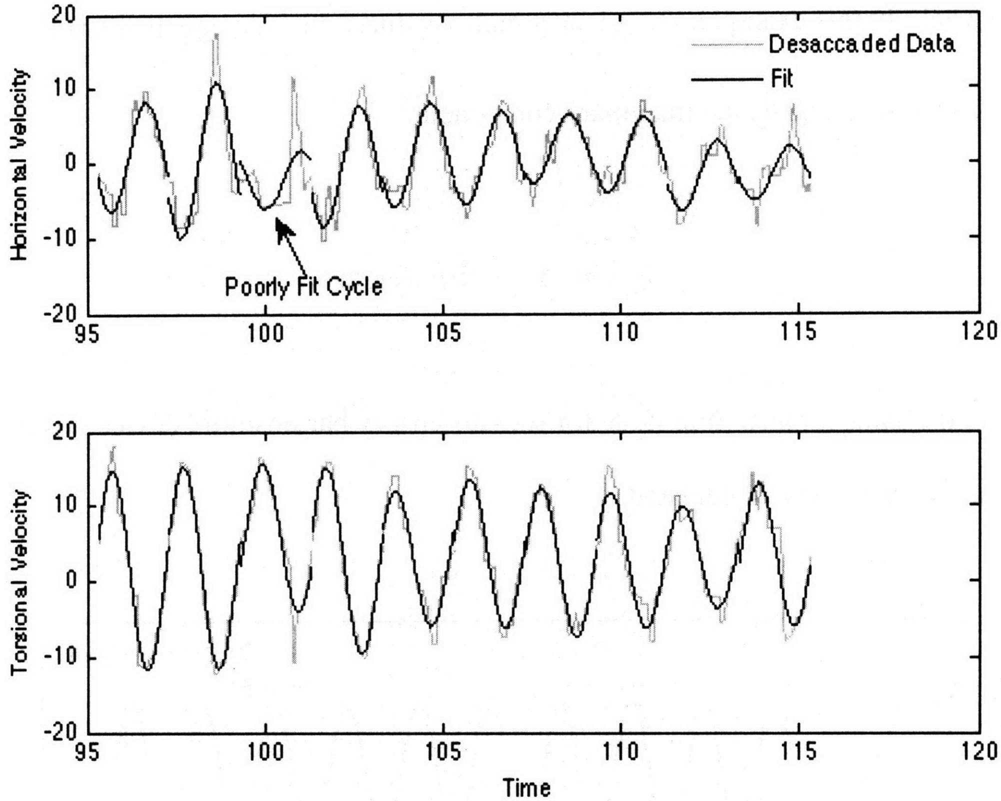


Figure 6: An example of a "bad" fit. In the first subplot, horizontal eye movements are plotted against time; in the second subplot, torsional eye movements are plotted against time. In both subplots, the darker overlayed line represents the calculated "fit." Note that the fit equation does not do an adequate job of fitting a sinusoid to the third cycle of horizontal velocity. The remaining 9 cycles were used for analysis.

Gain and Absolute Phase Calculations:

For each trial, measured response amplitude was divided by device motion amplitude. These values were then averaged across the two equal frequency trials within a measurement session to generate a gain value for each measurement session (equation 8):

$$G_{measurement} = \frac{\dfrac{A^{response}_{trial1}}{A^{device}_{trial1}} + \dfrac{A^{response}_{trial2}}{A^{response}_{trial2}}}{2} \quad \text{(eqn. 8)}$$

Finally, $G_{measurements}$ were averaged across subjects within a common group (e.g. light, dark, experienced, or naïve subjects). By taking the absolute value and arctan of these complex values, group gain (G_{group}, equation 9) and group absolute phase (P_{group}, equation 10), respectively, were calculated.

$$G_{group} = \left|\overline{G}_{measurements}\right| \qquad P_{group} = \tan^{-1}(\overline{G}_{measurements}) \quad \text{(eqn. 9\& 10)}$$

Amplitude vs. Gain:

Gain reflects both the response and the stimulus (as opposed to the response alone). However, horizontal "gain" is not a pertinent calculation in this experiment, as it is defined by a horizontal eye response divided by a roll stimulus. For this reason, amplitude, not gain, is depicted in the subsequent figures presenting eye movement data[9]. Group horizontal amplitude was calculated by multiplying group horizontal "gain" by group device amplitude (equation 11).

$$A^{response}_{group} = G_{group} \times A^{device}_{group} \quad \text{(eqn. 11)}$$

[9] Torsional gain *is* an appropriate calculation; as such, a relevant gain scale appears on the right side of each torsion figure.

Standard Error

Response amplitude and phase standard error were derived from a two-dimensional covariance ellipses calculated from sine and cosine components ((Johnson and Wichern 1982), as cited by Merfeld (Merfeld et al. 2005a)) [10].

Statistical Analysis:

Multivariate analysis of variance (MANOVA) tests were used to identify statistical differences between subject groups in eye responses and bar responses, except where specifically indicated. The real and imaginary components of these complex values were considered two independent variables in the MANOVA analysis.

Data analysis:

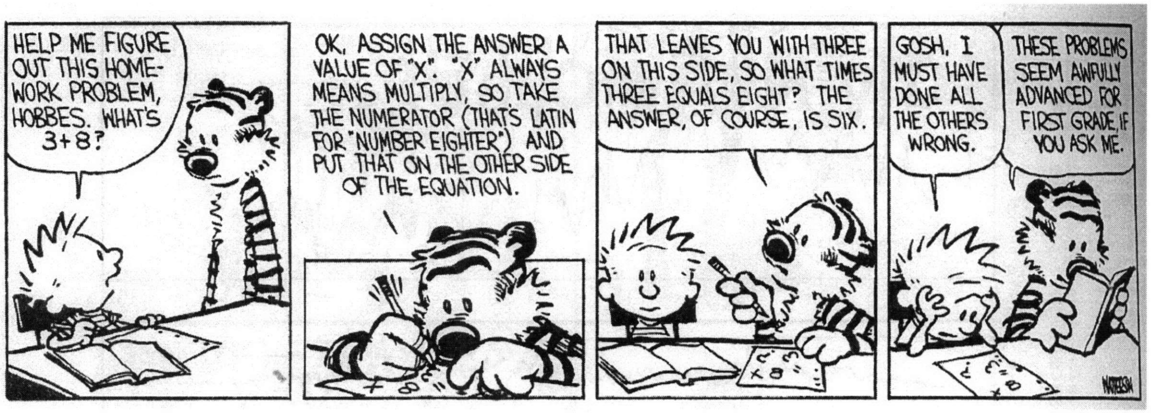
Preliminary analysis of the data revealed noise in the right eye torsion signal. Numerous attempts were made to try and eliminate the source of this noise; unfortunately, these efforts were not fully successful. For this reason, only left eye measurements were included in the final analysis.

In addition, only somatosensory bar and eye movement data recorded during 0.5 Hz trials were analyzed. 1.0 Hz eye data proved difficult to characterize, as these data could not be readily fit to a sinusoid.

---

[10] Gain standard error was proportional to amplitude standard error, since response standard error was much greater than tilt device standard error.

Results:



Vergence:

Vergence is known to affect horizontal VOR gain in humans (Paige et al. 1998). To ensure variations in vergence did not bias our gain calculations, subjects were shown an identical fixation target prior to each measurement test. This established vergence at a set angle prior to eye movement recording. Since vergence measured in darkness compares favorably with vergence previously measured in light (Merfeld et al. 2005a), we anticipated vergence to remain uniform across tests and across subjects.

Indeed, vergence angle did not vary to a great degree for most eye movement recordings (see figure 9). Vergence angle was plotted against uncorrected horizontal amplitude, and a line of best fit was calculated using the method of least squares (see figure 10). This regression line did not significantly differ from zero (R^2=0.013, F=2.12, p=0.148), suggesting vergence was not correlated with amplitude. In fact, the regression line's negative slope is the reverse of what is normally associated with vergence. This suggests vergence did not preferentially affect horizontal gain measurements.

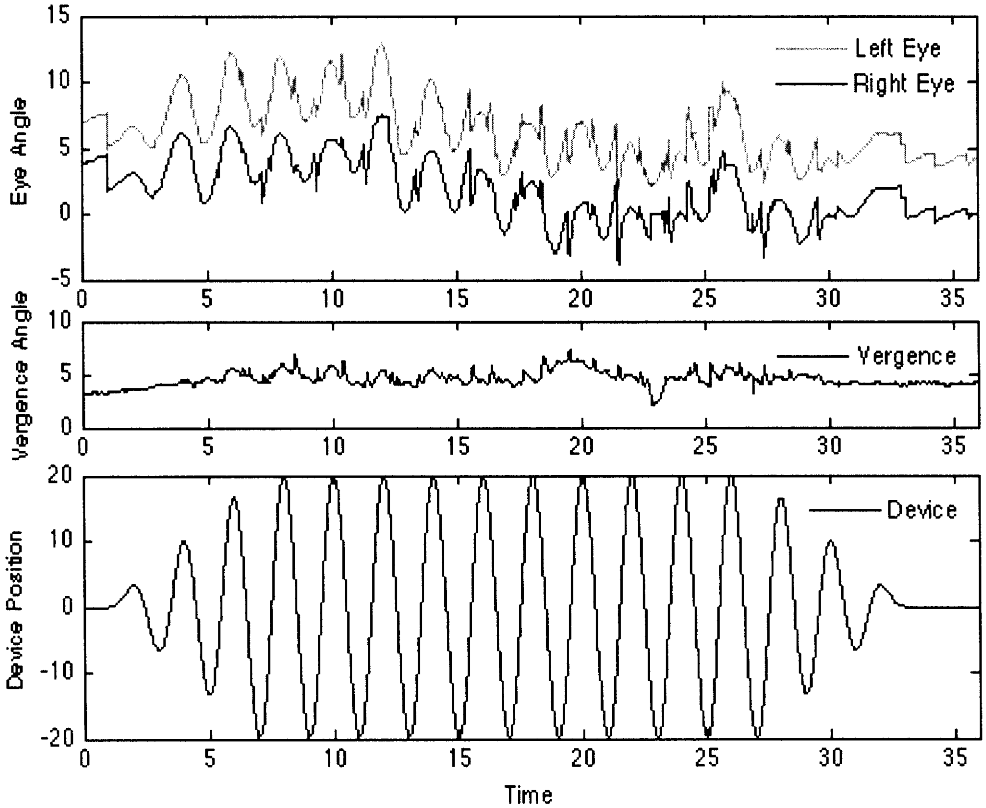


Figure 7: An example of vergence angle during 0.5 Hz motion. The top subplot represents the left and right eye angles over the course of one measurement session. The middle subplot represents the calculated vergence angle, and the bottom subplot represents the motion of the tilt device.

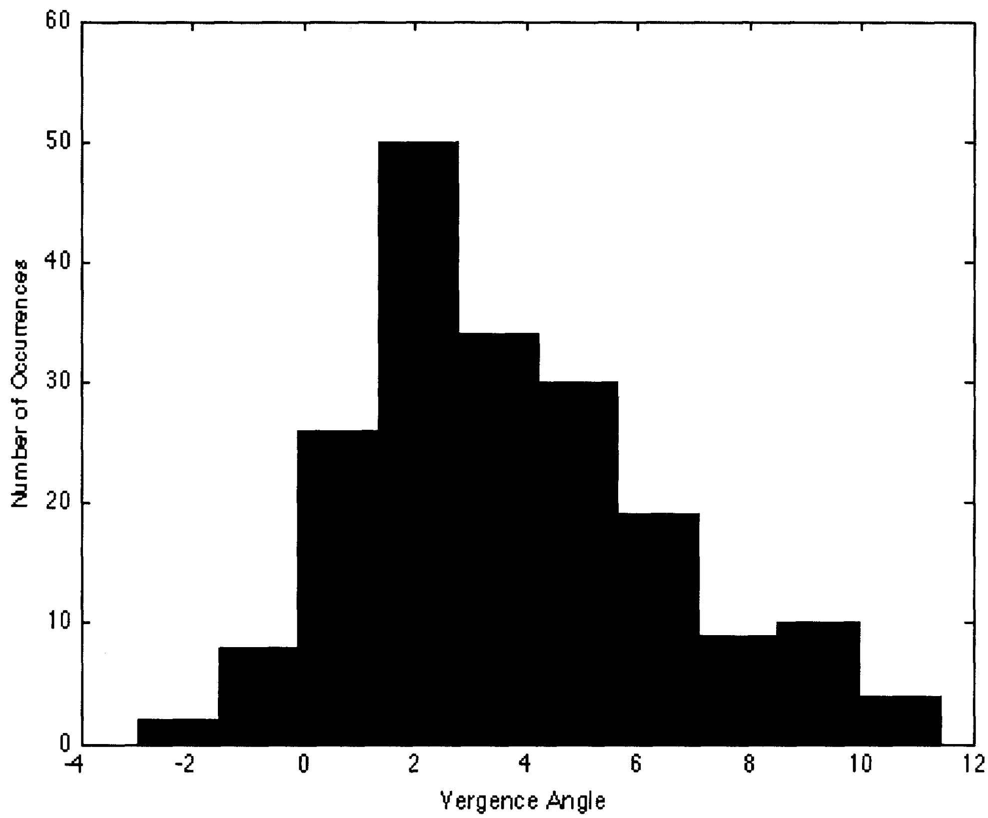


Figure 8: Number of vergence angle occurrences across 192 0.5 Hz trials. The roughly bell shaped curve suggests vergence was relatively constant from subject to subject and test to test.

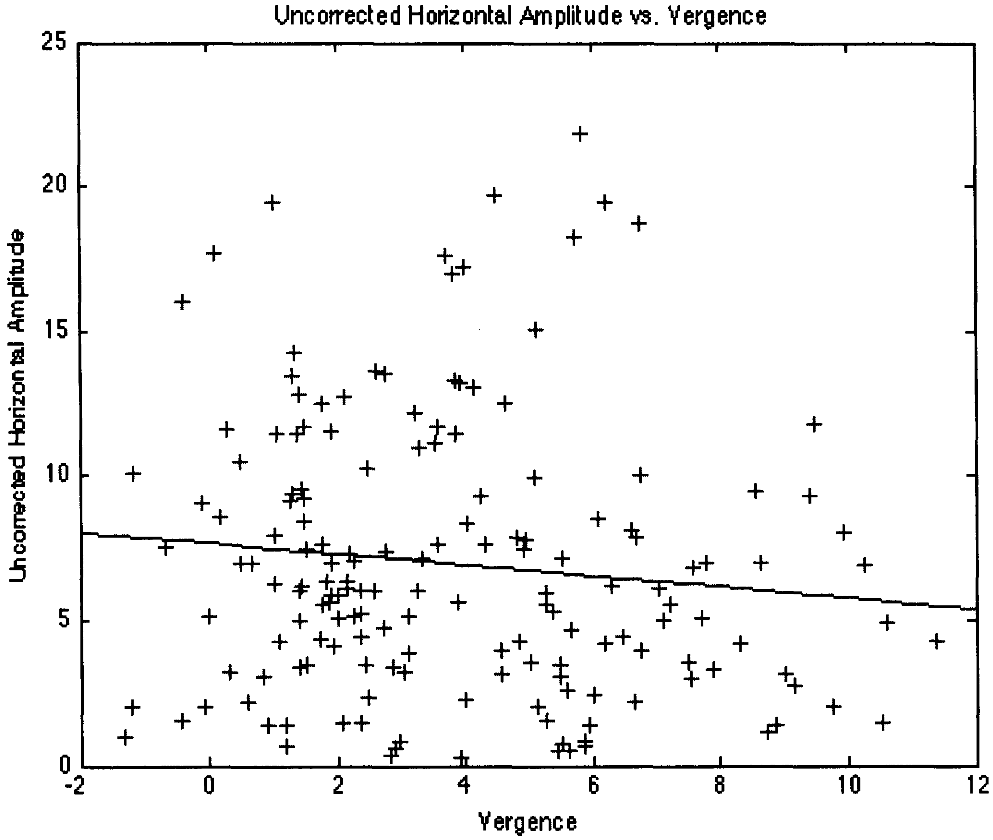


Figure 9: Uncorrected horizontal amplitude vs. vergence. The regression line to the data is not significant, suggesting vergence variability does not affect uncorrected horizontal amplitude.

Eye Movements—Overview:

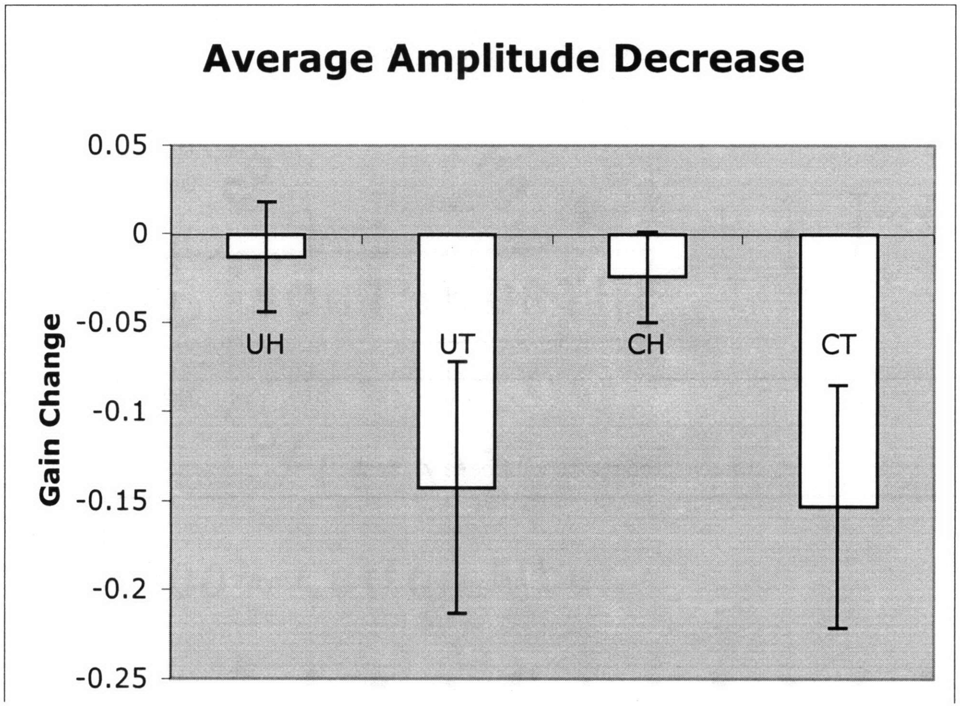


Figure 10: ***Change*** **in eye movement amplitude between the start of day 1 and the end of day 3. (UH=Uncorrected Horizontal, UT=Uncorrected Torsional, CH=Corrected Horizontal, CT=Corrected Torsional)**

A two-tailed paired t-test indicates an overall decrease in uncorrected torsional (UT, $p=0.04$) and corrected torsional (CT, $p=0.03$) amplitude between the first and last measurement sessions. Subjects on average exhibited nearly a 23% reduction in uncorrected torsional amplitude, and a 25% reduction in corrected torsional amplitude. Conversely, horizontal eye movements (both corrected and uncorrected) did not show a significant amplitude change (see figure 11).

Phase did not change between initial and final measurement sessions for any of the four eye movement types (two-tailed paired t-test).

Eye Movements--Light vs. Dark Overall Comparisons:

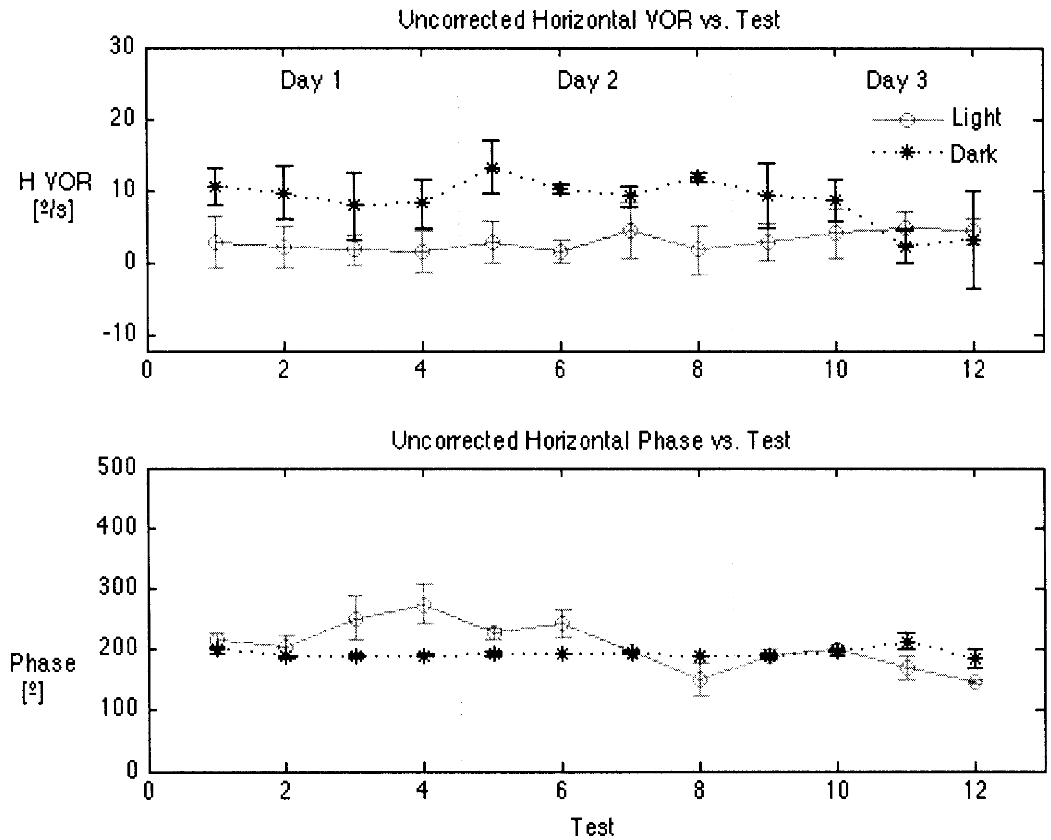


Figure 12: Uncorrected horizontal amplitude (top subplot) and phase (bottom subplot) vs. measurement session. Across the 12 measurement sessions, dark-adapted subjects showed significantly greater horizontal amplitude than light-adapted subjects. Error bars represent standard error of the mean.

Across the 12 measurement sessions, UH, UT, CH, and CT eye responses[11] were significantly different between subjects adapted in the light and subjects adapted in the dark (MANOVA, $p<0.01$, for all four eye movement types). Subjects who experienced the adaptation protocol in the dark (to be described from here on as "dark-adapted

[11] "Eye responses" refer to the average complex value computed for each test (see "Gain and Phase Calculations", p. 32). MANOVA tests that reach significance indicate complex values that exist in significantly different two-dimensional space, but do not imply that both gain and phase are significantly different. The results instead suggest that either the real and/or imaginary components and/or both of the complex values differ.

subjects) exhibited larger horizontal and torsional amplitude than "light-adapted subjects" (see figures 12, 13, 14, 15).

Surprisingly, these amplitude differences were evident even before the first adaptation session. Dark-adapted subjects expressed significantly larger UH ($p=0.009$), UT ($p=0.048$), and CT ($p=0.032$) amplitude during the first measurement session (one-tailed between subjects t-test). However, for both types of horizontal eye movements, these differences in amplitude disappear during measurements 11 and 12. Indeed, between subjects t-tests indicate that horizontal amplitude for light and dark-adapted subjects do not significantly differ during the last two measurement sessions.

Phase did not significantly differ for any of the four eye movement types.

Eye Movements—Light vs. Dark Horizontal Eye Movements:

When grouped by adaptation type, horizontal amplitude measured during the final measurement session (measurement 12) did not differ significantly from horizontal amplitude measured at the start of testing (measurement 1) for either light or dark-adapted subjects (one-tailed paired t-test, both UH and CH eye movements). These findings compare favorably with the absence of horizontal changes for subjects as a whole (see "Eye Movement—Overview" section, p.39).

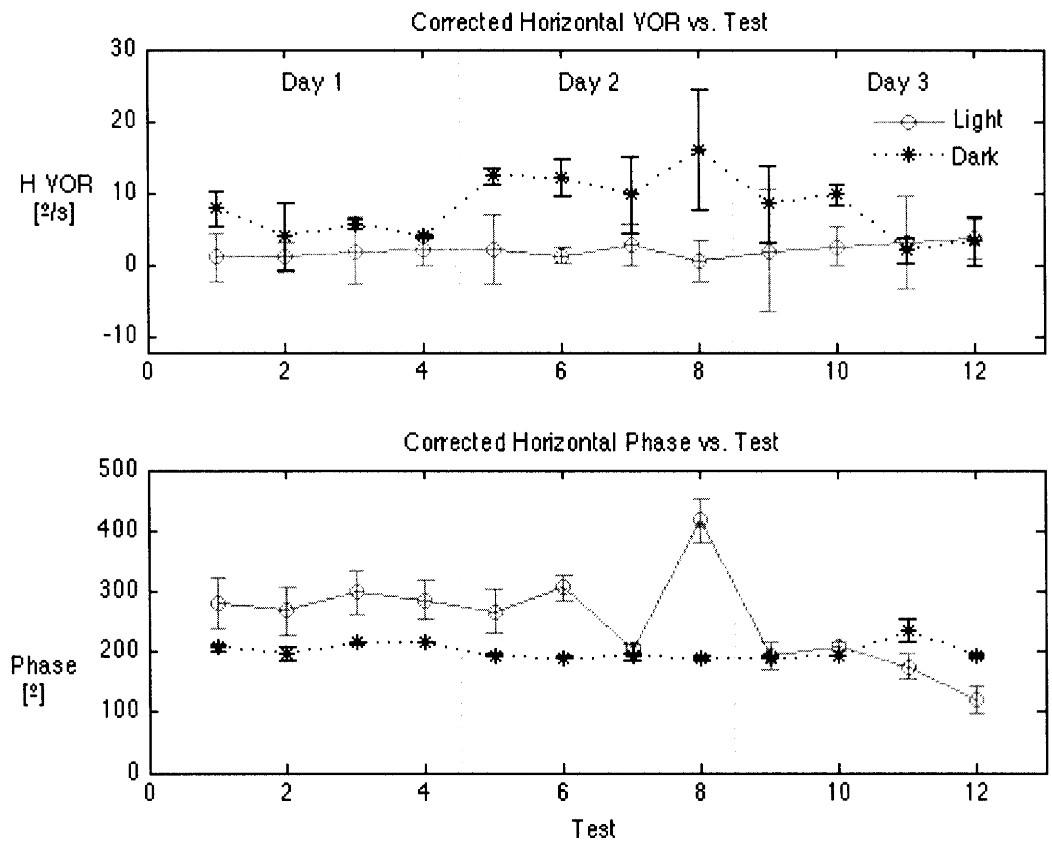


Figure 13: Corrected horizontal amplitude (top subplot) and phase (bottom subplot) vs. measurement session. Across the 12 measurement sessions, dark-adapted subjects showed significantly greater horizontal amplitude than light-adapted subjects. Error bars represent standard error of the mean.

Eye Movements—Light vs. Dark Torsional Eye Movements:

Overall, subjects showed a significant decrease in UT and CT amplitude from tests 1 to tests 12 (see "Eye Movement—Overview" section, p.39). However, when segregated by adaptation type, neither light nor dark subjects showed a significant change in either corrected or uncorrected torsional amplitude (one-tailed paired t-test).

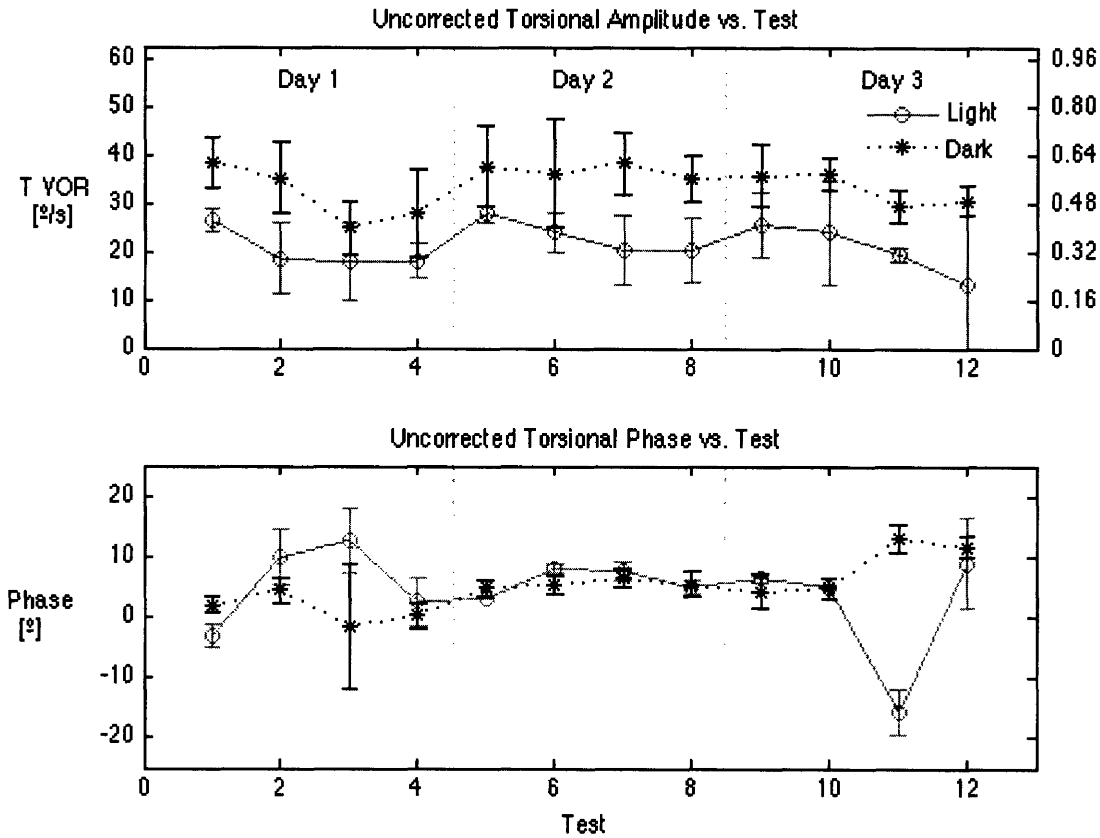


Figure 14: Uncorrected torsional amplitude (top subplot) and phase (bottom subplot) vs. measurement session. Across the 12 measurement sessions, dark-adapted subjects showed significantly greater torsional amplitude than light-adapted subjects. Note the sawtooth pattern exists for both groups of subjects. Error bars represent standard error of the mean.

Torsional gain for light and dark subjects exhibited a sawtooth-like pattern across the 12 measurement sessions (see figure 14, 15). Gain decreases within days were offset by gain increases between days. To assess whether these changes were significant, regression lines for light and dark subjects were fitted to amplitude values measured during each day of testing. In addition, regression lines were fitted to amplitude values obtained during the last measurement session of the day and the first measurement session of the following day. Specifically, lines of best fit were calculated for amplitude values measured between tests 1 and 4, tests 4 and 5, tests 5 and 8, tests 8 and 9, and tests 9 and 12, using the method of least squares (see figure 16).

For subjects who experienced adaptation in the dark, amplitude showed a significant change between tests 1 and 4, and between tests 4 and 5, for both uncorrected and corrected torsional eye movements (regression analysis, UT, tests1:4, $p=0.038$; UT, tests 4:5, $p=0.04$; CT, tests 1:4, $p=0.04$; CT, tests 4:5, $p=0.05$). The slope of the remaining regression lines for dark-adapted subjects did not differ significantly from zero. Additionally, none of the fitted regression lines for light adapted subjects reached significance.

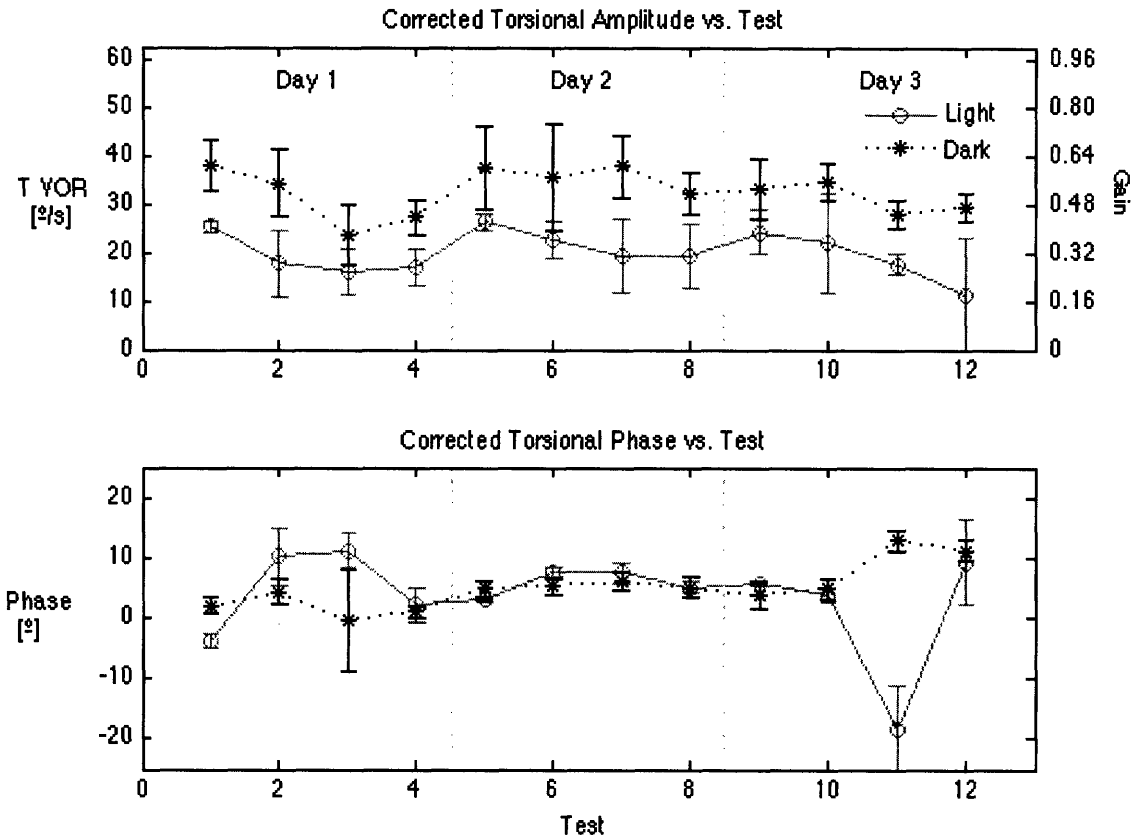


Figure 15: Corrected torsional amplitude (top subplot) and phase (bottom subplot) vs. measurement session. Across the 12 measurement sessions, dark-adapted subjects showed significantly greater torsional amplitude than light-adapted subjects. Note the sawtooth pattern exists for both groups of subjects. Error bars represent standard error of the mean.

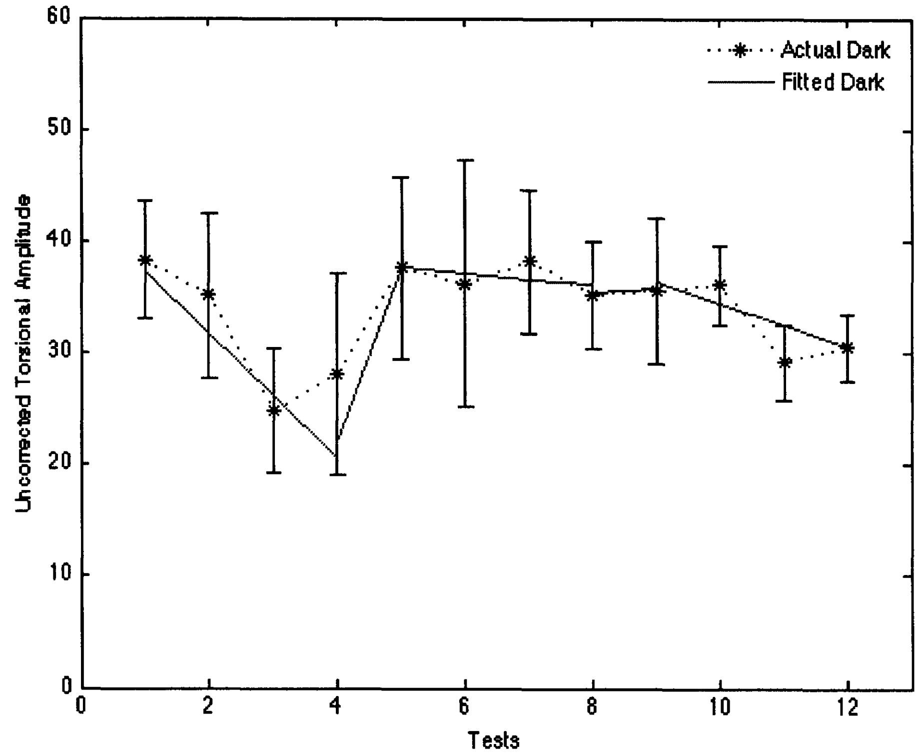


Figure 16: An example of how regression lines were fitted to torsion data across test sessions. Regression lines were calculated using the method of least squares.

Eye Movements--Experienced vs. Naïve Overall Comparisons[12]:

Across the 12 measurement sessions, experienced subjects exhibited significantly greater uncorrected and corrected horizontal eye movements than naïve subjects (MANOVA, UH: $p<0.01$, CH: $p<0.01$). However, neither uncorrected nor corrected torsional eye movements were significantly affected by experience level.

[12] The evidence suggests that differences in light and dark-adapted subjects may be a function of subject grouping, as opposed to an effect of adaptation (for a thorough explanation of why this may be, see Eye Movement Response Characteristics—Light vs. Dark, p. 58). For this reason, subjects were collapsed across the adaptation variable during our analysis of experience on horizontal and torsional eye movements. This course of action had the added benefit of substantially simplifying the MANOVA analysis.

Eye Movements—Experienced vs. Naive Horizontal Eye Movements:

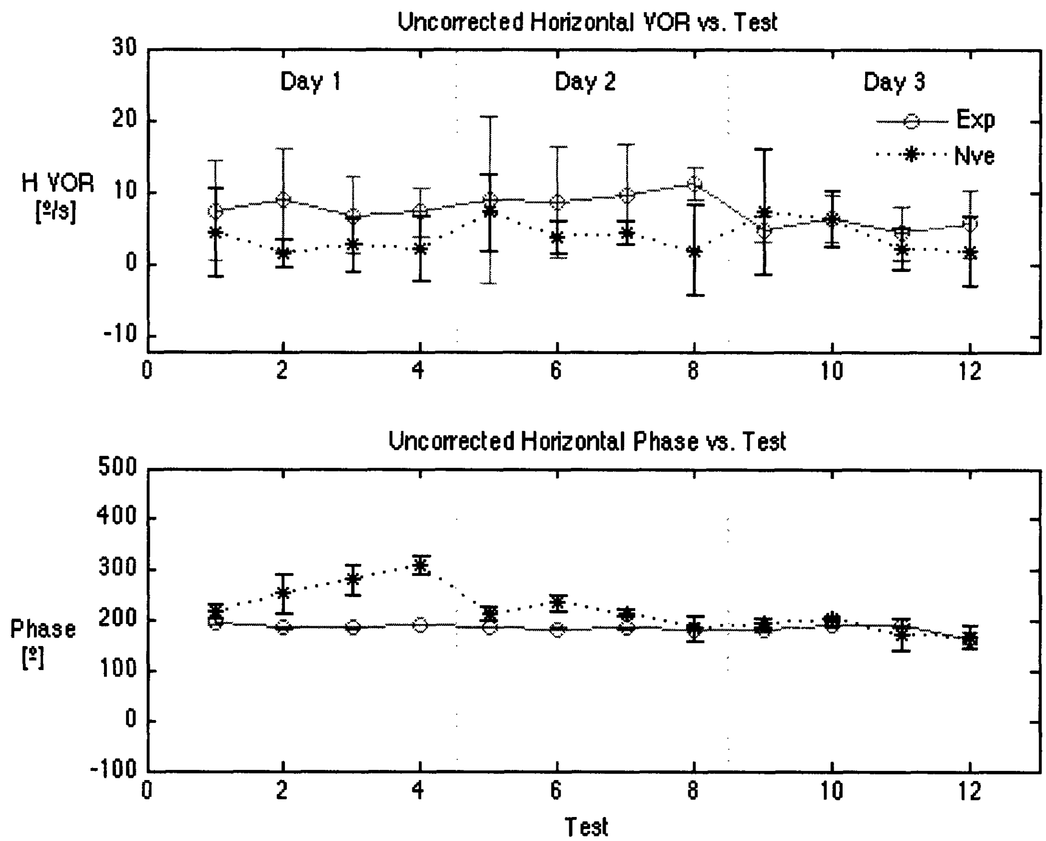


Figure 17: Uncorrected horizontal amplitude (top subplot) and phase (bottom subplot) vs. measurement session. Across the 12 measurement sessions, experienced subjects showed significantly greater horizontal amplitude than naive subjects. Error bars represent standard error of the mean.

At the start of testing, experienced and naïve subjects exhibited similar UH and CH amplitude (2-tailed between subject t-test). As testing progressed, neither experienced nor naïve subjects showed a significant change in UH or CH amplitude (one-tailed paired t-test, comparing measurement 1 with measurement 12).

While MANOVA tests indicate UH eye movements differed significantly between experienced and naïve subjects across the 12 measurement sessions, it should be noted that when grouped by day, UH amplitude differed between experienced and naïve subjects on days 1 and 2, but *not* on day 3.

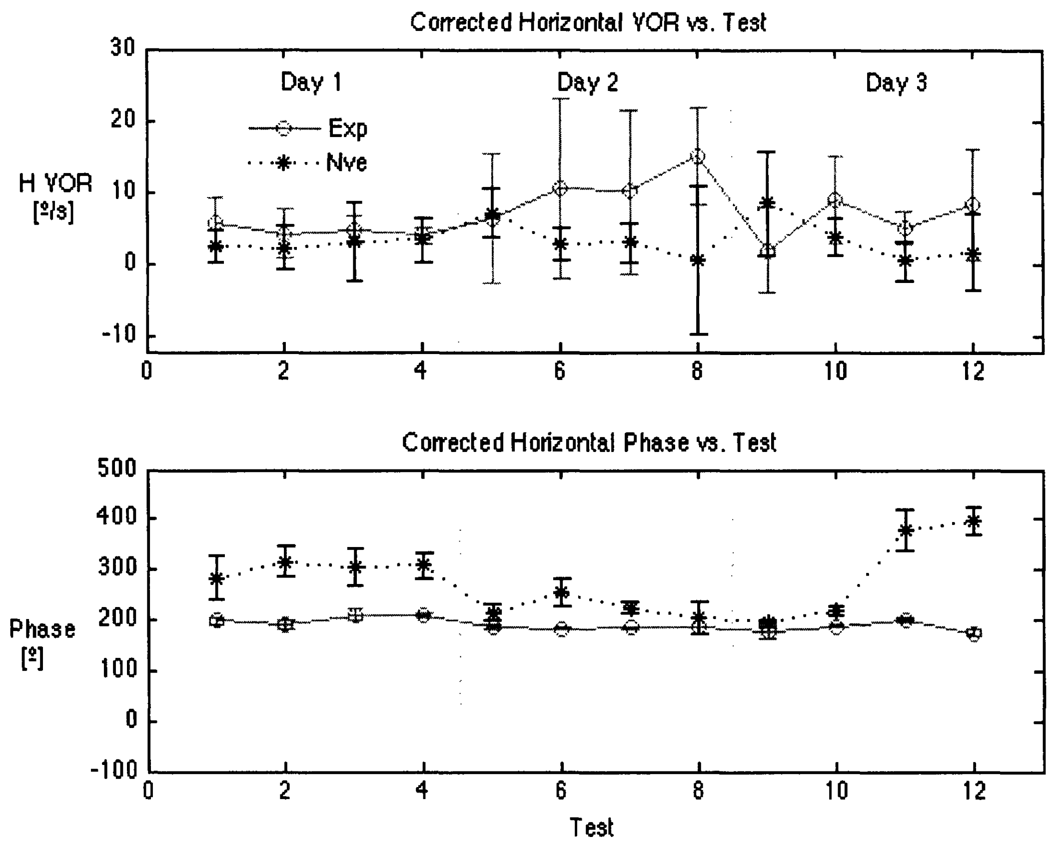


Figure 18: Corrected horizontal amplitude (top subplot) and phase (bottom subplot) vs. measurement session. Across the 12 test sessions, experienced subjects showed significantly greater horizontal amplitude than naive subjects. Error bars represent standard error of the mean.

Eye Movements—Experienced vs. Naïve Torsional Eye Movements:

Experienced and naïve subjects did not exhibit significantly different UT or CT amplitude during measurement 1. In addition, neither experienced nor naïve subjects showed a significant decrease in either corrected or uncorrected torsional amplitude between the first and last measurement sessions. (one-tailed paired t-test). This contrasts with the decrease in torsional amplitude found across all 8 subjects (see "Eye Movements—Overview" section, p.39).

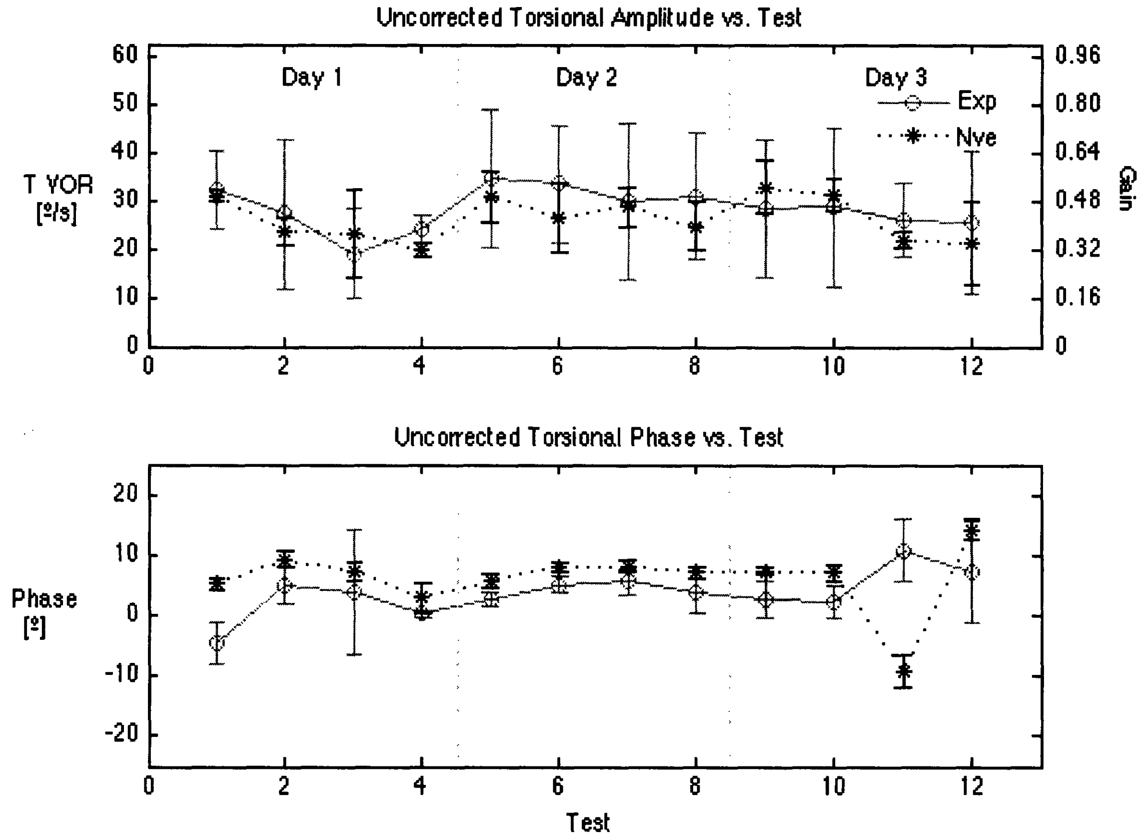


Figure 19: Uncorrected torsional amplitude (top subplot) and phase (bottom subplot) vs. test session. Across the 12 measurement sessions, experienced subjects did not significantly differ than naïve subjects. Note the sawtooth pattern exists for both groups of subjects. Error bars represent standard error of the mean.

Torsional amplitude for naïve and experienced subjects exhibited a sawtooth-like pattern. As with the light-dark analysis, regression lines were fitted to measurement sessions within days and between days to test for significant changes. Corrected and uncorrected torsional eye movements showed significant changes between tests 1 and 4, but only for naïve subjects (UT 1:4, p=0.01; CT 1:4, p=0.02).

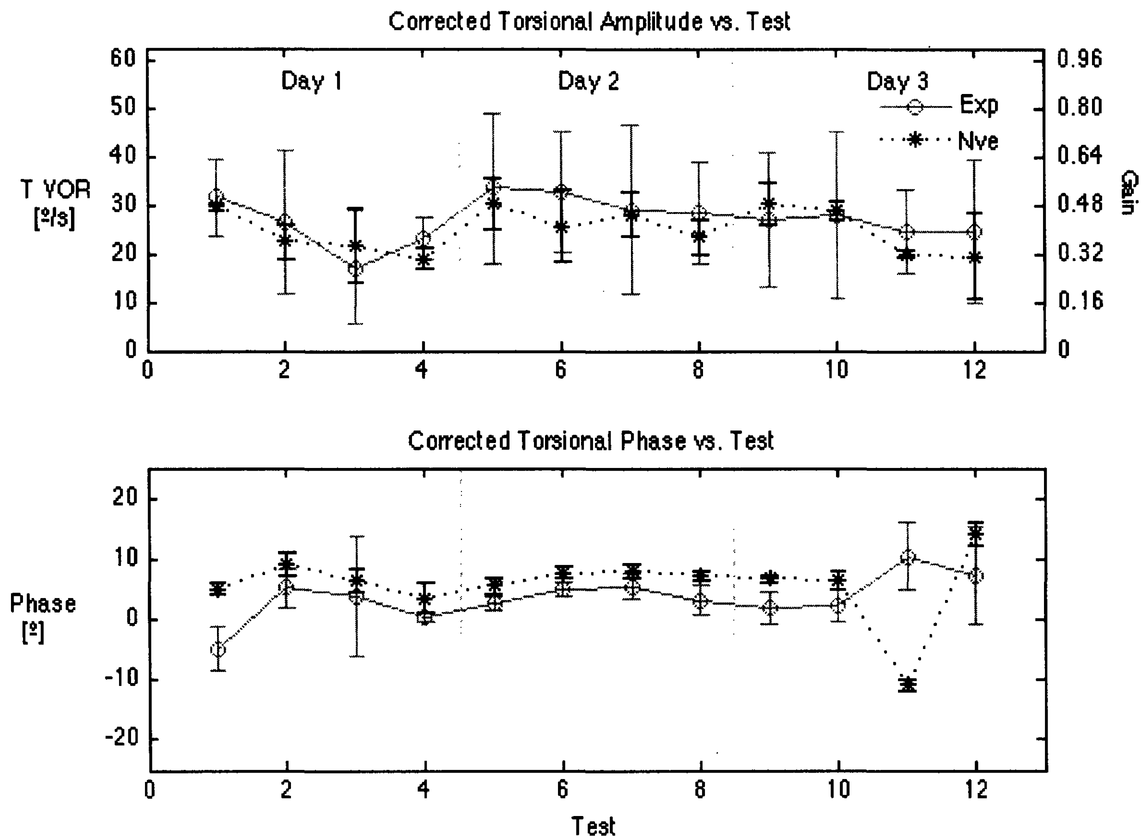


Figure 20: Corrected torsional amplitude (top subplot) and phase (bottom subplot) vs. test session. Across the 12 measurement sessions, experienced subjects did not significantly differ than naïve subjects. Note the sawtooth pattern exists for both groups of subjects. Error bars represent standard error of the mean.

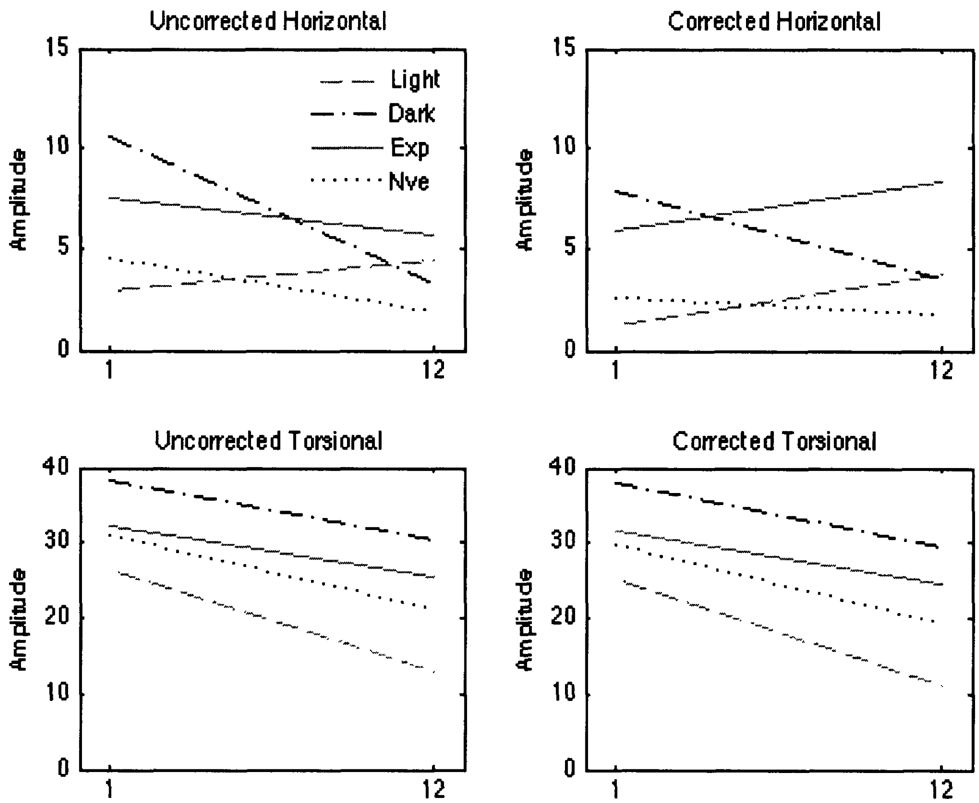


Figure 11: Overview of the ocular response for each of the four "groups" of subjects. Clockwise from top left: uncorrected horizontal amplitude vs. measurement session, corrected horizontal amplitude vs. measurement session, corrected torsional amplitude vs. measurement session, and uncorrected torsional amplitude vs. measurement session.

Somatosensory Bar--Overview:

As a whole, the 8 subjects did not show a significant change in somatosensory bar gain between the first and last measurement session (two-tailed, paired t-test). In addition, gain remained relatively constant across subjects and across measurement sessions. At the start of testing, gain was 0.7; at the end, average gain was 0.77.

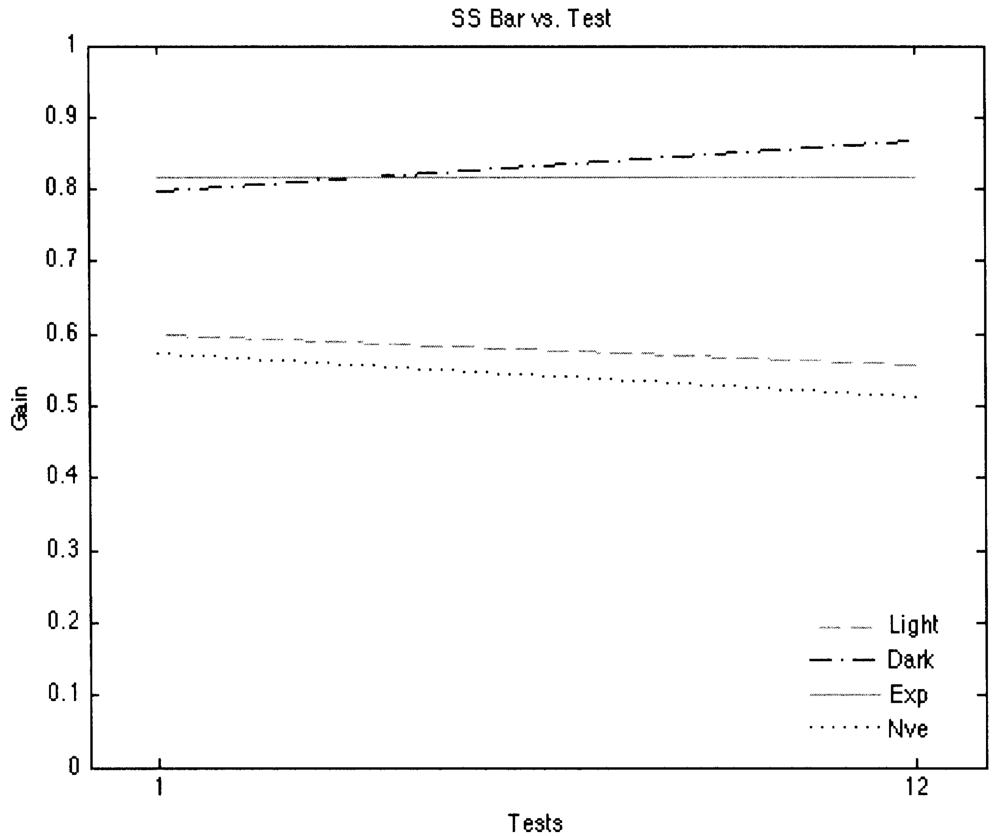


Figure 22: An overview of the somatosensory bar gain changes between measurements 1 and 12, for all four "groups".

Somatosensory Bar—Light vs. Dark Overall Comparisons:

The somatosensory bar showed an effect of adaptation type across the 12 test sessions (MANOVA, $p<0.01$). Indeed, plots of both gain and phase for show a larger amplitude and greater phase lag for dark-adapted subjects. However, unlike eye responses, somatosensory gain did not show a significant difference between light and dark subjects for test 1 (two-tailed between subjects t-test), though large initial differences are clearly evident (see figure 23).

When grouped by adaptation type, neither light nor dark-adapted subjects showed a significant change in somatosensory bar gain between measurement 1 and measurement 12 (two-tailed paired t-test).

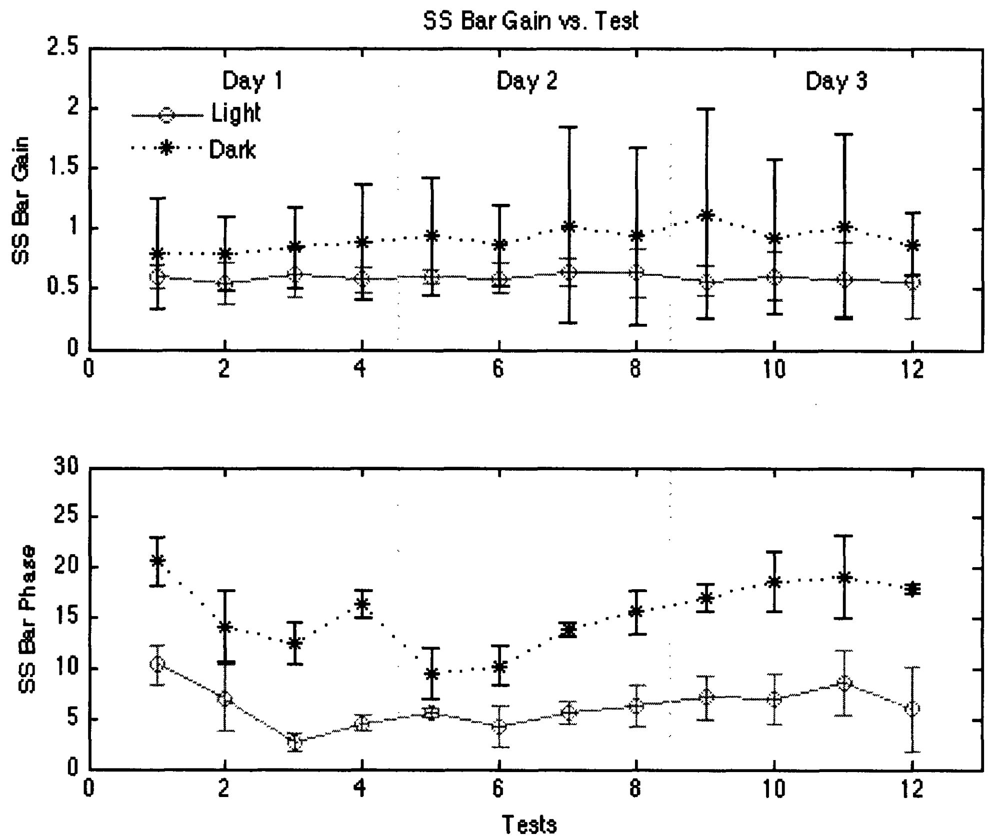


Figure 23: Somatosensory bar gain (top subplot) and phase (bottom subplot) vs. measurement session. Across the 12 measurement sessions, dark-adapted subjects showed significantly greater somatosensory bar responses than light-adapted subjects. Error bars represent standard error of the mean.

Somatosensory Bar—Experienced vs. Naïve:

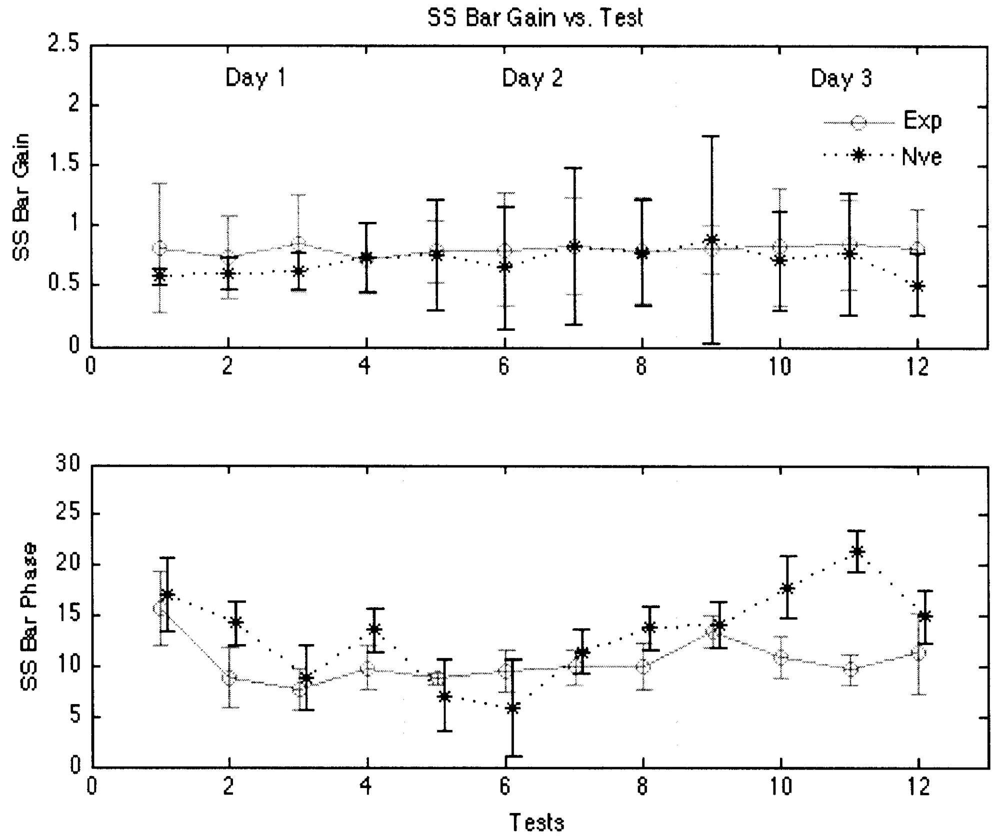


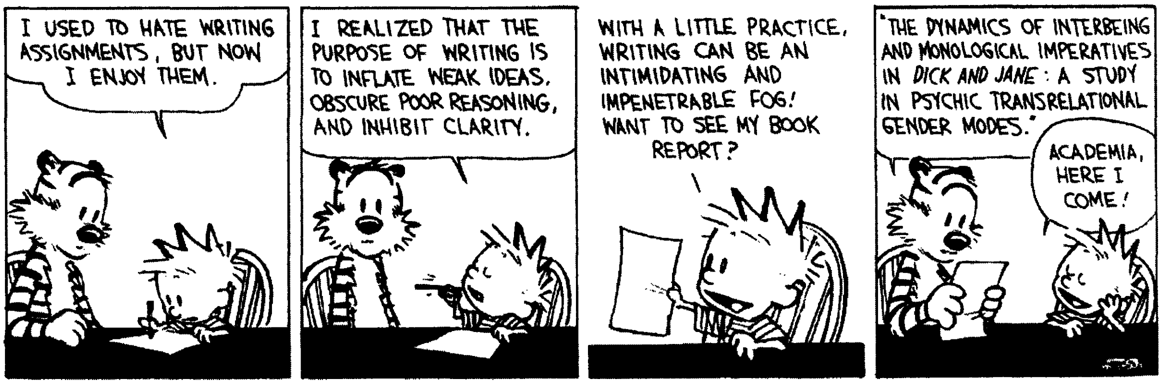
Figure 24: Somatosensory bar gain (top subplot) and phase (bottom subplot) vs. measurement session. Across the 12 measurement sessions, experienced and naïve subjects did not show a significantly different somatosensory bar gain. Error bars represent standard error of the mean.

As with eye movements, bar responses were collapsed across the light/dark variable during analysis of experience. Across the 12 testing sessions, experienced subjects did not exhibit a significantly different somatosensory response than naïve subjects (MANOVA). Additionally, for the first measurement session, gain did not significantly differ between the two groups of subjects. Furthermore, gain did not significantly change between tests 1 and tests 12 for either experienced or naïve subjects (two-tailed paired t-test).

	UH	CH	UT	CT	SSBar
Light vs. Dark Responses	D, $p<0.01$	D, $p<0.01$	D, $p<0.01$	D, $p<0.01$	D, $p<0.01$
Experienced vs. Naïve Responses	E, $p<0.01$	E, $p<0.01$	NO	NO	NO
Overall 1 vs 12 Amplitude	NO	NO	1, $p=0.04$	1, $p=0.03$	NO
Light 1 vs 12 Amplitude	NO	NO	NO	NO	NO
Dark 1 vs 12 Amplitude	NO	NO	NO	NO	NO
Experienced 1 vs 12 Amplitude	NO	NO	NO	NO	NO
Naïve 1 vs 12 Amplitude	NO	NO	NO	NO	NO

Figure 25: Significant Response Overview. Cells shaded in black indicate a significant change, with the first letter symbolizing the larger value, and the first number indicating the significance value. Cells shaded in white are not significant ($p > 0.05$).

Discussion:



Eye Movement Response Characteristics—Overview:

Inappropriate horizontal VOR was expected to decrease over the course of testing as subjects became acclimated to roll tilt stimuli. Torsional VOR was also predicted to decrease over the 12 measurement sessions. In this study, eye movement responses showed a somewhat different effect: while torsional amplitude significantly decreased between measurement 1 and measurement 12, horizontal amplitude remained unchanged.

The measured decline in torsional amplitude is not without precedent. Anecdotally, pilots flying under instrument flight rules (IFR) learn to inhibit torsional eye movements during roll so as to accurately fixate on instruments within the aircraft[13]. In a similar fashion, fixation on a target attached to the tilt device may have triggered a decline in torsional eye movements. During the adaptation sessions, light-adapted subjects were presented with a target rolling synchronously with them (in this case, the moving arm of the tilt device). Though the target was not visible during measurement sessions conducted in the dark, it is reasonable to assume subjects were aware the target

[13] IFR pilots rely solely on aircraft instruments—not outside references—for geographic and spatial orientation. Torsional eye movements are undesirable during roll, as torsion stabilizes the eye with respect to the horizon, not with respect to the aircraft.

was continuing to move with them[14]. Under this premise, a reduction in torsional amplitude is the appropriate adaptive response, as the visual cues remained stationary with respect to the tilting subject.

Horizontal amplitude was small but non-negligible during the pre-adaptation period (measurement 1). This finding is not unexpected, as the presence of horizontal eye movements epitomizes the ambiguity inherent to the GIF vector (Angelaki et al. 2000; Lichtenberg et al. 1982). The consistency of horizontal eye movements across the 12 measurement sessions, however, is more difficult to interpret. Previous studies have shown reductions in horizontal gain can occur anytime between 30 minutes (Faulstich et al. 2004; Watanabe et al. 2003) and 2 hours (Angelaki and Hess 1998; Wei and Angelaki 2001). In this investigation, neither uncorrected nor corrected horizontal eye movements displayed a significant decrease in amplitude, despite adaptation sessions cumulatively lasting over 2 hours.

Conceivably, the uniformity of horizontal eye movements across test sessions may stem from limitations inherent to the statistical analysis, as opposed to a true absence of change. When initial values are low, as they were for horizontal amplitude, statistical reductions can prove impossible to achieve[15]. However, calculations show that floor effects did not impede statistical reductions, as even a 10% horizontal gain reduction (from 6.96°/s to 6.27°/s for UH, and from 7.51°/s to 6.75°/s for CH) constitutes a

[14] To a lesser extent, this effect also holds true for dark-adapted subjects. Though they did not experience adaptation in the light, they were exposed to the visual scene, for a very short period, after adaptation sessions. During this time, dark-adapted subjects likely acquired the same target to fixate on as light-adapted subjects.
[15] As amplitude can never be less than zero, there is a physical limit to how much it can be reduced. This is known statistically as a floor effect.

significant decrease in amplitude. These decreases were certainly plausible from a physical standpoint.

This leads us to conclude three days exposure to sum of sines roll tilt *does not* produce a significant horizontal VOR change. The motion paradigm was of sufficient duration and intensity to trigger a torsional amplitude decrease, but was incapable of stimulating a similar decline in inappropriate horizontal amplitude.

VOR adaptation is not perfect, nor is it absolute. In microgravity, where translation is the only source of stimulation for the otoliths, torsional eye movements remain (Young et al. 1984). Perhaps in a similar manner, inappropriate horizontal eye movements endure after repeated roll tilt motion. Presumably these inappropriate responses are retained for re-adaptive purposes. While small horizontal and torsional eye movements do not substantially destabilize the image during roll tilt or spaceflight, respectively, they contribute to image stabilization during normal motion paradigms, which predominate.

Eye Movement Response Characteristics—Light vs. Dark:

The presence or absence of visual cues during adaptation does not appear to influence torsional or horizontal eye movements during 0.5 Hz roll tilt motion. While MANOVA analyses indicate light and dark-adapted subjects differ significantly in UH, CH, UT, and CT eye responses post-adaptation, it is relevant to note these differences existed even during the first measurement session[16]. This suggests that an irregular partitioning of subjects—not necessarily the adaptation variable itself—contributed to

[16] Theoretically, subjects should show similar responses during measurement 1, regardless of grouping variable, as adaptation has yet to occur.

overall eye response differences between light and dark-subjects. Indeed, if visual cues do have an effect on eye responses, differences between light and dark-adapted subjects recorded during measurement 1 should change with time. Tellingly, these differences remained constant across the 12 measurement sessions. Neither light nor dark-adapted subjects exhibited a significant change in UH, CH, UT, or CT amplitude between the start and end of testing

Arguably, the presence of visual cues during adaptation may *not* alter the VOR response. This assertion, however, contradicts a number of studies that have found an effect of visual cues on VOR gain (Angelaki and Hess 1998; Faulstich et al. 2004; Jones 1977; Paige and Sargent 1991; Watanabe et al. 2003; Wei and Angelaki 2001).

What then could have led to such irreconcilable findings? As noted previously, the select group of subjects who participated in adaptation in the light differed from their dark-adapted peers even *before* the start of testing. Logically, these subjects may also have differed from the normal population in their ability to adapt to roll tilt stimuli. Therefore, conclusions based on the presence or absence of visual cues *must* be interpreted with these a priori differences in mind.

Eye Movement Response Characteristics—Light vs. Dark Torsion

MANOVA analyses indicate light-adapted subjects exhibited smaller torsional eye movements across 12 measurement sessions than their dark-adapted peers. However, because both groups of subjects differed in their torsional responses at the start of testing, and because neither group showed a significant change in torsional amplitude between measurements 1 and 12, discussion of statistical differences must be interpreted with caution. Presumably, innate differences between groups, not visual cues, were responsible for the torsion disparities between light and dark-adapted subjects.

Eye Movement Response Characteristics—Light vs. Dark Sawtooth Response:

Torsional amplitude showed a significant decrease between measurements 1 and 4, and a significant increase between measurements 4 and 5 for dark-adapted subjects. However, after measurement 5, both uncorrected and corrected torsional eye movements remained constant for the remainder of the experiment.

The reduction of torsional eye movements during the first day of testing could exemplify a type of *non-associative* learning known as habituation, which "yields reduced responses to sensory stimuli following repeated exposure to the stimuli" (Merfeld et al. 2007). VOR habituation occurs in a number of species, including goldfish (Anastasio 2001) and humans (Ahn 2003).

Conceivably, this reduction could *also* be classified as adaptation. Adaptation, unlike habituation, requires learning to be *associative*. During the first day of testing, dark-adapted subjects may have associated afferent canal and otolith cues with the roll tilt motion. This may have triggered an adaptive torsional response, similar to the reduced vertical response to cross-coupled head turns seen during whole-body centrifugation (Young et al. 2003). Unfortunately, due to limitations inherent to the methodology, this study cannot definitively state what role adaptation or habituation played in influencing the VOR response characteristics.

Eye Movement Response Characteristics—Naïve vs. Experienced:

Experience, as categorized by the motion questionnaire, was predicted to negatively correlate with uncorrected horizontal amplitude. Instead, the reverse held true: during the first two days of testing, UH amplitude for experienced subjects was significantly greater than UH amplitude for naïve subjects. This finding suggests experience may *not* actually improve the horizontal VOR response.

This explanation contradicts past research. Several studies have found a favorable effect of experience on VOR response characteristics (Ahn 2003; Lee et al. 2004). Conceivably, the categorization of naïve and experienced subjects may have been inadequate. Perhaps subjects classified as naïve had experience with atypical motion not identified by the motion questionnaire; that, in fact, "naïve" subjects had greater experience with motion than "experienced" subjects. However, given the disparity in pilot experience between "naïve" and "experienced" subjects, and the strong correlation between flight training and VOR enhancement, it is unlikely that the classification system was inappropriate.

Furthermore, as "naïve" subjects gained experience with the motion paradigm, inappropriate UH eye movements increased in amplitude. This finding *cannot* be attributed to an inappropriate classification system. Therefore, it appears that the roll tilt stimuli used herein to yield adaptation did not improve the horizontal VOR.

Eye Movement Response Characteristics—Naïve vs. Experienced Sawtooth Response:

Naïve subjects expressed a "sawtooth" pattern of torsional amplitude change between measurements 1 and 4, and between measurements 4 and 5. This pattern of amplitude change is similar to what is seen in dark-adapted subjects, and provides further evidence for the theory that torsional eye responses either habituate or adapt during the first day of testing for subjects without experimental advantages[17].

Across 12 measurement sessions, naïve and experienced subjects showed similar torsion responses. This indicates experience, as defined in this study, did not significantly improve the torsional response to motion.

Somatosensory Bar Overview:

Previous studies have shown that humans underestimate tilt at frequencies above 0.05 Hz (Park et al. 2006). Thus, the average somatosensory bar gain for the first measurement session (0.70), though less than unity, does not appreciably differ from published values. This suggests improvements in somatosensory gain, limited as they were by statistical and physical ceilings, were unlikely. Indeed, during the last measurement session, gain had increased only slightly (from 0.70 to 0.77).

Somatosensory Bar Characteristics—Light vs. Dark:

Human motion perception heavily depends on visual cues, particularly at low frequencies ((Dichgans et al. 1972; Zupan and Merfeld 2003), as cited by (Merfeld et al. 2005a)). The availability of both visual and vestibular cues during adaptation was thus predicted to improve spatial perception; light-adapted subjects were expected to exhibit gain values closer to unity than dark-adapted subjects.

---

[17] Experienced subjects have the advantage of experience, and light-adapted subjects have the advantage of visual cues.

While visual cues did not hinder motion perception per se (gain values for light-adapted subjects remained constant between measurements 1 and measurements 12), it is surprising that visual cues did not serve to *improve* spatial perception. Indeed, across all 12 measurement sessions, subjects who experienced adaptation in the dark showed somatosensory bar gain values closer to unity than subjects who experienced adaptation in the light. Notably, somatosensory bar gain did not differ for light and dark-adapted subjects during measurement 1, suggesting that gain differences for subsequent measurement sessions were due to the visual cue variable, not to group differences established prior to testing.

Conceivably, dark adapted subjects may be more sensitive to motion in the dark than their light-adapted peers. As mentioned previously, humans typically base their perception of motion on visual signals ((Dichgans et al. 1972; Zupan and Merfeld 2003), as cited by(Merfeld et al. 2005a)). With visual cues absent, dark-adapted subjects were forced to rely entirely on vestibular, proprioceptive, and kinesthetic cues for spatial orientation. This increased reliance on non-visual cues may have carried over to the measurement sessions, which were also conducted in the dark. Such carryover could explain the larger somatosensory bar gain values exhibited by dark-adapted subjects.

Somatosensory Bar Characteristics—Experienced vs. Naïve:

Spatial perception, as measured by somatosensory bar response, was largely unaffected by experience level. Experienced and naïve subjects showed similar somatosensory bar gain across the 12 measurement sessions. Furthermore, somatosensory bar gain did not differ between measurement 1 and measurement 12 for either experienced or naïve subjects.

This finding has two potential explanations: either (1) naïve and experienced subjects do not differ in their perception of motion, or (2) the measurement of spatial perception used in this experiment was not sensitive enough to resolve the prevailing differences. The former seems unlikely, as a number of studies have shown an effect of experience on motion perception (Bringoux et al. 2000; Clement et al. 2001; Woodard et al. 1987). Neither experienced nor naïve subjects identified the 0.5 Hz somatosensory task as particularly difficult, suggesting instead that a ceiling effect may have hindered potentially significant differences from being identified. This assertion is consistent with the uniform somatosensory bar responses described previously.

Sources of error:

Noise considerably reduced the quality of the right eye torsion signal, forcing us to rely entirely on left eye responses for our analysis. Unfortunately, torsion measurements of the left eye were also impaired by noise. This had a two-fold effect: it reduced the reliability of our torsion measurements, and hindered our ability to fit the data. Fortunately, the noise in the left eye torsion signal was relatively small, and limited to only a few measurement sessions. Consequently, our method of fitting (and thus characterizing) movement data for the left eye should not have been significantly affected.

Future Studies:

To improve statistical power, future investigations *must* examine a larger number of subjects. Relatively small sample sizes, such as the one described here, typically suffer from large within-group variability (Aron and Aron 1999). The two dark experienced subjects, for example, expressed uncorrected horizontal amplitude values during measurement 12 that ranged from 3.3°/s to 11.0°/s. This within-group variability made identifying between-group differences difficult, and may have adversely influenced the statistical tests conducted in this study.

A stricter criterion for recruiting subjects will help ensure "naïve" subjects truly have less experience with roll tilt motion than "experienced" subjects. Although recruiting naïve subjects in such a manner is logistically taxing, it may yield improved scientific benefits.

Lastly, the results that *were* significant apply only to monocular responses at 0.5 Hz. Compensatory eye movements are strongly tied to frequency (Paige and Tomko 1991); as such, future studies must confirm these results hold true at both higher and lower frequencies. Furthermore, future research should analyze the effect of motion experience on *binocular* eye responses. While it's unlikely that a left/right discrepancy exists with regards to horizontal or torsional VOR, the absence of such a disparity must be verified experimentally.

Conclusions:

Previous studies indicate that humans and monkeys respond to motion in a fundamentally different manner. Monkey VOR appears governed by an internal model

(Angelaki et al. 1999; Angelaki et al. 2001), Conversely, human VOR is presumably defined by frequency segregation (Merfeld et al. 2005a;b). Some researchers suspect these VOR differences are due to disparities in motion experience. The results found herein, however, *do not* support this hypothesis.

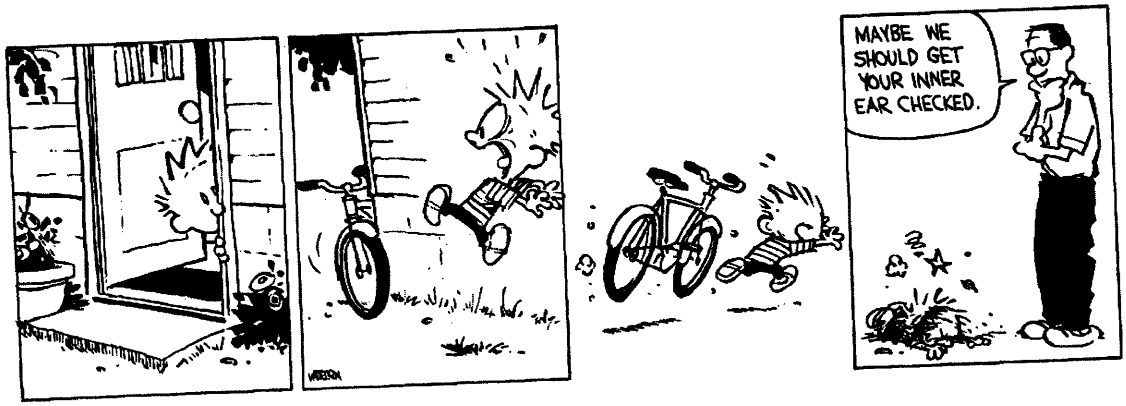
In this study, experience was operationalized two ways: (1) as between group differences between naïve and experienced subjects, and (2) as within subject differences between measurement 1 and measurement 12. The first method categorized subjects as either "experienced" or "naïve" based on their response to a motion questionnaire. Level of experience was positively correlated with UH eye amplitude, a result inconsistent with past research. Nevertheless, the second method of operationalizing experience *also* produced results that suggest experience does not improve horizontal VOR responses[18]. As subjects acquired experience with the motion paradigm, torsional amplitude declined, but horizontal amplitude remained stable. The measured decline in torsional amplitude indicates the motion experience was of sufficient duration and intensity to yield a significant VOR change. Therefore, the absence of a horizontal amplitude reduction suggests motion experience, as provided by the adaptation motion paradigm, does not improve the horizontal VOR response to motion.

If experience does not affect horizontal VOR, then measured horizontal VOR differences between monkeys and humans cannot be attributed to variations in motion experience. Of course, motion experience acquired during this investigation may not exemplify the background motion differences between primates and human. Subjects at the start and end of testing differ by three days of motion testing; primates and humans

[18] This method of operationalizing experience is probably more appropriate than the classification system, as it does not rely on subjective measures.

differ by years of testing. Thus, these findings may not generalize to human/primate VOR discrepancies. However, until further research proves otherwise, explicit testing of both species must be performed for the vestibulo-ocular reflex to be accurately and fully understood.

Appendix A—Clinical Testing Instructions



The following were the explicit instructions given to each subject prior to clinical testing. The instructions were taken verbatim from the Jenks Vestibular Diagnostic Laboratory, the location where clinical testing took place.

1. Dynamic Posture Testing: In this test, you will be asked to stand on a platform surrounded by a box providing a visual field. During testing the platform and/or the visual field may move. Your responses to these movements will be measured. Small disk electrodes may be placed on your legs to measure the electric signals emitted by your leg muscles as they contract. The forces you exert on the platform will be measured by pressure sensors in the platform. Your body movements will be directly measured by belts attached to your hips and shoulders.

2. Rotation Testing: In this test, identical to the clinical examination performed on patients, you will be seated in a rotation chair which will swivel from side to side like an office chair. Some parts of the test will be done in the dark. During other parts you will look at a moving or stationary lighted pattern. Small electrodes placed on the skin around your eyes will record your eye movements during the test

3. Electronystagmography (ENG) Testing: This test, which is identical to a widely used clinical examination, has several parts. Small electrodes placed on the skin around your eyes will record your eye movements during each of the following tests. We will test your ability to follow a moving target with your eyes and measure your eye movements while your head and body are placed in different positions. In the caloric test a small balloon will be placed in your ear canal and your eye movements will be monitored while warm or cool water is circulated through the balloon. In the Hallpike test you will be moved rapidly from a sitting position to one in which you are reclining with your head hanging slightly over the edge of the examination table, while your eye movements are measured.

Appendix B—Motion Questionnaire

1. Have you ever flown in an aircraft? Y N *(Please circle your answer)*

   1a. If yes, please estimate the number of times: *(Please circle your answer*)

In your lifetime?	0 times	1-10 times	11-50 times	≥ 51 times
In the last month?	0 times	1-2 times	3-5 times	≥ 5 times
In the last week?	0 times	1 times	2 times	≥ 2 times

   1b. If yes, what type(s) of aircraft? *(Please place a check next to all that apply)*

_____Commercial Jetliner
_____Helicopter
_____ Glider
_____GA Aircraft (Cessna, Piper, etc.)
_____Aerobatic Aircraft
_____Other

2. Have you ever been on board a boat? Y N *(Please circle your answer)*

   2a. If yes, please estimate the number of times: *(Please circle your answer*)

In your lifetime?	0 times	1-10 times	11-50 times	≥ 51 times
In the last month?	0 times	1-2 times	3-5 times	≥ 5 times
In the last week?	0 times	1 times	2 times	≥ 2 times

   2b. If yes, what type(s) of boats? *(Please place a check next to all that apply)*

_____Sailboat
_____Yacht
_____Cruise Ship
_____Motor Boat
_____Row Boat

3. Have you ever ridden an amusement park ride? Y N *(Please circle your answer)*

   3a If yes, please estimate the number of times: *(Please circle your answer*)

In your lifetime?	0 times	1-10 times	11-50 times	≥ 51 times
In the last month?	0 times	1-2 times	3-5 times	≥ 5 times
In the last week?	0 times	1 times	2 times	≥ 2 times

   3b. If yes, what type(s) of rides? *(Please place a check next to all that apply)*

_____Roller Coaster
_____Merry Go Round
_____Motion Simulator
_____Other Rotating Rides

4. Have you ever ridden an active, motion-based simulators? Y N *(Please circle your answer)*

   4a. If yes, please estimate the number of times: *(Please circle your answer*)

In your lifetime?	0 times	1-10 times	11-50 times	≥ 51 times
In the last month?	0 times	1-2 times	3-5 times	≥ 5 times
In the last week?	0 times	1 times	2 times	≥ 2 times

5. Have you ever ridden in a high performance sports car? Y N *(Please circle your answer)*

   5a. If yes, please estimate the number of times: *(Please circle your answer*)

In your lifetime?	0 times	1-10 times	11-50 times	≥ 51 times
In the last month?	0 times	1-2 times	3-5 times	≥ 5 times

In the last week?	0 times	1 times	2 times	≥ 2 times

6. In the space below, please describe any other types of atypical motion you have experienced, especially in the last year. (Enter "None", if no additional motion experience to report.)

Appendix C

Four programs, created in Matlab, were used to analyze the data. The first program, "VNAS", gives the user the ability to manually delete saccades. Because of its length and number of subroutines, as well as its successful use in other experiments (it has been used by this lab for over 10 years), it will not be described in this thesis.
The second program, "lvb_daq_vog" (p. 70) allows the user to graphically display the three primary variables of interest—tilt device position, somatosensory bar position, and desaccaded eye velocity—and generates the necessary values used to temporally sync the three variables. In addition, "lvb_daq_vog" calculates the somatosensory bar offset based on the 0.1 second time period prior to button push. The third program ("SS Bar Fit", p. 78) fits either the somatosensory bar or slow phase velocity to a sinusoid as described in the results section, and from this calculates gain and relative phase. The fourth program ("Standard Error", p. 84) employed in this study calculates the standard error for gain and phase. (Several other programs were employed in automating the analysis, but are not described here, as they are basic derivatives of the four programs listed.) The last two programs in this study were used to calculate day and group amplitude and phase, for eye movements ("Eye Movement Response Plots", p. 87) and somatosensory bar ("Somatosensory Bar Plots", p. 140), respectively. Sections in the program denoted by a "%" indicate a programming comment.

lvb_daq_vog

The first portion of "lvb_daq_vog" program is based on a program originally written by Lionel Zupan. It converts lvb data (tilt motion and somatosensory bar) into a format accessible by Matlab. The primary goal of this program is to identify the time points in the VOG data file where the tilt device motion first reaches steady state. By calculating the trigger and the motion start in the .lvb file, and knowing the trigger start in the vog file, one can identify the motion start in the vog file. Steady state begins 6 seconds after motion start for 0.5 Hz data, and 4 seconds after motion start for 1.0 Hz data.

Additionally, the program allows the user to either manually or automatically calculate the somatosensory bar bias. The time points where the right button is pressed (indiciating the subject had set the bar to his or her perception of earth horizontal) were used to identify the actual angle of the somatosensory bar. From these values, an average bias was calculated

```
clear all; %ckears all variables

%Selects the tilt motion files (.lvb files) and then opens and reads the DAQparams files

[filename,pathname]=uigetfile(,'Choose an LVB data file');
l=length(filename);filepar=[filename(1:(l-16)),'_DAQparams.txt'];
fid=fopen([pathname,filepar]);
filebit=[filename(1:(l-16)),'_DigitalInData.bit'];

nchannel=0;
category=0;
while 1

        tline = fgetl(fid);
        if (category==1)& strncmp(tline, '[', 1), category =2;end;
        if strncmp(tline, '[Acquisition]', 13) category=1; end;
        indq=find(tline=="");
        if length(indq)>1,
        tline(indq(1))="";tline(indq(length(indq)))="";
end;

        if (category==1) & strncmp(tline, 'Channel', 7)

           nchannel=nchannel+1;
           ChannelN=tline(1:8);
           indq2=find(tline=="");
           string = tline((indq2(1)+1):(indq2(2)-1));
           indc = find(string==',');indc=[indc,length(string)+1];
           dindc=diff(indc);
           eval([ChannelN,'_Nr=string(1:(indc(1)-1));']);
           if dindc(1)>1,eval([ChannelN,'_Lbl=string((indc(1)+1):(indc(2)-1));']),
```

```
            else eval([ChannelN,'_Lbl=[];']);end;

            if dindc(2)>1,eval([ChannelN,'_Sca=string((indc(2)+1):(indc(3)-1));']),
            else eval([ChannelN,'_Sca="1";']);end;

            if dindc(3)>1,eval([ChannelN,'_Off=string((indc(3)+1):(indc(4)-1));']),
            else eval([ChannelN,'_Off="0";']);end;

            if dindc(4)>1,eval([ChannelN,'_Uni=string((indc(4)+1):length(string));']),
            else eval([ChannelN,'_Uni=[];']);end;

        end;

        if (category==1) & strncmp(tline, 'SampleFrequency', 15),eval([tline,';']),end;
%Determine sample frequency

        if (category==1) & strncmp(tline, 'ScaleFactor', 11),eval([tline,';']),end;

 %Determine scale factor that converts A/D counts to volts

        if ~ischar(tline), break, end
    end

fclose(fid);

% Open and read the *.bit file

fid1=fopen([pathname,filebit],'r','ieee-be');
[data_bit,count]=fread(fid1,'uint8');
fclose(fid1);

for i=0:7

  eval(['DigitalIn',int2str(i),'=bitget(data_bit,',int2str(i+1),');'])
  subplot(8,1,i+1), eval(['plot(DigitalIn',int2str(i),')']); ylim([0 2]);

end

% Open and read *.lvb (tilt motion) file

fid2=fopen([pathname,filename],'r','ieee-be');
[data_open,count]=fread(fid2,'short');
fclose(fid2);

%Convert data
```

```
for i=1:nchannel  % This for-loop converts data to matlab array containing A-to-D counts
  eval(['data_b(:,',int2str(i),')=data_open(',int2str(i),':nchannel:count);']);

end

% In figure 1, plot data in physical units as a function of time (in seconds)

figure(1),clf;
[a,b]=size(data_b);
t = (0:1:(a-1))/SampleFrequency; %Calculate time vector
I = ones([a,1]);

for i=1:nchannel
   eval(['Channel_Nr=Channel',char(64+i),'_Nr;']);
   eval(['Channel_Lbl=Channel',char(64+i),'_Lbl;']);
   eval(['Channel_Sca=eval(Channel',char(64+i),'_Sca);']);
   eval(['Channel_Off=eval(Channel',char(64+i),'_Off);']);
   eval(['Channel_Uni=Channel',char(64+i),'_Uni;']);

   subplot(nchannel,1,i), plot(t,((data_b(:,i)/ScaleFactor)-Channel_Off*I)*Channel_Sca);
   ylabel(Channel_Uni);
   title(['Channel',char(64+i),': ',Channel_Lbl,' [',Channel_Nr,']']);
   hold on

   subplot(3,1,1), plot(t,(DigitalIn5*20),'r',t,(DigitalIn5*-20),'r')

   subplot(3,1,3), plot(t,(DigitalIn7*100),'k',t,(DigitalIn7*-
100),'k',t,(DigitalIn5*100),'r',t,(DigitalIn5*-100),'r')

end

%Selects the somatosensory offset based on right button pushes

offset=input('Do you want to manually select the offsets [Y/N]: ','s');

if offset=='N' %Automatically identifies when the button was pushed, and calculates the
%SS bar offset from these time points

[s7,s7a]=size(DigitalIn7);
[val,ind1_b]=min(abs(DigitalIn7-0));
ind1_b;
[val,ind1f_b]=min(abs(DigitalIn7(ind1_b:s7)-1));
ind1f_b=ind1f_b+ind1_b-1;
[val,ind2_b]=min(abs(DigitalIn7(ind1f_b:s7)-0));
ind2_b=ind2_b+ind1f_b-1;
```

```
[val,ind2f_b]=min(abs(DigitalIn7(ind2_b:s7)-1));
ind2f_b=ind2f_b+ind2_b-1;
[val,ind3_b]=min(abs(DigitalIn7(ind2f_b:s7)-0));
ind3_b=ind3_b+ind2f_b-1;
[val,ind3f_b]=min(abs(DigitalIn7(ind3_b:s7)-1));
ind3f_b=ind3f_b+ind3_b-1;
[val,ind4_b]=min(abs(DigitalIn7(ind3f_b:s7)-0));
ind4_b=ind4_b+ind3f_b-1;
[val,ind4f_b]=min(abs(DigitalIn7(ind4_b:s7)-1));
ind4f_b=ind4f_b+ind4_b-1;
[val,ind5_b]=min(abs(DigitalIn7(ind4f_b:s7)-0));
ind5_b=ind5_b+ind4f_b-1;
[val,ind5f_b]=min(abs(DigitalIn7(ind5_b:s7)-1));
ind5f_b=ind5f_b+ind5_b-1;
[val,ind6_b]=min(abs(DigitalIn7(ind5f_b:s7)-0));
ind6_b=ind6_b+ind5f_b-1;
[val,ind6f_b]=min(abs(DigitalIn7(ind6_b:s7)-1));
ind6f_b=ind6f_b+ind6_b-1;

q=((data_b(:,3)/ScaleFactor)-Channel_Off*I)*Channel_Sca;

SSaverage=((mean(q((ind1_b-12):ind1_b)))+(mean(q((ind2_b-
12):ind2_b)))+(mean(q((ind3_b-12):ind3_b)))+(mean(q((ind4_b-
12):ind4_b)))+(mean(q((ind5_b-12):ind5_b)))+(mean(q((ind6_b-12):ind6_b))))/6;

else

'Select the starting points for the 6 offset button pushes' %Allows the user to manually
%select the timepoints used to calculate the offset

pause
[x1]=ginput(1);
x1=round(SampleFrequency*x1);
q=((data_b(:,3)/ScaleFactor)-Channel_Off*I)*Channel_Sca;
SS1=mean(q((x1(1,1)-12):x1(1,1)));

pause
[x2]=ginput(1);
x2=round(SampleFrequency*x2);
q=((data_b(:,3)/ScaleFactor)-Channel_Off*I)*Channel_Sca;
SS2=mean(q((x2(1,1)-12):x2(1,1)));

pause
[x3]=ginput(1);
x3=round(SampleFrequency*x3);
q=((data_b(:,3)/ScaleFactor)-Channel_Off*I)*Channel_Sca;
```

```
SS3=mean(q((x3(1,1)-12):x3(1,1)));

pause
[x4]=ginput(1);
x4=round(SampleFrequency*x4);
q=((data_b(:,3)/ScaleFactor)-Channel_Off*I)*Channel_Sca;
SS4=mean(q((x4(1,1)-12):x4(1,1)));

pause
[x5]=ginput(1);
x5=round(SampleFrequency*x5);
q=((data_b(:,3)/ScaleFactor)-Channel_Off*I)*Channel_Sca;4
SS5=mean(q((x5(1,1)-12):x5(1,1)));

pause
[x6]=ginput(1);
x6=round(SampleFrequency*x6);
q=((data_b(:,3)/ScaleFactor)-Channel_Off*I)*Channel_Sca;
SS6=mean(q((x6(1,1)-12):x6(1,1)));

SSaverage=(SS1+SS2+SS3+SS4+SS5+SS6)/6;

end

%Displays the trigger start in points, There were four triggers in each measurement
%session that helped define the starting points for each of the four motions

trigger=input('Do you want to manually select the triggers [Y/N]: ','s');

if trigger=='N'

[s5,s7a]=size(DigitalIn5);
[val,ind1_t]=min(abs(DigitalIn5-1));
ind1_t;

[val,ind2_t]=min(abs(DigitalIn5(ind1_t+125:s5)-1));
ind2_t=ind2_t+ind1_t+125-1;
ind2_t;

[val,ind3_t]=min(abs(DigitalIn5(ind2_t+125:s5)-1));
ind3_t=ind3_t+ind2_t+125-1;
ind3_t;

[val,ind4_t]=min(abs(DigitalIn5(ind3_t+125:s5)-1));
ind4_t=ind4_t+ind3_t+125-1;
ind4_t;
```

```
else

'Select the starting points for the 4 triggers'

[s5,s7a]=size(DigitalIn5);

[t1]=ginput(1);
[t2]=ginput(1);
[t3]=ginput(1);
[t4]=ginput(1);

t1=round(SampleFrequency*t1);

[val,ind1_t]=min(abs(DigitalIn5(t1:t1+500)-1));
ind1_t=ind1_t(1)+t1(1)-1;

t2=round(SampleFrequency*t2);
[val,ind2_t]=min(abs(DigitalIn5(t2:t2+500)-1));
ind2_t=ind2_t(1)+t2(1)-1;

t3=round(SampleFrequency*t3);
[val,ind3_t]=min(abs(DigitalIn5(t3:t3+500)-1));
ind3_t=ind3_t(1)+t3(1)-1;

t4=round(SampleFrequency*t4);
[val,ind4_t]=min(abs(DigitalIn5(t4:t4+500)-1));
ind4_t=ind4_t(1)+t4(1)-1;

end

%DAQ Motion Finder, automatically identifies the start of motion for each of the four
movements

r=(((data_b(:,1)/ScaleFactor)-Channel_Off*I)*Channel_Sca);
steady=mean(r(ind1_t-155:ind1_t-145));
for i=100:130;

   if r(ind1_t-i)-steady>=0.02
      ind1_m=ind1_t-i;
   end

end

steady=mean(r(ind2_t-155:ind2_t-145));
for i=100:130;
```

```
  if r(ind2_t-i)-steady>=0.02
    ind2_m=ind2_t-i;
  end

end

steady=mean(r(ind3_t-155:ind3_t-145));
for i=100:130;
  if r(ind3_t-i)-steady>=0.02
    ind3_m=ind3_t-i;
  end

end

steady=mean(r(ind4_t-155:ind4_t-145));
for i=100:130;
  if r(ind4_t-i)-steady>=0.02
    ind4_m=ind4_t-i;
  end
end

%Selects the VOG data file, then identifies the temporal location where the trigger
occured
[fnameVOG,pnameVOG] = uigetfile('*.txt','SELECT VOG FILE (_cor.txt)');
[header,M]=hdrload([pnameVOG fnameVOG]);

t=M(:,1);
Analog1=M(:,28);

for i=1:size(t)-1
  if Analog1(i+1)-Analog1(i)>=4.7;
    break
  end
end
ind1_v=i+1;

for i=ind1_v+61:size(t)-1
  if Analog1(i+1)-Analog1(i)>=4.7;
    break
  end
end
ind2_v=i+1;

for i=ind2_v+61:size(t)-1
  if Analog1(i+1)-Analog1(i)>=4.7;
    break
```

```
  end
end
ind3_v=i+1;

for i=ind3_v+61:size(t)-1
   if Analog1(i+1)-Analog1(i)>=4.7;
      break
  end
end
ind4_v=i+1;

%Verifies that the motion index points are correct
if (r(ind1_m)-r(ind1_m-1)>=0.0245) & (r(ind2_m)-r(ind2_m-1)>=0.0245) & (r(ind3_m)-
r(ind3_m-1)>=0.0245) & (r(ind4_m)-r(ind4_m-1)>=0.0245)

   fprintf('Motion OK\n')

else

   fprintf('Motion NOT OK\n')

end

%Calculates the steady state motion start time
p5aSS=((ind1_v/60)-((ind1_t-ind1_m)/120))+6;
op0aSS=((ind2_v/60)-((ind2_t-ind2_m)/120))+4;
p5bSS=((ind3_v/60)-((ind3_t-ind3_m)/120))+6;
op0bSS=((ind4_v/60)-((ind4_t-ind4_m)/120))+4;

fprintf('\nOffset:  %.4f degrees\n0.5 Hz DAQ Trigger:  %.0f  DAQ Motion:  %.0f  VOG
Trigger:  %.0f  VOG Steady State Start:  %.4fs\n1.0 Hz DAQ Trigger:  %.0f  DAQ
Motion:  %.0f  VOG Trigger:  %.0f  VOG Steady State Start:  %.4fs\n0.5 Hz DAQ
Trigger:  %.0f  DAQ Motion:  %.0f  VOG Trigger:  %.0f  VOG Steady State Start:
%.4fs\n1.0 Hz DAQ Trigger:  %.0f  DAQ Motion:  %.0f  VOG Trigger:  %.0f  VOG
Steady State Start:
%.4fs\n',SSaverage,ind1_t,ind1_m,ind1_v,p5aSS,ind2_t,ind2_m,ind2_v,op0aSS,ind3_t,in
d3_m,ind3_v,p5bSS,ind4_t,ind4_m,ind4_v,op0bSS)
```

SS Bar Fit

The SS Bar Fit program fit the somatosensory bar and device position to a sinusoid, and from these fits, calculated gain and phase. If a portion of the fit did not accurately characterize the data, the user could remove this cycle from the gain and phase calculation.

A variation of this program was used to fit the slow phase velocity for horizontal and torsional eye movements. This program also had the additional task of converting the tilt device position signal, as recorded by in the .lvb file, into a measure of tilt velocity.

```
% Manually read the labview files
    cd /Volumes/Untitled/Motion_Experience/RCT74_D2/RCT74_D2_Motion/
    %cd H:/Motion_Experience/RPO_D1/SZ80_D1_Motion/
    [filename,pathname]=uigetfile({'*.lvb'},'Choose an LVB data file');
     %questions
     offset=input('What is the somatosensory bar offset value?');
     t_daq_start=input('What is the DAQ motion start time (pts)?');


  l=length(filename);filepar=[filename(1:(l-16)),'_DAQparams.txt'];
  fid=fopen([pathname,filepar]);
  if (fid<0)
     error(['error opening file ',[pathname,filepar]]);
  end
  filebit=[filename(1:(l-16)),'_DigitalInData.bit'];

  nchannel=0;
  category=0;
  while 1

         tline = fgetl(fid);
         if (category==1)& strncmp(tline, '[', 1), category =2;end;
         if strncmp(tline, '[Acquisition]', 13) category=1; end;
         indq=find(tline=='"');
         if length(indq)>1,
            tline(indq(1))='''';tline(indq(length(indq)))=''''; %Replace quotations (") with
apostrophes (') for MATLAB
            end;
         if (category==1) & strncmp(tline, 'Channel', 7)
            nchannel=nchannel+1;
            ChannelN=tline(1:8);
            indq2=find(tline=='''');
            string = tline((indq2(1)+1):(indq2(2)-1));
            indc = find(string==',');indc=[indc,length(string)+1];
            dindc=diff(indc);
```

```
            eval([ChannelN,'_Nr=string(1:(indc(1)-1));']);
            if dindc(1)>1,eval([ChannelN,'_Lbl=string((indc(1)+1):(indc(2)-1));']),
            else eval([ChannelN,'_Lbl=[];']);end;
            if dindc(2)>1,eval([ChannelN,'_Sca=string((indc(2)+1):(indc(3)-1));']),
            else eval([ChannelN,'_Sca="1";']);end;
            if dindc(3)>1,eval([ChannelN,'_Off=string((indc(3)+1):(indc(4)-1));']),
            else eval([ChannelN,'_Off="0";']);end;
            if dindc(4)>1,eval([ChannelN,'_Uni=string((indc(4)+1):length(string));']),
            else eval([ChannelN,'_Uni=[];']);end;
          end;
          if (category==1) & strncmp(tline, 'SampleFrequency', 15),eval([tline,';']),end;
%Determine sample frequency
          if (category==1) & strncmp(tline, 'ScaleFactor', 11),eval([tline,';']),end;
%Determine scale factor that converts A/D counts to volts
          if ~ischar(tline), break, end
        end
    fclose(fid);

    clear data_bit data_open data_b SSbar var
    % Open and read the *.bit file
    fid1=fopen([pathname,filebit],'r','ieee-be');
    [data_bit,count]=fread(fid1,'uint8');
    fclose(fid1);

    for i=0:7
      eval(['DigitalIn',int2str(i),'=bitget(data_bit,',int2str(i+1),');'])
    end

    % Open and read *.lvb file
    fid2=fopen([pathname,filename],'r','ieee-be');
    [data_open,count]=fread(fid2,'short');
    fclose(fid2);

    %Convert data
    for i=1:nchannel  % This for-loop converts data to matlab array containing A-to-D
counts
      eval(['data_b(:,',int2str(i),')=data_open(',int2str(i),':nchannel:count);']);
    end

    %Calculate time vector
    [a,b]=size(data_b);
    t = (0:1:(a-1))/SampleFrequency;

    I = ones([a,1]);
    for i=1:nchannel
      eval(['Channel_Nr=Channel',char(64+i),'_Nr;']);
```

```
    eval(['Channel_Lbl=Channel',char(64+i),'_Lbl;']);
    eval(['Channel_Sca=eval(Channel',char(64+i),'_Sca);']);
    eval(['Channel_Off=eval(Channel',char(64+i),'_Off);']);
    eval(['Channel_Uni=Channel',char(64+i),'_Uni;']);
  end

  %load /Volumes/Untitled/Motion_Experience/RPO81_D1/L_Test1_0.5 Hz.mat
  cd /Volumes/Untitled/Analysis_Programs/endallbeall
  %cd H:/Analysis_Programs/endallbeall

  %t_VOG_start=input('What is the VOG state motion start time (pts)?');
  %shift=(t_daq_start/120)-(t_VOG_start/60);

  SSBar=(((data_b(:,3)/3125)-1.3248*I)*76.6834)-offset;
  Device_Position=((data_b(:,1)/3125)-0*I)*3.018;

  var(:,1)=SSBar(:,1);
  var(:,2)=Device_Position(:,1);

  for j = 1:2   % j: index of varibles
    for i = 1:10          % i: index of cycles
      cycle_begin = (t_daq_start/120) +6  + (i-1)/0.5;
      cycle_end = (t_daq_start/120) +6 + i/0.5;
      t_index(i,:) = find( t >= cycle_begin & t < cycle_end);

      a = var(t_index(i,:),j);
      samp = (t(t_index(i,:))*0.5)';
      e = ones(size(samp));

      [y_complex(i,j),bias(i,j),amp,phase] = slowdft3(a,samp,e,1);
      fit_curve(:,i,j) = amp*cos(2*pi*samp+phase/180*pi) + bias(i,j);
        %slowdft3 is a subroutine that computes a discrete fourier transform of the data
    end

  end

    figure(3)
    subplot(2,1,1)
    for i = 1:10          % i: index of cycles
        plot(t(t_index(i,:)), var(t_index(i,:),1),'c');hold on;                % SSBar
Position
    end
    subplot(2,1,2)
    for i = 1:10          % i: index of cycles
        plot(t(t_index(i,:)), var(t_index(i,:),2),'c');hold on;                % Device
Position
```

```
    end

    pause

    figure(3)
    subplot(2,1,1)
    for i = 1:10        % i: index of cycles
        plot(t(t_index(i,:)), fit_curve(:,i,1),'r');hold on;
    end
    subplot(2,1,2)
    for i = 1:10        % i: index of cycles
        plot(t(t_index(i,:)), fit_curve(:,i,2),'r');hold on;
    end

  % pick out bad cycles
  num_cycl=10;
  delet_mark = zeros(num_cycl,1);
  mouse_button = 1;
  while (mouse_button ~= 2)
    [x, y, mouse_button] = ginput(1);
    i = ceil((x - (t_daq_start/120)+ 6) * 0.5);
    if (mouse_button == 1)
      delet_mark(i) = 1;
      subplot(2,1,1); hold on, plot(t(t_index(i,:)), fit_curve(:,i,1),'b');
      subplot(2,1,2); hold on, plot(t(t_index(i,:)), fit_curve(:,i,2),'b');
    end
    if (mouse_button == 3)
      delet_mark(i) = 0;
      subplot(2,1,1); hold on; plot(t(t_index(i,:)), fit_curve(:,i,1),'r');
      subplot(2,1,2); hold on, plot(t(t_index(i,:)), fit_curve(:,i,2),'r');
    end
  end

  % transfer good cycle values to new variables,
  %clear y_complex_good bias_good;
  j = 1;
  for i = 1:10
    if (delet_mark(i)==0)              % 1: deleted, 0: not deleted
      y_complex_good(j,:) = y_complex(i,:);
      bias_good(j,:) = bias(i,:);
      j = j + 1;
    end
  end


%clear y_complex_good_ave;
```

```
for j = 1:2
  y_complex_good_ave(j) = mean(y_complex_good(:,j));
  bias_good_ave(j) = mean(bias_good(:,j));
end

  SS_gain=abs(y_complex_good_ave(1)/y_complex_good_ave(2));
  SS_amp=abs(y_complex_good_ave(1));
  SS_phase=180/pi*angle(y_complex_good_ave(1)/y_complex_good_ave(2));
  if SS_phase<0, SS_phase= SS_phase+360; end;
  %SS_phase=SS_phase-180;

  fprintf('The SSBar amplitude is %7.4f. \n',SS_amp)
  fprintf('The SSbar gain is %7.4f. \n',SS_gain)
  fprintf('The SSBar phase is %6.2f degrees.\n',SS_phase)
  fprintf('The SSBar bias is %6.2f deg/s\n\n',bias_good_ave(1))
  cd /Volumes/Untitled/Motion_Experience/MM80_D3/MM80_D3_SSBar_Variables
  %cd H:/Motion_Experience/MM80_D1/MM80_D1_SSBar_Variables
  fprintf('%7.4f\t%7.4f\t%6.2f\t%6.2f\n',SS_amp,SS_gain,SS_phase,bias_good_ave(1))
```

Slowdft3—subroutine for all fit programs

Slowdft3 is the subroutine used by each variation of the fit program. See comments for details.

```
function [y,bias,amp,phase]=slowdft3(a,samp,e,units)

% SLOWDFT3 - Computes a discrete Fourier transform by a least squared error fit of
%cosine and sine waves to the amplitude series that are sampled at specific sample point
%coordinates. Points from the time series can be excluded from the fit procedure by
%flagging them with a zero value in the vector "e". The input data series assumes that
%you are giving one cycle of the series, and the amplitude and phase data of the sine fit
%are at the fundamental frequency of that single cycle.

%	a = amplitude series (a column vector)
%	samp = sample point corrdinates (a column vector)
%			(e.g. radians or degrees)
%	e = vector the same size as "a" with elements:
%				1 = include point in fit
%				0 = exclude point from fit
%	units = the value of one full cycle of sample points
%			(e.g. 2*pi or 360)
%	bias = DC offset of sine fit
%	amp = amplitude of the sine fit
%	phase = phase of sine fit - phase is relative to a cosine
%		waveform with negative indicating a phase lag
```

```
%x=detrend(x);                    % take out any linear trend
ag=a(e==1);                       % good points only
c=cos(2*pi*samp/units);
s=sin(2*pi*samp/units);
cg=c(e==1);                       % good points only
sg=s(e==1);
csum=sum(cg);
ssum=sum(sg);
cssum=sum(cg.*sg);
A=[max(size(ag)) csum ssum
      csum sum(cg.*cg) cssum
      ssum cssum sum(sg.*sg)];
B=[sum(ag) sum(ag.*cg) sum(ag.*sg)];
X=A\B';                           % least squares fit of dc, cosine, sine
y=X(2)-X(3)*i;                    % cosine & sine terms in complex number
bias=X(1);                        % DC term
amp=abs(y);                       % amplitude of sine fit
phase=180/pi*atan2(imag(y),real(y));     % phase -pi <= phase <= pi
```

Standard Error

To calculate standard error, one program and one routine were used in this study. Real and imaginary values, which represented gain and phase for a particular category on a particular testing session, were entered, and phase and amplitude standard error were computed. The standard error for gain was proportionate to the standard error of the amplitude.

```
%Subject enters real and imaginary values below

A_real = [-19.8738 -7.7469 0];
A_imag = [-15.3710 - 1.1592 0];

A_complex = complex(A_real, A_imag)
A_mean = mean(A_complex, 2);
A_amp = abs(A_mean)
A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));

display = 1;   % 1:display, 0:no
e = 1;         % e = [ ]: standard error, e = 1: standard deviation
A_std = f_covellips_wg(A_complex,e,display);
%f_covellips_wg is a subroutine called by this program.  It is shown below

Amp_STDorSE = A_std(1)
Ph_STDorSE  = A_std(2) * 180/pi

function output = f_covellips(xorg,e,plot_option)
```

f_covellips_wg—subroutine for standard error program

```
% CALUCLATE COVARIANCE ELLIPSE AND RELATED PARAMETERS
% INPUT:     x shoud have complex form data components
%                          e = []  indicates the output as standard error
%                          e = 1   indicates the output as standard deviation
% OUTPUT:  the first is the std (or se) on magnitude
%          the second is std (or se) on phase

% how to use: (from Lionel)
% x=abs(randn(100,2));  % create a positive-only random 2D sample (n =100)
% z=x(:,1)+i*x(:,2);    % convert it in imaginary format
% o=f_covellips(z,[],1)  % This gives standard error
% o=f_covellips(z,1,1)   % This gives std

%K = 1;         % probability = 1-1/sqrt(e) = 0.39
%K = 4.6;     % probability = 90%
```

```
K = 6;    % probability = 1-e^(-3)= 0.95
%K = 9.2;      % probability = 1-e^(-4.6)= 0.99

% Preparing data
x = real(xorg);
y = imag(xorg);

center = [mean(x),mean(y)];
th = [-pi:0.0005*pi:pi];

% Calculate covariance and covariance ellipse radius, r
a = cov(x,y);
rho = a(2)/sqrt(a(1)*a(4));
A = (cos(th).^2/a(1) - 2*rho^2/a(2)*sin(th).*cos(th) + sin(th).^2/a(4));
B = K*(1-rho^2);
r = sqrt(B./A);

% principal axis radius
alpha = atan(2*a(2)/(a(1)-a(4)))/2;
p1 = sqrt(a(1)*a(4)*K*(1-rho^2)/(a(4)*(cos(alpha))^2-
2*a(2)*sin(alpha)*cos(alpha)+a(1)*(sin(alpha))^2));
p2 = sqrt(a(1)*a(4)*K*(1-
rho^2)/(a(4)*(sin(alpha))^2+2*a(2)*sin(alpha)*cos(alpha)+a(1)*(cos(alpha))^2));

% STD of the magnitude of complex data
phi = atan2(center(2),center(1));
R = sqrt(center(1).^2+center(2).^2);
A1 = (cos(phi).^2/a(1) - 2*rho^2/a(2)*sin(phi).*cos(phi) + sin(phi).^2/a(4));
Rstd = sqrt(B/A1);

% STD of the angle of complex data: Definied by tangent to the ellipse
A2 = 2*center(1)/a(1)-2*rho^2*center(2)/a(2);
B2 = 2*center(2)/a(4)-2*rho^2*center(1)/a(2);
C2 = center(1)^2/a(1)+center(2)^2/a(4)-2*rho^2/a(2)*center(1)*center(2)-(1-rho^2);
A3 = B2^2-4*C2/a(4);
B3 = 2*A2*B2+8*rho^2*C2/a(2);
C3 = A2^2-4*C2/a(1);
beta = (atan(roots([A3,B3,C3])))';

% STD of the angle of complex data: Defined by orthogonal to the radius
m = -1/tan(phi);
b = center(1)/tan(phi) + center(2);
A3 = 1/a(1) - 2*rho^2*m/a(2) + m^2/a(4);
B3 = -2*center(1)/a(1) - 2*rho^2/a(2)*(b-center(2)-m*center(1)) + 2*m*(b-
center(2))/a(4);
```

```
C3 = center(1)^2/a(1) + 2*rho^2*center(1)/a(2)*(b-center(2)) + (b-center(2))^2/a(4) - (1-
rho^2);
sol = roots([A3,B3,C3])';
beta = [atan2(m*sol(1)+b,sol(1)),atan2(m*sol(2)+b,sol(2))];

% Plotting
if plot_option
   figure;
   clf
   plot(x,y,'.')
   %for i = 1:length(x)
   %   text(x(i),y(i),num2str(i))
   %   hold on
   %end
   hold on,
   plot(r.*cos(th)+center(1),r.*sin(th)+center(2)),axis equal
   grid on
   plot(center(1),center(2),'ro')
   R2 = R*2;
   line([0,(R2)*cos(phi)],[0,(R2)*sin(phi)])
   line([0,(R2)*cos(beta(1))],[0,(R2)*sin(beta(1))])
   line([0,(R2)*cos(beta(1)+pi)],[0,(R2)*sin(beta(1)+pi)])
   line([0,(R2)*cos(beta(2))],[0,(R2)*sin(beta(2))])
   line([0,(R2)*cos(beta(2)+pi)],[0,(R2)*sin(beta(2)+pi)])
   %    title(['p_1 = ',num2str(p1), ', p_2 = ',num2str(p2),', R_{mean} = ',num2str(R),',
R_{\sigma} = ',num2str(Rstd),', \phi = ',num2str(phi),', \beta = ',num2str(beta)])
   title(['R_{\sigma} = ',num2str(Rstd),', \phi = ',num2str(phi),', \beta = ',num2str(beta),',
\Delta\beta = ',num2str(beta-phi)])
   % xorg
   % pause
end

beta = min(abs(beta - phi));

if isempty(e)
   output = [Rstd beta]/sqrt(length(x)); % standard error
else
   output = [Rstd beta];    % standard deviation
end
```

Eye Movement Response Plots

```
clear all

load /Volumes/Untitled/Motion_Experience/eye10.mat

cd /Volumes/Untitled/Analysis_Programs/

cf=62.46;

q=input('Show graphs? [Y/N]:','s');

%RPO=subject 1, AAR=subject 2, DMM=subject 3, FK=subject 4, MM=SUbject 5,
%JK=subject 6 SZ=subject 7, RCT=subject 8

%y_complex_good_all(:,1)=Uncorrected Horizontal Velcoity
%y_complex_good_all(:,2)=Uncorrected Torsional Velcoity
%y_complex_good_all(:,3)=Corrected Horizontal Velcoity
%y_complex_good_all(:,4)=Corrected Torsional Velcoity
%y_complex_good_all(:,5)=Device velocity

%Subject 7 discluded for tests 1 through 4 because of thermoplastic mask
%occlusion
%Subject 6 Movement 3 removed because not recorded
%Subject 2 Test 12 removed because not recorded




%Calculates the non-nan length of each MOVEMENT, e.g. calculates the number
%of used cycles in the MOVEMENT
for subject=1:8
  for days=1:3
    for tests=1:4
      for movements=1:2
        new_length(subject,days,tests,movements)=10-
length(find(isnan(y_complex_good_all(subject,days,tests,movements,:,1))));
      end
    end
  end
end

%Calculates the average complex value for each MOVEMENT
for subject=1:8
  for days=1:3
```

```
        for tests=1:4
          for movements=1:2

            average_uh(subject,((days-1)*8)+((tests-
1)*2)+movements)=nansum(squeeze(y_complex_good_all(subject,days,tests,movements
,:,1)))/new_length(subject,days,tests,movements);
            average_ut(subject,((days-1)*8)+((tests-
1)*2)+movements)=nansum(squeeze(y_complex_good_all(subject,days,tests,movements
,:,2)))/new_length(subject,days,tests,movements);
            average_ch(subject,((days-1)*8)+((tests-
1)*2)+movements)=nansum(squeeze(y_complex_good_all(subject,days,tests,movements
,:,3)))/new_length(subject,days,tests,movements);
            average_ct(subject,((days-1)*8)+((tests-
1)*2)+movements)=nansum(squeeze(y_complex_good_all(subject,days,tests,movements
,:,4)))/new_length(subject,days,tests,movements);
            average_device(subject,((days-1)*8)+((tests-
1)*2)+movements)=nansum(squeeze(y_complex_good_all(subject,days,tests,movements
,:,5)))/new_length(subject,days,tests,movements);

          end
        end
    end
end



%Andrew,Jen,Suzanne
for subject=1:8
    for movements=1:2:23
        if subject==6 & movements==3

complex_uh(subject,round(movements/2))=(average_uh(subject,movements)/average_de
vice(subject,movements));

complex_ut(subject,round(movements/2))=(average_ut(subject,movements)/average_dev
ice(subject,movements));

complex_ch(subject,round(movements/2))=(average_ch(subject,movements)/average_de
vice(subject,movements));

complex_ct(subject,round(movements/2))=(average_ct(subject,movements)/average_devi
ce(subject,movements));
        else

complex_uh(subject,round(movements/2))=((average_uh(subject,movements)/average_d
```

```
evice(subject,movements))+(average_uh(subject,movements+1)/average_device(subject,
movements+1)))/2;

complex_ut(subject,round(movements/2))=((average_ut(subject,movements)/average_de
vice(subject,movements))+(average_ut(subject,movements+1)/average_device(subject,m
ovements+1)))/2;

complex_ch(subject,round(movements/2))=((average_ch(subject,movements)/average_de
vice(subject,movements))+(average_ch(subject,movements+1)/average_device(subject,m
ovements+1)))/2;

complex_ct(subject,round(movements/2))=((average_ct(subject,movements)/average_de
vice(subject,movements))+(average_ct(subject,movements+1)/average_device(subject,m
ovements+1)))/2;
      end
   end
end

for tests=1:12
   if tests==12

gain_uh_light(tests)=abs((complex_uh(4,tests)+complex_uh(5,tests)+complex_uh(6,tests
))/3);

gain_ut_light(tests)=abs((complex_ut(4,tests)+complex_ut(5,tests)+complex_ut(6,tests))/
3);

gain_ch_light(tests)=abs((complex_ch(4,tests)+complex_ch(5,tests)+complex_ch(6,tests)
)/3);

gain_ct_light(tests)=abs((complex_ct(4,tests)+complex_ct(5,tests)+complex_ct(6,tests))/
3);


phase_uh_light(tests)=180/pi*angle((complex_uh(4,tests)+complex_uh(5,tests)+complex
_uh(6,tests))/3);

phase_ut_light(tests)=180/pi*angle((complex_ut(4,tests)+complex_ut(5,tests)+complex_
ut(6,tests))/3);

phase_ch_light(tests)=180/pi*angle((complex_ch(4,tests)+complex_ch(5,tests)+complex
_ch(6,tests))/3);

phase_ct_light(tests)=180/pi*angle((complex_ct(4,tests)+complex_ct(5,tests)+complex_c
t(6,tests))/3);
   else
```

```
gain_uh_light(tests)=abs((complex_uh(3,tests)+complex_uh(4,tests)+complex_uh(5,tests
)+complex_uh(6,tests))/4);

gain_ut_light(tests)=abs((complex_ut(3,tests)+complex_ut(4,tests)+complex_ut(5,tests)+
complex_ut(6,tests))/4);

gain_ch_light(tests)=abs((complex_ch(3,tests)+complex_ch(4,tests)+complex_ch(5,tests)
+complex_ch(6,tests))/4);

gain_ct_light(tests)=abs((complex_ct(3,tests)+complex_ct(4,tests)+complex_ct(5,tests)+c
omplex_ct(6,tests))/4);


phase_uh_light(tests)=180/pi*angle((complex_uh(3,tests)+complex_uh(4,tests)+complex
_uh(5,tests)+complex_uh(6,tests))/4);

phase_ut_light(tests)=180/pi*angle((complex_ut(3,tests)+complex_ut(4,tests)+complex_
ut(5,tests)+complex_ut(6,tests))/4);

phase_ch_light(tests)=180/pi*angle((complex_ch(3,tests)+complex_ch(4,tests)+complex
_ch(5,tests)+complex_ch(6,tests))/4);

phase_ct_light(tests)=180/pi*angle((complex_ct(3,tests)+complex_ct(4,tests)+complex_c
t(5,tests)+complex_ct(6,tests))/4);
   end

   if tests==1 | tests==2 | tests==3 | tests==4

gain_uh_dark(tests)=abs((complex_uh(1,tests)+complex_uh(2,tests)+complex_uh(8,tests
))/3);

gain_ut_dark(tests)=abs((complex_ut(1,tests)+complex_ut(2,tests)+complex_ut(8,tests))/
3);

gain_ch_dark(tests)=abs((complex_ch(1,tests)+complex_ch(2,tests)+complex_ch(8,tests)
)/3);

gain_ct_dark(tests)=abs((complex_ct(1,tests)+complex_ct(2,tests)+complex_ct(8,tests))/
3);


phase_uh_dark(tests)=180/pi*angle((complex_uh(1,tests)+complex_uh(2,tests)+complex
_uh(8,tests))/3);
```

```
phase_ut_dark(tests)=180/pi*angle((complex_ut(1,tests)+complex_ut(2,tests)+complex_
ut(8,tests))/3);

phase_ch_dark(tests)=180/pi*angle((complex_ch(1,tests)+complex_ch(2,tests)+complex
_ch(8,tests))/3);

phase_ct_dark(tests)=180/pi*angle((complex_ct(1,tests)+complex_ct(2,tests)+complex_c
t(8,tests))/3);
    else

gain_uh_dark(tests)=abs((complex_uh(1,tests)+complex_uh(2,tests)+complex_uh(7,tests
)+complex_uh(8,tests))/4);

gain_ut_dark(tests)=abs((complex_ut(1,tests)+complex_ut(2,tests)+complex_ut(7,tests)+
complex_ut(8,tests))/4);

gain_ch_dark(tests)=abs((complex_ch(1,tests)+complex_ch(2,tests)+complex_ch(7,tests)
+complex_ch(8,tests))/4);

gain_ct_dark(tests)=abs((complex_ct(1,tests)+complex_ct(2,tests)+complex_ct(7,tests)+c
omplex_ct(8,tests))/4);


phase_uh_dark(tests)=180/pi*angle((complex_uh(1,tests)+complex_uh(2,tests)+complex
_uh(7,tests)+complex_uh(8,tests))/4);

phase_ut_dark(tests)=180/pi*angle((complex_ut(1,tests)+complex_ut(2,tests)+complex_
ut(7,tests)+complex_ut(8,tests))/4);

phase_ch_dark(tests)=180/pi*angle((complex_ch(1,tests)+complex_ch(2,tests)+complex
_ch(7,tests)+complex_ch(8,tests))/4);

phase_ct_dark(tests)=180/pi*angle((complex_ct(1,tests)+complex_ct(2,tests)+complex_c
t(7,tests)+complex_ct(8,tests))/4);
    end

    if tests==12

gain_uh_exp(tests)=abs((complex_uh(1,tests)+complex_uh(2,tests)+complex_uh(4,tests)
)/3);

gain_ut_exp(tests)=abs((complex_ut(1,tests)+complex_ut(2,tests)+complex_ut(4,tests))/3
);
```

```
gain_ch_exp(tests)=abs((complex_ch(1,tests)+complex_ch(2,tests)+complex_ch(4,tests))
/3);

gain_ct_exp(tests)=abs((complex_ct(1,tests)+complex_ct(2,tests)+complex_ct(4,tests))/3
);


phase_uh_exp(tests)=180/pi*angle((complex_uh(1,tests)+complex_uh(2,tests)+complex_
uh(4,tests))/3);

phase_ut_exp(tests)=180/pi*angle((complex_ut(1,tests)+complex_ut(2,tests)+complex_u
t(4,tests))/3);

phase_ch_exp(tests)=180/pi*angle((complex_ch(1,tests)+complex_ch(2,tests)+complex_
ch(4,tests))/3);

phase_ct_exp(tests)=180/pi*angle((complex_ct(1,tests)+complex_ct(2,tests)+complex_ct
(4,tests))/3);
   else

gain_uh_exp(tests)=abs((complex_uh(1,tests)+complex_uh(2,tests)+complex_uh(3,tests)
+complex_uh(4,tests))/4);

gain_ut_exp(tests)=abs((complex_ut(1,tests)+complex_ut(2,tests)+complex_ut(3,tests)+c
omplex_ut(4,tests))/4);

gain_ch_exp(tests)=abs((complex_ch(1,tests)+complex_ch(2,tests)+complex_ch(3,tests)
+complex_ch(4,tests))/4);

gain_ct_exp(tests)=abs((complex_ct(1,tests)+complex_ct(2,tests)+complex_ct(3,tests)+c
omplex_ct(4,tests))/4);


phase_uh_exp(tests)=180/pi*angle((complex_uh(1,tests)+complex_uh(2,tests)+complex_
uh(3,tests)+complex_uh(4,tests))/4);

phase_ut_exp(tests)=180/pi*angle((complex_ut(1,tests)+complex_ut(2,tests)+complex_u
t(3,tests)+complex_ut(4,tests))/4);

phase_ch_exp(tests)=180/pi*angle((complex_ch(1,tests)+complex_ch(2,tests)+complex_
ch(3,tests)+complex_ch(4,tests))/4);

phase_ct_exp(tests)=180/pi*angle((complex_ct(1,tests)+complex_ct(2,tests)+complex_ct
(3,tests)+complex_ct(4,tests))/4);
   end
```

```
if tests==1 | tests==2 | tests==3 | tests==4

gain_uh_nve(tests)=abs((complex_uh(5,tests)+complex_uh(6,tests)+complex_uh(8,tests)
)/3);

gain_ut_nve(tests)=abs((complex_ut(5,tests)+complex_ut(6,tests)+complex_ut(8,tests))/3
);

gain_ch_nve(tests)=abs((complex_ch(5,tests)+complex_ch(6,tests)+complex_ch(8,tests))
/3);

gain_ct_nve(tests)=abs((complex_ct(5,tests)+complex_ct(6,tests)+complex_ct(8,tests))/3
);


phase_uh_nve(tests)=180/pi*angle((complex_uh(5,tests)+complex_uh(6,tests)+complex_
uh(8,tests))/3);

phase_ut_nve(tests)=180/pi*angle((complex_ut(5,tests)+complex_ut(6,tests)+complex_u
t(8,tests))/3);

phase_ch_nve(tests)=180/pi*angle((complex_ch(5,tests)+complex_ch(6,tests)+complex_
ch(8,tests))/3);

phase_ct_nve(tests)=180/pi*angle((complex_ct(5,tests)+complex_ct(6,tests)+complex_ct
(8,tests))/3);
else

gain_uh_nve(tests)=abs((complex_uh(5,tests)+complex_uh(6,tests)+complex_uh(7,tests)
+complex_uh(8,tests))/4);

gain_ut_nve(tests)=abs((complex_ut(5,tests)+complex_ut(6,tests)+complex_ut(7,tests)+c
omplex_ut(8,tests))/4);

gain_ch_nve(tests)=abs((complex_ch(5,tests)+complex_ch(6,tests)+complex_ch(7,tests)
+complex_ch(8,tests))/4);

gain_ct_nve(tests)=abs((complex_ct(5,tests)+complex_ct(6,tests)+complex_ct(7,tests)+c
omplex_ct(8,tests))/4);


phase_uh_nve(tests)=180/pi*angle((complex_uh(5,tests)+complex_uh(6,tests)+complex_
uh(7,tests)+complex_uh(8,tests))/4);
```

```
phase_ut_nve(tests)=180/pi*angle((complex_ut(5,tests)+complex_ut(6,tests)+complex_u
t(7,tests)+complex_ut(8,tests))/4);

phase_ch_nve(tests)=180/pi*angle((complex_ch(5,tests)+complex_ch(6,tests)+complex_
ch(7,tests)+complex_ch(8,tests))/4);

phase_ct_nve(tests)=180/pi*angle((complex_ct(5,tests)+complex_ct(6,tests)+complex_ct
(7,tests)+complex_ct(8,tests))/4);
  end
end

%Light Standard Error
clear complextype realtype imagtype gaintype amptype
type=[{'uh','ut','ch','ct'}];
for i=1:4
  for tests=1:12
    complextype=eval(['complex_',type{i}]);
    realtype=real(complextype);
    imagtype=imag(complextype);

    if tests==12
    A_real=[realtype(5,tests),realtype(6,tests),realtype(4,tests)];
    A_imag=[imagtype(5,tests),imagtype(6,tests),imagtype(4,tests)];
    else
    A_real=[realtype(5,tests),realtype(6,tests),realtype(3,tests),realtype(4,tests)];
    A_imag=[imagtype(5,tests),imagtype(6,tests),imagtype(3,tests),imagtype(4,tests)];
    end

    A_complex = complex(A_real, A_imag);
    A_mean = mean(A_complex, 2);
    A_amp = abs(A_mean);
    A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
    cd /Volumes/Untitled/Analysis_Programs/
    display = 0;   % 1:display, 0:no
    e = [];        % e = []: standard error, e = 1: standard deviation
    A_std = f_covellips_wg(A_complex,e,display);

    gain_light_se(tests,i) = A_std(1);
    phase_light_se(tests,i)  = A_std(2) * 180/pi;

  end
end
```

```
%Dark Standard Error
clear complextype realtype imagtype gaintype amptype
type=[{'uh','ut','ch','ct'}];
for i=1:4
  for tests=1:12
    complextype=eval(['complex_',type{i}]);
    realtype=real(complextype);
    imagtype=imag(complextype);


    if tests==1 | tests==2 | tests==3 | tests==4
    A_real=[realtype(1,tests),realtype(2,tests),realtype(8,tests)];
    A_imag=[imagtype(1,tests),imagtype(2,tests),imagtype(8,tests)];
    else
    A_real=[realtype(1,tests),realtype(2,tests),realtype(7,tests),realtype(8,tests)];
    A_imag=[imagtype(1,tests),imagtype(2,tests),imagtype(7,tests),imagtype(8,tests)];
    end

    A_complex = complex(A_real, A_imag);
    A_mean = mean(A_complex, 2);
    A_amp = abs(A_mean);
    A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
    cd /Volumes/Untitled/Analysis_Programs/
    display = 0;   % 1:display, 0:no
    e = [];        % e = []: standard error, e = 1: standard deviation
    A_std = f_covellips_wg(A_complex,e,display);

    gain_dark_se(tests,i) = A_std(1);
    phase_dark_se(tests,i)  = A_std(2) * 180/pi;

  end
end


%Experienced Standard Error
clear complextype realtype imagtype gaintype amptype
type=[{'uh','ut','ch','ct'}];
for i=1:4
  for tests=1:12
    complextype=eval(['complex_',type{i}]);
    realtype=real(complextype);
    imagtype=imag(complextype);


    if tests==12
    A_real=[realtype(1,tests),realtype(2,tests),realtype(4,tests)];
```

```
    A_imag=[imagtype(1,tests),imagtype(2,tests),imagtype(4,tests)];
    else
    A_real=[realtype(1,tests),realtype(2,tests),realtype(3,tests),realtype(4,tests)];
    A_imag=[imagtype(1,tests),imagtype(2,tests),imagtype(3,tests),imagtype(4,tests)];
    end

    A_complex = complex(A_real, A_imag);
    A_mean = mean(A_complex, 2);
    A_amp = abs(A_mean);
    A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
    cd /Volumes/Untitled/Analysis_Programs/
    display = 0;   % 1:display, 0:no
    e = [];        % e = []: standard error, e = 1: standard deviation
    A_std = f_covellips_wg(A_complex,e,display);

    gain_exp_se(tests,i) = A_std(1);
    phase_exp_se(tests,i)  = A_std(2) * 180/pi;

  end
end


%Naive Standard Error
clear complextype realtype imagtype gaintype amptype
type=[{'uh','ut','ch','ct'}];
for i=1:4
  for tests=1:12
    complextype=eval(['complex_',type{i}]);
    realtype=real(complextype);
    imagtype=imag(complextype);


    if tests==1 | tests==2 | tests==3 | tests==4
    A_real=[realtype(5,tests),realtype(6,tests),realtype(8,tests)];
    A_imag=[imagtype(5,tests),imagtype(6,tests),imagtype(8,tests)];
    else
    A_real=[realtype(5,tests),realtype(6,tests),realtype(7,tests),realtype(8,tests)];
    A_imag=[imagtype(5,tests),imagtype(6,tests),imagtype(7,tests),imagtype(8,tests)];
    end

    A_complex = complex(A_real, A_imag);
    A_mean = mean(A_complex, 2);
    A_amp = abs(A_mean);
    A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
    cd /Volumes/Untitled/Analysis_Programs/
    display = 0;   % 1:display, 0:no
```

```
    e = [];        % e = []: standard error, e = 1: standard deviation
    A_std = f_covellips_wg(A_complex,e,display);

    gain_nve_se(tests,i) = A_std(1);
    phase_nve_se(tests,i)  = A_std(2) * 180/pi;

  end
end






%Test Values
types=[{'uh','ut','ch','ct'}];
for type=1:4
  complextype=eval(['complex_',types{type}]);
  for days=1:3
    for tests=1:4

      one(tests,days,type)=complextype(1,((days-1)*4)+tests);
      two(tests,days,type)=complextype(2,((days-1)*4)+tests);
      three(tests,days,type)=complextype(3,((days-1)*4)+tests);
      four(tests,days,type)=complextype(4,((days-1)*4)+tests);
      five(tests,days,type)=complextype(5,((days-1)*4)+tests);
      six(tests,days,type)=complextype(6,((days-1)*4)+tests);
      seven(tests,days,type)=complextype(7,((days-1)*4)+tests);
      eight(tests,days,type)=complextype(8,((days-1)*4)+tests);

    end
  end
end

%Look for 2 and 7
for type=1:4
  for days=1:3


gain_light_day(days,type)=abs((nansum(three(:,days,type))+nansum(four(:,days,type))+n
ansum(five(:,days,type))+nansum(six(:,days,type)))/15);

phase_light_day(days,type)=180/pi*angle((nansum(three(:,days,type))+nansum(four(:,da
ys,type))+nansum(five(:,days,type))+nansum(six(:,days,type)))/15);
```

```
if days==1

gain_dark_day(days,type)=abs((nansum(one(:,days,type))+nansum(two(:,days,type))+na
nsum(eight(:,days,type)))/12);

phase_dark_day(days,type)=180/pi*angle((nansum(one(:,days,type))+nansum(two(:,days
,type))+nansum(eight(:,days,type)))/12);
    elseif days==3

gain_dark_day(days,type)=abs((nansum(one(:,days,type))+nansum(two(1:3,days,type))+
nansum(seven(:,days,type))+nansum(eight(:,days,type)))/15);

phase_dark_day(days,type)=180/pi*angle((nansum(one(:,days,type))+nansum(two(1:3,da
ys,type))+nansum(seven(:,days,type))+nansum(eight(:,days,type)))/15);
    else

gain_dark_day(days,type)=abs((nansum(one(:,days,type))+nansum(two(:,days,type))+na
nsum(seven(:,days,type))+nansum(eight(:,days,type)))/16);

phase_dark_day(days,type)=180/pi*angle((nansum(one(:,days,type))+nansum(two(:,days
,type))+nansum(seven(:,days,type))+nansum(eight(:,days,type)))/16);
    end

    if days==3

gain_exp_day(days,type)=abs((nansum(three(:,days,type))+nansum(four(:,days,type))+na
nsum(one(:,days,type))+nansum(two(1:3,days,type)))/15);

phase_exp_day(days,type)=180/pi*angle((nansum(three(:,days,type))+nansum(four(:,day
s,type))+nansum(one(:,days,type))+nansum(two(1:3,days,type)))/15);
    else

gain_exp_day(days,type)=abs((nansum(three(:,days,type))+nansum(four(:,days,type))+na
nsum(one(:,days,type))+nansum(two(:,days,type)))/16);

phase_exp_day(days,type)=180/pi*angle((nansum(three(:,days,type))+nansum(four(:,day
s,type))+nansum(one(:,days,type))+nansum(two(:,days,type)))/16);
    end

    if days==1

gain_nve_day(days,type)=abs((nansum(eight(:,days,type))+nansum(five(:,days,type))+na
nsum(six(:,days,type)))/12);
```

```
phase_nve_day(days,type)=180/pi*angle((nansum(eight(:,days,type))+nansum(five(:,day
s,type))+nansum(six(:,days,type)))/12);
      else

gain_nve_day(days,type)=abs((nansum(seven(:,days,type))+nansum(eight(:,days,type))+
nansum(five(:,days,type))+nansum(six(:,days,type)))/16);

phase_nve_day(days,type)=180/pi*angle((nansum(seven(:,days,type))+nansum(eight(:,da
ys,type))+nansum(five(:,days,type))+nansum(six(:,days,type)))/16);
      end
   end
end


%Light Standard Error
for type=1:4
   for days=1:3


A_real=[real(three(:,days,type))',real(four(:,days,type))',real(five(:,days,type))',real(six(:,d
ays,type))'];

A_imag=[imag(three(:,days,type))',imag(four(:,days,type))',imag(five(:,days,type))',imag(
six(:,days,type))'];

      A_complex = complex(A_real, A_imag);
      A_mean = mean(A_complex, 2);
      A_amp = abs(A_mean);
      A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
      cd /Volumes/Untitled/Analysis_Programs/
      display = 0;    % 1:display, 0:no
      e = [ ];        % e = [ ]: standard error, e = 1: standard deviation
      A_std = f_covellips_wg(A_complex,e,display);

      gain_light_day_se(days,type) = A_std(1);
      phase_light_day_se(days,type)  = A_std(2) * 180/pi;

   end
end


%Dark Standard Error
for type=1:4
   for days=1:3
      if days==1
```

```
    A_real=[real(one(:,days,type))',real(two(:,days,type))',real(eight(:,days,type))'];
    A_imag=[imag(one(:,days,type))',imag(two(:,days,type))',imag(eight(:,days,type))'];
    elseif days==3

A_real=[real(one(:,days,type))',real(two(1:3,days,type))',real(seven(:,days,type))',real(eig
ht(:,days,type))'];

A_imag=[imag(one(:,days,type))',imag(two(1:3,days,type))',imag(seven(:,days,type))',im
ag(eight(:,days,type))'];
    else

A_real=[real(one(:,days,type))',real(two(:,days,type))',real(seven(:,days,type))',real(eight(
:,days,type))'];

A_imag=[imag(one(:,days,type))',imag(two(:,days,type))',imag(seven(:,days,type))',imag(
eight(:,days,type))'];
    end

    A_complex = complex(A_real, A_imag);
    A_mean = mean(A_complex, 2);
    A_amp = abs(A_mean);
    A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
    cd /Volumes/Untitled/Analysis_Programs/
    display = 0;   % 1:display, 0:no
    e = [];        % e = []: standard error, e = 1: standard deviation
    A_std = f_covellips_wg(A_complex,e,display);

    gain_dark_day_se(days,type) = A_std(1);
    phase_dark_day_se(days,type)  = A_std(2) * 180/pi;

  end
end

%Experienced Standard Error
for type=1:4
  for days=1:3
    if days==3

A_real=[real(three(:,days,type))',real(four(:,days,type))',real(one(:,days,type))',real(two(1:
3,days,type))'];

A_imag=[imag(three(:,days,type))',imag(four(:,days,type))',imag(one(:,days,type))',imag(
two(1:3,days,type))'];

    else
```

```
A_real=[real(three(:,days,type))',real(four(:,days,type))',real(one(:,days,type))',real(two(:,
days,type))'];

A_imag=[imag(three(:,days,type))',imag(four(:,days,type))',imag(one(:,days,type))',imag(
two(:,days,type))'];
      end

      A_complex = complex(A_real, A_imag);
      A_mean = mean(A_complex, 2);
      A_amp = abs(A_mean);
      A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
      cd /Volumes/Untitled/Analysis_Programs/
      display = 0;   % 1:display, 0:no
      e = [];        % e = []: standard error, e = 1: standard deviation
      A_std = f_covellips_wg(A_complex,e,display);

      gain_exp_day_se(days,type) = A_std(1);
      phase_exp_day_se(days,type)  = A_std(2) * 180/pi;

   end
end


%Naive Standard Error
for type=1:4
   for days=1:3
      if days==1
      A_real=[real(five(:,days,type))',real(six(:,days,type))',real(eight(:,days,type))'];
      A_imag=[imag(five(:,days,type))',imag(six(:,days,type))',imag(eight(:,days,type))'];

      else

A_real=[real(five(:,days,type))',real(six(:,days,type))',real(seven(:,days,type))',real(eight(:
,days,type))'];

A_imag=[imag(five(:,days,type))',imag(six(:,days,type))',imag(seven(:,days,type))',imag(
eight(:,days,type))'];
      end

      A_complex = complex(A_real, A_imag);
      A_mean = mean(A_complex, 2);
      A_amp = abs(A_mean);
      A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
      cd /Volumes/Untitled/Analysis_Programs/
      display = 0;   % 1:display, 0:no
```

```
    e = [];        % e = []: standard error, e = 1: standard deviation
    A_std = f_covellips_wg(A_complex,e,display);

    gain_nve_day_se(days,type) = A_std(1);
    phase_nve_day_se(days,type)  = A_std(2) * 180/pi;

  end
end
```

```
%Stats
%Compare 1st test to last test x
%Compare 1st day to last day x
%Compare overall Manova x
%Specifically for Torsion--Regression x
%1st test to last test for individual

%Light Tests
s_e_light_uh=[complex_uh(3,1),complex_uh(4,1),complex_uh(5,1),complex_uh(6,1),co
mplex_uh(3,12),complex_uh(4,12),complex_uh(5,12),complex_uh(6,12)]';
s_e_light_ut=[complex_ut(3,1),complex_ut(4,1),complex_uh(5,1),complex_uh(6,1),com
plex_ut(3,12),complex_ut(4,12),complex_ut(5,12),complex_ut(6,12)]';
s_e_light_ch=[complex_ch(3,1),complex_ch(4,1),complex_uh(5,1),complex_uh(6,1),co
mplex_ch(3,12),complex_ch(4,12),complex_ch(5,12),complex_ch(6,12)]';
s_e_light_ct=[complex_ct(3,1),complex_ct(4,1),complex_uh(5,1),complex_uh(6,1),comp
lex_ct(3,12),complex_ct(4,12),complex_ct(5,12),complex_ct(6,12)]';

for i=1:length(s_e_light_uh)
   start_end_light_uh(i,1)=real(s_e_light_uh(i));
   start_end_light_uh(i,2)=imag(s_e_light_uh(i));
   start_end_light_ut(i,1)=real(s_e_light_ut(i));
   start_end_light_ut(i,2)=imag(s_e_light_ut(i));
   start_end_light_ch(i,1)=real(s_e_light_ch(i));
   start_end_light_ch(i,2)=imag(s_e_light_ch(i));
   start_end_light_ct(i,1)=real(s_e_light_ct(i));
   start_end_light_ct(i,2)=imag(s_e_light_ct(i));
end

grouping_light(1:4)={'starting'};
grouping_light(5:8)={'ending'};

[d,p,stats]=manova1(start_end_light_uh,grouping_light)
```

```
[d,p,stats]=manova1(start_end_light_ut,grouping_light)
[d,p,stats]=manova1(start_end_light_ch,grouping_light)
[d,p,stats]=manova1(start_end_light_ct,grouping_light)

%Dark Tests
s_e_dark_uh=[complex_uh(1,1),complex_uh(2,1),complex_uh(8,1),complex_uh(1,12),co
mplex_uh(2,12),complex_uh(7,12),complex_uh(8,12)]';
s_e_dark_ut=[complex_ut(1,1),complex_ut(2,1),complex_ut(8,1),complex_ut(1,12),com
plex_ut(2,12),complex_ut(7,12),complex_ut(8,12)]';
s_e_dark_ch=[complex_ch(1,1),complex_ch(2,1),complex_ch(8,1),complex_ch(1,12),co
mplex_ch(2,12),complex_ch(7,12),complex_ch(8,12)]';
s_e_dark_ct=[complex_ct(1,1),complex_ct(2,1),complex_ct(8,1),complex_ct(1,12),comp
lex_ct(2,12),complex_ct(7,12),complex_ct(8,12)]';

for i=1:length(s_e_dark_uh)
   start_end_dark_uh(i,1)=real(s_e_dark_uh(i));
   start_end_dark_uh(i,2)=imag(s_e_dark_uh(i));
   start_end_dark_ut(i,1)=real(s_e_dark_ut(i));
   start_end_dark_ut(i,2)=imag(s_e_dark_ut(i));
   start_end_dark_ch(i,1)=real(s_e_dark_ch(i));
   start_end_dark_ch(i,2)=imag(s_e_dark_ch(i));
   start_end_dark_ct(i,1)=real(s_e_dark_ct(i));
   start_end_dark_ct(i,2)=imag(s_e_dark_ct(i));
end

grouping_dark(1:3)={'starting'};
grouping_dark(4:7)={'ending'};

[d,p,stats]=manova1(start_end_dark_uh,grouping_dark)
[d,p,stats]=manova1(start_end_dark_ut,grouping_dark)
[d,p,stats]=manova1(start_end_dark_ch,grouping_dark)
[d,p,stats]=manova1(start_end_dark_ct,grouping_dark)


%Experienced Tests
s_e_exp_uh=[complex_uh(1,1),complex_uh(2,1),complex_uh(3,1),complex_uh(4,1),com
plex_uh(1,12),complex_uh(2,12),complex_uh(3,12),complex_uh(4,12)]';
s_e_exp_ut=[complex_ut(1,1),complex_ut(2,1),complex_ut(3,1),complex_uh(4,1),compl
ex_ut(1,12),complex_ut(2,12),complex_ut(3,12),complex_ut(4,12)]';
s_e_exp_ch=[complex_ch(1,1),complex_ch(2,1),complex_ch(3,1),complex_uh(4,1),com
plex_ch(1,12),complex_ch(2,12),complex_ch(3,12),complex_ch(4,12)]';
s_e_exp_ct=[complex_ct(1,1),complex_ct(2,1),complex_ct(3,1),complex_uh(4,1),comple
x_ct(1,12),complex_ct(2,12),complex_ct(3,12),complex_ct(4,12)]';

for i=1:length(s_e_exp_uh)
   start_end_exp_uh(i,1)=real(s_e_exp_uh(i));
```

```
  start_end_exp_uh(i,2)=imag(s_e_exp_uh(i));
  start_end_exp_ut(i,1)=real(s_e_exp_ut(i));
  start_end_exp_ut(i,2)=imag(s_e_exp_ut(i));
  start_end_exp_ch(i,1)=real(s_e_exp_ch(i));
  start_end_exp_ch(i,2)=imag(s_e_exp_ch(i));
  start_end_exp_ct(i,1)=real(s_e_exp_ct(i));
  start_end_exp_ct(i,2)=imag(s_e_exp_ct(i));
end

grouping_exp(1:4)={'starting'};
grouping_exp(4:8)={'ending'};

[d,p,stats]=manova1(start_end_exp_uh,grouping_exp)
[d,p,stats]=manova1(start_end_exp_ut,grouping_exp)
[d,p,stats]=manova1(start_end_exp_ch,grouping_exp)
[d,p,stats]=manova1(start_end_exp_ct,grouping_exp)


%Naive Tests
s_e_nve_uh=[complex_uh(5,1),complex_uh(6,1),complex_uh(8,1),complex_uh(5,12),co
mplex_uh(6,12),complex_uh(7,12),complex_uh(8,12)]';
s_e_nve_ut=[complex_ut(5,1),complex_ut(6,1),complex_uh(8,1),complex_ut(5,12),comp
lex_ut(6,12),complex_ut(7,12),complex_ut(8,12)]';
s_e_nve_ch=[complex_ch(5,1),complex_ch(6,1),complex_uh(8,1),complex_ch(5,12),co
mplex_ch(6,12),complex_ch(7,12),complex_ch(8,12)]';
s_e_nve_ct=[complex_ct(5,1),complex_ct(6,1),complex_uh(8,1),complex_ct(5,12),compl
ex_ct(6,12),complex_ct(7,12),complex_ct(8,12)]';

for i=1:length(s_e_nve_uh)
  start_end_nve_uh(i,1)=real(s_e_nve_uh(i));
  start_end_nve_uh(i,2)=imag(s_e_nve_uh(i));
  start_end_nve_ut(i,1)=real(s_e_nve_ut(i));
  start_end_nve_ut(i,2)=imag(s_e_nve_ut(i));
  start_end_nve_ch(i,1)=real(s_e_nve_ch(i));
  start_end_nve_ch(i,2)=imag(s_e_nve_ch(i));
  start_end_nve_ct(i,1)=real(s_e_nve_ct(i));
  start_end_nve_ct(i,2)=imag(s_e_nve_ct(i));
end

grouping_nve(1:3)={'starting'};
grouping_nve(4:7)={'ending'};

[d,p,stats]=manova1(start_end_nve_uh,grouping_nve)
[d,p,stats]=manova1(start_end_nve_ut,grouping_nve)
[d,p,stats]=manova1(start_end_nve_ch,grouping_nve)
[d,p,stats]=manova1(start_end_nve_ct,grouping_nve)
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Light Days
s_e_light_uh_days=[complex_uh(3,1),complex_uh(3,2),complex_uh(3,3),complex_uh(3,
4),complex_uh(4,1),complex_uh(4,2),complex_uh(4,3),complex_uh(4,4),complex_uh(5,
1),complex_uh(5,2),complex_uh(5,3),complex_uh(5,4),complex_uh(6,1),complex_uh(6,
2),complex_uh(6,3),complex_uh(6,4),complex_uh(3,9),complex_uh(3,10),complex_uh(3
,11),complex_uh(3,12),complex_uh(4,9),complex_uh(4,10),complex_uh(4,11),complex_
uh(4,12),complex_uh(5,9),complex_uh(5,10),complex_uh(5,11),complex_uh(5,12),comp
lex_uh(6,9),complex_uh(6,10),complex_uh(6,11),complex_uh(6,12)];
s_e_light_ut_days=[complex_ut(3,1),complex_ut(3,2),complex_ut(3,3),complex_ut(3,4),
complex_ut(4,1),complex_ut(4,2),complex_ut(4,3),complex_ut(4,4),complex_ut(5,1),co
mplex_ut(5,2),complex_ut(5,3),complex_ut(5,4),complex_ut(6,1),complex_ut(6,2),compl
ex_ut(6,3),complex_ut(6,4),complex_ut(3,9),complex_ut(3,10),complex_ut(3,11),comple
x_ut(3,12),complex_ut(4,9),complex_ut(4,10),complex_ut(4,11),complex_ut(4,12),compl
ex_ut(5,9),complex_ut(5,10),complex_ut(5,11),complex_ut(5,12),complex_ut(6,9),compl
ex_ut(6,10),complex_ut(6,11),complex_ut(6,12)];
s_e_light_ch_days=[complex_ch(3,1),complex_ch(3,2),complex_ch(3,3),complex_ch(3,4
),complex_ch(4,1),complex_ch(4,2),complex_ch(4,3),complex_ch(4,4),complex_ch(5,1),
complex_ch(5,2),complex_ch(5,3),complex_ch(5,4),complex_ch(6,1),complex_ch(6,2),c
omplex_ch(6,3),complex_ch(6,4),complex_ch(3,9),complex_ch(3,10),complex_ch(3,11),
complex_ch(3,12),complex_ch(4,9),complex_ch(4,10),complex_ch(4,11),complex_ch(4,
12),complex_ch(5,9),complex_ch(5,10),complex_ch(5,11),complex_ch(5,12),complex_c
h(6,9),complex_ch(6,10),complex_ch(6,11),complex_ch(6,12)];
s_e_light_ct_days=[complex_ct(3,1),complex_ct(3,2),complex_ct(3,3),complex_ct(3,4),c
omplex_ct(4,1),complex_ct(4,2),complex_ct(4,3),complex_ct(4,4),complex_ct(5,1),comp
lex_ct(5,2),complex_ct(5,3),complex_ct(5,4),complex_ct(6,1),complex_ct(6,2),complex_
ct(6,3),complex_ct(6,4),complex_ct(3,9),complex_ct(3,10),complex_ct(3,11),complex_ct
(3,12),complex_ct(4,9),complex_ct(4,10),complex_ct(4,11),complex_ct(4,12),complex_c
t(5,9),complex_ct(5,10),complex_ct(5,11),complex_ct(5,12),complex_ct(6,9),complex_ct
(6,10),complex_ct(6,11),complex_ct(6,12)];

for i=1:length(s_e_light_uh_days)
   start_end_light_uh_days(i,1)=real(s_e_light_uh_days(i));
   start_end_light_uh_days(i,2)=imag(s_e_light_uh_days(i));
   start_end_light_ut_days(i,1)=real(s_e_light_ut_days(i));
   start_end_light_ut_days(i,2)=imag(s_e_light_ut_days(i));
   start_end_light_ch_days(i,1)=real(s_e_light_ch_days(i));
   start_end_light_ch_days(i,2)=imag(s_e_light_ch_days(i));
   start_end_light_ct_days(i,1)=real(s_e_light_ct_days(i));
   start_end_light_ct_days(i,2)=imag(s_e_light_ct_days(i));
end

grouping_light_days(1:16)={'starting'};
grouping_light_days(17:32)={'ending'};
```

```
[d,p,stats]=manova1(start_end_light_uh_days,grouping_light_days)
[d,p,stats]=manova1(start_end_light_ut_days,grouping_light_days)
[d,p,stats]=manova1(start_end_light_ch_days,grouping_light_days)
[d,p,stats]=manova1(start_end_light_ct_days,grouping_light_days)

%Dark Days
s_e_dark_uh_days=[complex_uh(1,1),complex_uh(1,2),complex_uh(1,3),complex_uh(1,
4),complex_uh(2,1),complex_uh(2,2),complex_uh(2,3),complex_uh(2,4),complex_uh(8,
1),complex_uh(8,2),complex_uh(8,3),complex_uh(8,4),complex_uh(1,9),complex_uh(1,
10),complex_uh(1,11),complex_uh(1,12),complex_uh(2,9),complex_uh(2,10),complex_u
h(2,11),complex_uh(2,12),complex_uh(7,9),complex_uh(7,10),complex_uh(7,11),compl
ex_uh(7,12),complex_uh(8,9),complex_uh(8,10),complex_uh(8,11),complex_uh(8,12)];
s_e_dark_ut_days=[complex_ut(1,1),complex_ut(1,2),complex_ut(1,3),complex_ut(1,4),
complex_ut(2,1),complex_ut(2,2),complex_ut(2,3),complex_ut(2,4),complex_ut(8,1),co
mplex_ut(8,2),complex_ut(8,3),complex_ut(8,4),complex_ut(1,9),complex_ut(1,10),com
plex_ut(1,11),complex_ut(1,12),complex_ut(2,9),complex_ut(2,10),complex_ut(2,11),co
mplex_ut(2,12),complex_ut(7,9),complex_ut(7,10),complex_ut(7,11),complex_ut(7,12),c
omplex_ut(8,9),complex_ut(8,10),complex_ut(8,11),complex_ut(8,12)];
s_e_dark_ch_days=[complex_ch(1,1),complex_ch(1,2),complex_ch(1,3),complex_ch(1,4
),complex_ch(2,1),complex_ch(2,2),complex_ch(2,3),complex_ch(2,4),complex_ch(8,1),
complex_ch(8,2),complex_ch(8,3),complex_ch(8,4),complex_ch(1,9),complex_ch(1,10),
complex_ch(1,11),complex_ch(1,12),complex_ch(2,9),complex_ch(2,10),complex_ch(2,
11),complex_ch(2,12),complex_ch(7,9),complex_ch(7,10),complex_ch(7,11),complex_c
h(7,12),complex_ch(8,9),complex_ch(8,10),complex_ch(8,11),complex_ch(8,12)];
s_e_dark_ct_days=[complex_ct(1,1),complex_ct(1,2),complex_ct(1,3),complex_ct(1,4),c
omplex_ct(2,1),complex_ct(2,2),complex_ct(2,3),complex_ct(2,4),complex_ct(8,1),comp
lex_ct(8,2),complex_ct(8,3),complex_ct(8,4),complex_ct(1,9),complex_ct(1,10),complex
_ct(1,11),complex_ct(1,12),complex_ct(2,9),complex_ct(2,10),complex_ct(2,11),comple
x_ct(2,12),complex_ct(7,9),complex_ct(7,10),complex_ct(7,11),complex_ct(7,12),compl
ex_ct(8,9),complex_ct(8,10),complex_ct(8,11),complex_ct(8,12)];

for i=1:length(s_e_dark_uh_days)
   start_end_dark_uh_days(i,1)=real(s_e_dark_uh_days(i));
   start_end_dark_uh_days(i,2)=imag(s_e_dark_uh_days(i));
   start_end_dark_ut_days(i,1)=real(s_e_dark_ut_days(i));
   start_end_dark_ut_days(i,2)=imag(s_e_dark_ut_days(i));
   start_end_dark_ch_days(i,1)=real(s_e_dark_ch_days(i));
   start_end_dark_ch_days(i,2)=imag(s_e_dark_ch_days(i));
   start_end_dark_ct_days(i,1)=real(s_e_dark_ct_days(i));
   start_end_dark_ct_days(i,2)=imag(s_e_dark_ct_days(i));
end

grouping_dark_days(1:12)={'starting'};
grouping_dark_days(13:28)={'ending'};
```

```
[d,p,stats]=manova1(start_end_dark_uh_days,grouping_dark_days)
[d,p,stats]=manova1(start_end_dark_ut_days,grouping_dark_days)
[d,p,stats]=manova1(start_end_dark_ch_days,grouping_dark_days)
[d,p,stats]=manova1(start_end_dark_ct_days,grouping_dark_days)


%Experienced Days
s_e_exp_uh_days=[complex_uh(3,1),complex_uh(3,2),complex_uh(3,3),complex_uh(3,4
),complex_uh(4,1),complex_uh(4,2),complex_uh(4,3),complex_uh(4,4),complex_uh(1,1)
,complex_uh(1,2),complex_uh(1,3),complex_uh(1,4),complex_uh(2,1),complex_uh(2,2),
complex_uh(2,3),complex_uh(2,4),complex_uh(3,9),complex_uh(3,10),complex_uh(3,11
),complex_uh(3,12),complex_uh(4,9),complex_uh(4,10),complex_uh(4,11),complex_uh(
4,12),complex_uh(1,9),complex_uh(1,10),complex_uh(1,11),complex_uh(1,12),complex
_uh(2,9),complex_uh(2,10),complex_uh(2,11),complex_uh(2,12)];
s_e_exp_ut_days=[complex_ut(3,1),complex_ut(3,2),complex_ut(3,3),complex_ut(3,4),c
omplex_ut(4,1),complex_ut(4,2),complex_ut(4,3),complex_ut(4,4),complex_ut(1,1),com
plex_ut(1,2),complex_ut(1,3),complex_ut(1,4),complex_ut(2,1),complex_ut(2,2),comple
x_ut(2,3),complex_ut(2,4),complex_ut(3,9),complex_ut(3,10),complex_ut(3,11),complex
_ut(3,12),complex_ut(4,9),complex_ut(4,10),complex_ut(4,11),complex_ut(4,12),comple
x_ut(1,9),complex_ut(1,10),complex_ut(1,11),complex_ut(1,12),complex_ut(2,9),comple
x_ut(2,10),complex_ut(2,11),complex_ut(2,12)];
s_e_exp_ch_days=[complex_ch(3,1),complex_ch(3,2),complex_ch(3,3),complex_ch(3,4)
,complex_ch(4,1),complex_ch(4,2),complex_ch(4,3),complex_ch(4,4),complex_ch(1,1),c
omplex_ch(1,2),complex_ch(1,3),complex_ch(1,4),complex_ch(2,1),complex_ch(2,2),co
mplex_ch(2,3),complex_ch(2,4),complex_ch(3,9),complex_ch(3,10),complex_ch(3,11),c
omplex_ch(3,12),complex_ch(4,9),complex_ch(4,10),complex_ch(4,11),complex_ch(4,1
2),complex_ch(1,9),complex_ch(1,10),complex_ch(1,11),complex_ch(1,12),complex_ch
(2,9),complex_ch(2,10),complex_ch(2,11),complex_ch(2,12)];
s_e_exp_ct_days=[complex_ct(3,1),complex_ct(3,2),complex_ct(3,3),complex_ct(3,4),co
mplex_ct(4,1),complex_ct(4,2),complex_ct(4,3),complex_ct(4,4),complex_ct(1,1),compl
ex_ct(1,2),complex_ct(1,3),complex_ct(1,4),complex_ct(2,1),complex_ct(2,2),complex_
ct(2,3),complex_ct(2,4),complex_ct(3,9),complex_ct(3,10),complex_ct(3,11),complex_ct
(3,12),complex_ct(4,9),complex_ct(4,10),complex_ct(4,11),complex_ct(4,12),complex_c
t(1,9),complex_ct(1,10),complex_ct(1,11),complex_ct(1,12),complex_ct(2,9),complex_ct
(2,10),complex_ct(2,11),complex_ct(2,12)];

for i=1:length(s_e_exp_uh_days)
   start_end_exp_uh_days(i,1)=real(s_e_exp_uh_days(i));
   start_end_exp_uh_days(i,2)=imag(s_e_exp_uh_days(i));
   start_end_exp_ut_days(i,1)=real(s_e_exp_ut_days(i));
   start_end_exp_ut_days(i,2)=imag(s_e_exp_ut_days(i));
   start_end_exp_ch_days(i,1)=real(s_e_exp_ch_days(i));
   start_end_exp_ch_days(i,2)=imag(s_e_exp_ch_days(i));
   start_end_exp_ct_days(i,1)=real(s_e_exp_ct_days(i));
   start_end_exp_ct_days(i,2)=imag(s_e_exp_ct_days(i));
end
```

```
grouping_exp_days(1:16)={'starting'};
grouping_exp_days(17:32)={'ending'};

[d,p,stats]=manova1(start_end_exp_uh_days,grouping_exp_days)
[d,p,stats]=manova1(start_end_exp_ut_days,grouping_exp_days)
[d,p,stats]=manova1(start_end_exp_ch_days,grouping_exp_days)
[d,p,stats]=manova1(start_end_exp_ct_days,grouping_exp_days)


%Naive Days
s_e_nve_uh_days=[complex_uh(5,1),complex_uh(5,2),complex_uh(5,3),complex_uh(5,4
),complex_uh(6,1),complex_uh(6,2),complex_uh(6,3),complex_uh(6,4),complex_uh(8,1)
,complex_uh(8,2),complex_uh(8,3),complex_uh(8,4),complex_uh(5,9),complex_uh(5,10
),complex_uh(5,11),complex_uh(5,12),complex_uh(6,9),complex_uh(6,10),complex_uh(
6,11),complex_uh(6,12),complex_uh(7,9),complex_uh(7,10),complex_uh(7,11),complex
_uh(7,12),complex_uh(8,9),complex_uh(8,10),complex_uh(8,11),complex_uh(8,12)];
s_e_nve_ut_days=[complex_ut(5,1),complex_ut(5,2),complex_ut(5,3),complex_ut(5,4),c
omplex_ut(6,1),complex_ut(6,2),complex_ut(6,3),complex_ut(6,4),complex_ut(8,1),com
plex_ut(8,2),complex_ut(8,3),complex_ut(8,4),complex_ut(5,9),complex_ut(5,10),compl
ex_ut(5,11),complex_ut(5,12),complex_ut(6,9),complex_ut(6,10),complex_ut(6,11),com
plex_ut(6,12),complex_ut(7,9),complex_ut(7,10),complex_ut(7,11),complex_ut(7,12),co
mplex_ut(8,9),complex_ut(8,10),complex_ut(8,11),complex_ut(8,12)];
s_e_nve_ch_days=[complex_ch(5,1),complex_ch(5,2),complex_ch(5,3),complex_ch(5,4)
,complex_ch(6,1),complex_ch(6,2),complex_ch(6,3),complex_ch(6,4),complex_ch(8,1),c
omplex_ch(8,2),complex_ch(8,3),complex_ch(8,4),complex_ch(5,9),complex_ch(5,10),c
omplex_ch(5,11),complex_ch(5,12),complex_ch(6,9),complex_ch(6,10),complex_ch(6,1
1),complex_ch(6,12),complex_ch(7,9),complex_ch(7,10),complex_ch(7,11),complex_ch
(7,12),complex_ch(8,9),complex_ch(8,10),complex_ch(8,11),complex_ch(8,12)];
s_e_nve_ct_days=[complex_ct(5,1),complex_ct(5,2),complex_ct(5,3),complex_ct(5,4),co
mplex_ct(6,1),complex_ct(6,2),complex_ct(6,3),complex_ct(6,4),complex_ct(8,1),compl
ex_ct(8,2),complex_ct(8,3),complex_ct(8,4),complex_ct(5,9),complex_ct(5,10),complex
_ct(5,11),complex_ct(5,12),complex_ct(6,9),complex_ct(6,10),complex_ct(6,11),comple
x_ct(6,12),complex_ct(7,9),complex_ct(7,10),complex_ct(7,11),complex_ct(7,12),compl
ex_ct(8,9),complex_ct(8,10),complex_ct(8,11),complex_ct(8,12)];

for i=1:length(s_e_nve_uh_days)
   start_end_nve_uh_days(i,1)=real(s_e_nve_uh_days(i));
   start_end_nve_uh_days(i,2)=imag(s_e_nve_uh_days(i));
   start_end_nve_ut_days(i,1)=real(s_e_nve_ut_days(i));
   start_end_nve_ut_days(i,2)=imag(s_e_nve_ut_days(i));
   start_end_nve_ch_days(i,1)=real(s_e_nve_ch_days(i));
   start_end_nve_ch_days(i,2)=imag(s_e_nve_ch_days(i));
   start_end_nve_ct_days(i,1)=real(s_e_nve_ct_days(i));
   start_end_nve_ct_days(i,2)=imag(s_e_nve_ct_days(i));
end
```

```
grouping_nve_days(1:12)={'starting'};
grouping_nve_days(13:28)={'ending'};

[d,p,stats]=manova1(start_end_nve_uh_days,grouping_nve_days)
[d,p,stats]=manova1(start_end_nve_ut_days,grouping_nve_days)
[d,p,stats]=manova1(start_end_nve_ch_days,grouping_nve_days)
[d,p,stats]=manova1(start_end_nve_ct_days,grouping_nve_days)


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for subject=1:8
   for tests=1:12
      total_uh(((subject-1)*12)+tests,1)=real(complex_uh(subject,tests));
      total_uh(((subject-1)*12)+tests,2)=imag(complex_uh(subject,tests));

      total_ut(((subject-1)*12)+tests,1)=real(complex_ut(subject,tests));
      total_ut(((subject-1)*12)+tests,2)=imag(complex_ut(subject,tests));

      total_ch(((subject-1)*12)+tests,1)=real(complex_ch(subject,tests));
      total_ch(((subject-1)*12)+tests,2)=imag(complex_ch(subject,tests));

      total_ct(((subject-1)*12)+tests,1)=real(complex_ct(subject,tests));
      total_ct(((subject-1)*12)+tests,2)=imag(complex_ct(subject,tests));
   end
end


%Removes subject 7, tests 1:4
for i=73:92;
   total_uh(i,1)=total_uh(i+4,1);
   total_uh(i,2)=total_uh(i+4,2);

   total_ut(i,1)=total_ut(i+4,1);
   total_ut(i,2)=total_ut(i+4,2);

   total_ch(i,1)=total_ch(i+4,1);
   total_ch(i,2)=total_ch(i+4,2);

   total_ct(i,1)=total_ct(i+4,1);
   total_ct(i,2)=total_ct(i+4,2);
end

for i=1:92
   total_uh_new(i,1)=total_uh(i,1);
```

```
  total_uh_new(i,2)=total_uh(i,2);

  total_ut_new(i,1)=total_ut(i,1);
  total_ut_new(i,2)=total_ut(i,2);

  total_ch_new(i,1)=total_ch(i,1);
  total_ch_new(i,2)=total_ch(i,2);

  total_ct_new(i,1)=total_ct(i,1);
  total_ct_new(i,2)=total_ct(i,2);
end

grouping_ld_total(25:72)={'Light'};
grouping_ld_total(1:24)={'Dark'};
grouping_ld_total(73:92)={'Dark'};

[d,p,stats]=manova1(total_uh_new,grouping_ld_total)
[d,p,stats]=manova1(total_ut_new,grouping_ld_total)
[d,p,stats]=manova1(total_ch_new,grouping_ld_total)
[d,p,stats]=manova1(total_ct_new,grouping_ld_total)

grouping_en_total(48:92)={'Naive'};
grouping_en_total(1:48)={'Expereinced'};

[d,p,stats]=manova1(total_uh_new,grouping_en_total)
[d,p,stats]=manova1(total_ut_new,grouping_en_total)
[d,p,stats]=manova1(total_ch_new,grouping_en_total)
[d,p,stats]=manova1(total_ct_new,grouping_en_total)


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%Regression Analysis--see Excel


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Individual Comparisons for all subjects, regardless of groups
individ_uh=[complex_uh(1,1),complex_uh(2,1),complex_uh(3,1),complex_uh(4,1),comp
lex_uh(5,1),complex_uh(6,1),complex_uh(8,1),complex_uh(1,12),complex_uh(2,12),co
mplex_uh(3,12),complex_uh(4,12),complex_uh(5,12),complex_uh(6,12),complex_uh(7,
12),complex_uh(8,12)];
individ_ut=[complex_ut(1,1),complex_ut(2,1),complex_ut(3,1),complex_ut(4,1),comple
x_ut(5,1),complex_ut(6,1),complex_ut(8,1),complex_ut(1,12),complex_ut(2,12),complex
_ut(3,12),complex_ut(4,12),complex_ut(5,12),complex_ut(6,12),complex_ut(7,12),compl
ex_ut(8,12)];
```

```
individ_ch=[complex_ch(1,1),complex_ch(2,1),complex_ch(3,1),complex_ch(4,1),compl
ex_ch(5,1),complex_ch(6,1),complex_ch(8,1),complex_ch(1,12),complex_ch(2,12),comp
lex_ch(3,12),complex_ch(4,12),complex_ch(5,12),complex_ch(6,12),complex_ch(7,12),c
omplex_ch(8,12)];
individ_ct=[complex_ct(1,1),complex_ct(2,1),complex_ct(3,1),complex_ct(4,1),complex
_ct(5,1),complex_ct(6,1),complex_ct(8,1),complex_ct(1,12),complex_ct(2,12),complex_
ct(3,12),complex_ct(4,12),complex_ct(5,12),complex_ct(6,12),complex_ct(7,12),comple
x_ct(8,12)];

for z=1:length(individ_uh)
   individual_stats_uh(z,1)=real(individ_uh(z));
   individual_stats_uh(z,2)=imag(individ_uh(z));

   individual_stats_ut(z,1)=real(individ_ut(z));
   individual_stats_ut(z,2)=imag(individ_ut(z));

   individual_stats_ch(z,1)=real(individ_ch(z));
   individual_stats_ch(z,2)=imag(individ_ch(z));

   individual_stats_ct(z,1)=real(individ_ct(z));
   individual_stats_ct(z,2)=imag(individ_ct(z));
end

individual_grouping(1:7)={'first'};
individual_grouping(8:15)={'last'};

[d,p,stats]=manova1(individual_stats_uh,individual_grouping)
[d,p,stats]=manova1(individual_stats_ut,individual_grouping)
[d,p,stats]=manova1(individual_stats_ch,individual_grouping)
[d,p,stats]=manova1(individual_stats_ct,individual_grouping)

%Individual Comparisons for all subjects, segregated by group

%Light
all=[1:15];
erase=[1 2 7 8 9 14 15];
individ_uh_demo=individ_uh(setxor(all,erase));
individ_ut_demo=individ_ut(setxor(all,erase));
individ_ch_demo=individ_ch(setxor(all,erase));
individ_ct_demo=individ_ct(setxor(all,erase));

clear z
for z=1:length(individ_uh_demo)
   individ_uh_light(z,1)=real(individ_uh_demo(z));
   individ_uh_light(z,2)=imag(individ_uh_demo(z));
```

```
    individ_ut_light(z,1)=real(individ_ut_demo(z));
    individ_ut_light(z,2)=imag(individ_ut_demo(z));

    individ_ch_light(z,1)=real(individ_ch_demo(z));
    individ_ch_light(z,2)=imag(individ_ch_demo(z));

    individ_ct_light(z,1)=real(individ_ct_demo(z));
    individ_ct_light(z,2)=imag(individ_ct_demo(z));
end

individual_light(1:4)={'first'};
individual_light(5:8)={'last'};

[d,p,stats]=manova1(individ_uh_light,individual_light)
[d,p,stats]=manova1(individ_ut_light,individual_light)
[d,p,stats]=manova1(individ_ch_light,individual_light)
[d,p,stats]=manova1(individ_ct_light,individual_light)

%Dark
all=[1:15];
erase=[3 4 5 6 10 11 12 13];
individ_uh_demo=individ_uh(setxor(all,erase));
individ_ut_demo=individ_ut(setxor(all,erase));
individ_ch_demo=individ_ch(setxor(all,erase));
individ_ct_demo=individ_ct(setxor(all,erase));

clear z
for z=1:length(individ_uh_demo)
    individ_uh_dark(z,1)=real(individ_uh_demo(z));
    individ_uh_dark(z,2)=imag(individ_uh_demo(z));

    individ_ut_dark(z,1)=real(individ_ut_demo(z));
    individ_ut_dark(z,2)=imag(individ_ut_demo(z));

    individ_ch_dark(z,1)=real(individ_ch_demo(z));
    individ_ch_dark(z,2)=imag(individ_ch_demo(z));

    individ_ct_dark(z,1)=real(individ_ct_demo(z));
    individ_ct_dark(z,2)=imag(individ_ct_demo(z));
end

individual_dark(1:3)={'first'};
individual_dark(4:7)={'last'};

[d,p,stats]=manova1(individ_uh_dark,individual_dark)
[d,p,stats]=manova1(individ_ut_dark,individual_dark)
```

```
[d,p,stats]=manova1(individ_ch_dark,individual_dark)
[d,p,stats]=manova1(individ_ct_dark,individual_dark)


%Experienced
all=[1:15];
erase=[5 6 7 12 13 14 15];
individ_uh_demo=individ_uh(setxor(all,erase));
individ_ut_demo=individ_ut(setxor(all,erase));
individ_ch_demo=individ_ch(setxor(all,erase));
individ_ct_demo=individ_ct(setxor(all,erase));

clear z
for z=1:length(individ_uh_demo)
   individ_uh_exp(z,1)=real(individ_uh_demo(z));
   individ_uh_exp(z,2)=imag(individ_uh_demo(z));

   individ_ut_exp(z,1)=real(individ_ut_demo(z));
   individ_ut_exp(z,2)=imag(individ_ut_demo(z));

   individ_ch_exp(z,1)=real(individ_ch_demo(z));
   individ_ch_exp(z,2)=imag(individ_ch_demo(z));

   individ_ct_exp(z,1)=real(individ_ct_demo(z));
   individ_ct_exp(z,2)=imag(individ_ct_demo(z));
end

individual_exp(1:4)={'first'};
individual_exp(5:8)={'last'};

[d,p,stats]=manova1(individ_uh_exp,individual_exp)
[d,p,stats]=manova1(individ_ut_exp,individual_exp)
[d,p,stats]=manova1(individ_ch_exp,individual_exp)
[d,p,stats]=manova1(individ_ct_exp,individual_exp)


%Naive
all=[1:15];
erase=[1:4 8:11];
individ_uh_demo=individ_uh(setxor(all,erase));
individ_ut_demo=individ_ut(setxor(all,erase));
individ_ch_demo=individ_ch(setxor(all,erase));
individ_ct_demo=individ_ct(setxor(all,erase));

clear z
for z=1:length(individ_uh_demo)
```

```
  individ_uh_nve(z,1)=real(individ_uh_demo(z));
  individ_uh_nve(z,2)=imag(individ_uh_demo(z));

  individ_ut_nve(z,1)=real(individ_ut_demo(z));
  individ_ut_nve(z,2)=imag(individ_ut_demo(z));

  individ_ch_nve(z,1)=real(individ_ch_demo(z));
  individ_ch_nve(z,2)=imag(individ_ch_demo(z));

  individ_ct_nve(z,1)=real(individ_ct_demo(z));
  individ_ct_nve(z,2)=imag(individ_ct_demo(z));
end

individual_nve(1:3)={'first'};
individual_nve(4:7)={'last'};

[d,p,stats]=manova1(individ_uh_nve,individual_nve)
[d,p,stats]=manova1(individ_ut_nve,individual_nve)
[d,p,stats]=manova1(individ_ch_nve,individual_nve)
[d,p,stats]=manova1(individ_ct_nve,individual_nve)


%Light vs. Dark Test 1
day1_uh=[complex_uh(1,1),complex_uh(2,1),complex_uh(8,1),complex_uh(3,1),comple
x_uh(4,1),complex_uh(5,1),complex_uh(6,1)];
day1_ut=[complex_ut(1,1),complex_ut(2,1),complex_ut(8,1),complex_ut(3,1),complex_
ut(4,1),complex_ut(5,1),complex_ut(6,1)];
day1_ch=[complex_ch(1,1),complex_ch(2,1),complex_ch(8,1),complex_ch(3,1),complex
_ch(4,1),complex_ch(5,1),complex_ch(6,1)];
day1_ct=[complex_ct(1,1),complex_ct(2,1),complex_ct(8,1),complex_ct(3,1),complex_c
t(4,1),complex_ct(5,1),complex_ct(6,1)];

day1_grouping(1:3)={'Light'};
day1_grouping(4:7)={'Dark'};

for i=1:7
  day1_uh_final(i,1)=real(day1_uh(i));
  day1_uh_final(i,2)=imag(day1_uh(i));

  day1_ut_final(i,1)=real(day1_ut(i));
  day1_ut_final(i,2)=imag(day1_ut(i));

  day1_ch_final(i,1)=real(day1_ch(i));
  day1_ch_final(i,2)=imag(day1_ch(i));

  day1_ct_final(i,1)=real(day1_ct(i));
```

```
    day1_ct_final(i,2)=imag(day1_ct(i));
end

[d,p,stats]=manova1(day1_uh_final,day1_grouping)
[d,p,stats]=manova1(day1_ut_final,day1_grouping)
[d,p,stats]=manova1(day1_ch_final,day1_grouping)
[d,p,stats]=manova1(day1_ct_final,day1_grouping)


%Stats
%group 1 = Light vs. Dark
%group 2 = Experienced vs. Naive
%group 3 = Test Number

group1(1:24)={'Dark'};
group1(25:72)={'Light'};
group1(73:96)={'Dark'};

group2(1:48)={'Experienced'};
group2(49:96)={'Naive'};

for subject=1:8
    for tests=1:12
        a=['test ',num2str(tests)];
        group3(((subject-1)*12)+tests)={a};
    end
end


for subject=1:8
    for tests=1:12
        total_uh(((subject-1)*12)+tests,1)=real(complex_uh(subject,tests));
        total_uh(((subject-1)*12)+tests,2)=imag(complex_uh(subject,tests));

        total_ut(((subject-1)*12)+tests,1)=real(complex_ut(subject,tests));
        total_ut(((subject-1)*12)+tests,2)=imag(complex_ut(subject,tests));

        total_ch(((subject-1)*12)+tests,1)=real(complex_ch(subject,tests));
        total_ch(((subject-1)*12)+tests,2)=imag(complex_ch(subject,tests));

        total_ct(((subject-1)*12)+tests,1)=real(complex_ct(subject,tests));
        total_ct(((subject-1)*12)+tests,2)=imag(complex_ct(subject,tests));
    end
end

% testnum=1; idark=testnum+[0:12:23 84:12:95]; ilight=testnum+(24:12:71);
```

```
%
idark=[1:10 13:22 85:94];
% ilight=[25:34 37:46 49:58 61:70];
%
figure,plot(total_uh(idark,1),total_uh(idark,2),'*',total_uh(ilight,1),total_uh(ilight,2),'o')

% firsttentests=setxor(1:96,[11 12 23 24 35 36 47 48 59 60 71 72 83 84 95 96])';
firsttentests=(1:96)';
firsttentests=setxor(firsttentests,[76:84])';

%firsttentests isn't just the first 10 tests, it just excludes subject 7
%day 1
[d_uh1,p_uh1,stats_uh1]=manova1(total_uh(firsttentests),group1(firsttentests))
[d_ch1,p_ch1,stats_ch1]=manova1(total_ch(firsttentests),group1(firsttentests))

%
groupprepost([1:10 13:22 77:82 85:94])=1;
groupprepost([11 12 23 24 83 84 95 96])=2;
iprepost=([1:24 77:84 85:96]);
[d_uh1,p_uh1,stats_uh1]=manova1(total_uh(iprepost)',groupprepost(iprepost)')
[d_ch1,p_ch1,stats_ch1]=manova1(total_ch(iprepost)',groupprepost(iprepost)')

%group3
junk=(1:96)'
junk=setxor(junk,[72:75])
[d_ch1,p_ch1,stats_ch1]=manova1(total_uh(junk,:),group2(junk))




[d_uh2,p_uh2,stats_uh2]=manova1(total_uh,group2)
[d_uh3,p_uh3,stats_uh3]=manova1(total_uh,group3)

[d_ut1,p_ut1,stats_ut1]=manova1(total_ut,group1)
[d_ut2,p_ut2,stats_ut2]=manova1(total_ut,group2)
[d_ut3,p_ut3,stats_ut3]=manova1(total_ut,group3)

[d_ch1,p_ch1,stats_ch1]=manova1(total_ch(firsttentests),group1(firsttentests))
[d_ch2,p_ch2,stats_ch2]=manova1(total_ch,group2)
[d_ch3,p_ch3,stats_ch3]=manova1(total_ch,group3)

[d_ct1,p_ct1,stats_ct1]=manova1(total_ct,group1)
[d_ct2,p_ct2,stats_ct2]=manova1(total_ct,group2)
[d_ct3,p_ct3,stats_ct3]=manova1(total_ct,group3)
```

```
[d_uh1,p_uh1,stats_uh1]=manova1(total_uh,group1)
[d_uh2,p_uh2,stats_uh2]=manova1(total_uh,group2)
[d_uh3,p_uh3,stats_uh3]=manova1(total_uh,group3)

[d_ut1,p_ut1,stats_ut1]=manova1(total_ut,group1)
[d_ut2,p_ut2,stats_ut2]=manova1(total_ut,group2)
[d_ut3,p_ut3,stats_ut3]=manova1(total_ut,group3)

[d_ch1,p_ch1,stats_ch1]=manova1(total_ch,group1)
[d_ch2,p_ch2,stats_ch2]=manova1(total_ch,group2)
[d_ch3,p_ch3,stats_ch3]=manova1(total_ch,group3)

[d_ct1,p_ct1,stats_ct1]=manova1(total_ct,group1)
[d_ct2,p_ct2,stats_ct2]=manova1(total_ct,group2)
[d_ct3,p_ct3,stats_ct3]=manova1(total_ct,group3)



%Test 11
clear a b
%Light
a(1,1)=real(complex_uh(3,11));
a(2,1)=real(complex_uh(4,11));
a(3,1)=real(complex_uh(5,11));
a(4,1)=real(complex_uh(6,11));
a(1,2)=imag(complex_uh(3,11));
a(2,2)=imag(complex_uh(4,11));
a(3,2)=imag(complex_uh(5,11));
a(4,2)=imag(complex_uh(6,11));
%Dark
a(5,1)=real(complex_uh(1,11));
a(6,1)=real(complex_uh(2,11));
a(7,1)=real(complex_uh(7,11));
a(8,1)=real(complex_uh(8,11));
a(5,2)=imag(complex_uh(1,11));
a(6,2)=imag(complex_uh(2,11));
a(7,2)=imag(complex_uh(7,11));
a(8,2)=imag(complex_uh(8,11));

b(1:4)={'Light'};
b(5:8)={'Dark'};

[d,p,stats]=manova1(a,b)
```

```
%Test 5
clear a b
%Light
a(1,1)=real(complex_uh(3,6));
a(2,1)=real(complex_uh(4,6));
a(3,1)=real(complex_uh(5,6));
a(4,1)=real(complex_uh(6,6));
a(1,2)=imag(complex_uh(3,6));
a(2,2)=imag(complex_uh(4,6));
a(3,2)=imag(complex_uh(5,6));
a(4,2)=imag(complex_uh(6,6));
%Dark
a(5,1)=real(complex_uh(1,6));
a(6,1)=real(complex_uh(2,6));
a(7,1)=real(complex_uh(7,6));
a(8,1)=real(complex_uh(8,6));
a(5,2)=imag(complex_uh(1,6));
a(6,2)=imag(complex_uh(2,6));
a(7,2)=imag(complex_uh(7,6));
a(8,2)=imag(complex_uh(8,6));

b(1:4)={'Light'};
b(5:8)={'Dark'};

[d,p,stats]=manova1(a,b)
```

```
if q=='Y'
%Graphs
figure('Name','Light vs. Dark Uncorrected Horizontal','NumberTitle','off');
tests=1:12;
subplot(2,1,1)
```

```
errorbar(tests,cf*gain_uh_light,cf*gain_light_se(:,1),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,cf*gain_uh_dark,cf*gain_dark_se(:,1),'k:*')
X1=[4.5 4.5];
Y1=[-12 30];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-12 30];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,25,'Day 1')
hold on
text(6,25,'Day 2')
hold on
text(10.5,25,'Day 3')
legend('Light','Dark','Location',([0.8 0.815 .03 .02]))
legend('boxoff')
axis([0 13 -12 30])
text(-1.8,9,{'H VOR';'  [∫/s]'})
title('Uncorrected Horizontal VOR vs. Test')

for i=1:12
if phase_uh_light(i)<0,phase_uh_light(i)=phase_uh_light(i)+360;end
end
for i=1:12
if phase_uh_dark(i)<0,phase_uh_dark(i)=phase_uh_dark(i)+360;end
end

subplot(2,1,2)
errorbar(tests,phase_uh_light,phase_light_se(:,1),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,phase_uh_dark,phase_dark_se(:,1),'k:*')
hold on
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(-2,150,{'Phase';'  [∫]'})
X1=[4.5 4.5];
Y1=[-100 400];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-100 400];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
title('Uncorrected Horizontal Phase vs. Test')
xlabel('Test')
axis([0 13 0 500])
```

```
figure('Name','Light vs. Dark Uncorrected Torsional','NumberTitle','off');
tests=1:12;
subplot(2,1,1)
errorbar(tests,cf*gain_ut_light,cf*gain_light_se(:,2),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,cf*gain_ut_dark,cf*gain_dark_se(:,2),'k:*')
X1=[4.5 4.5];
Y1=[0 cf];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[0 cf];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,55,'Day 1')
hold on
text(6,55,'Day 2')
hold on
text(10.5,55,'Day 3')
text(-1.8,30,{'T VOR';' [∫/s]'})
legend('Light','Dark','Location',([0.8 0.82 .03 .02]))
legend('boxoff')
axis([0 13 0 cf])
text(13.2,0,'0')
text(13.2,10,'0.16')
text(13.2,20,'0.32')
text(13.2,30,'0.48')
text(13.2,40,'0.64')
text(13.2,50,'0.80')
text(13.2,60,'0.96')
axis([0 13 0 cf])
text(14.3, 37,'Gain', 'rotation',-90)
title('Uncorrected Torsional Amplitude vs. Test')

for i=1:12
if phase_ut_light(i)<0,phase_ut_light(i)=phase_ut_light(i)+360;end
end
for i=1:12
if phase_ut_dark(i)<0,phase_ut_dark(i)=phase_ut_dark(i)+360;end
end

for i=1:12
if phase_ut_light(i)>300,phase_ut_light(i)=phase_ut_light(i)-360;end
end
for i=1:12
if phase_ut_dark(i)>300,phase_ut_dark(i)=phase_ut_dark(i)-360;end
```

```
end

subplot(2,1,2)
errorbar(tests,phase_ut_light,phase_light_se(:,2),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,phase_ut_dark,phase_dark_se(:,2),'k:*')
hold on
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(-2,0,{'Phase';' [/]'})
X1=[4.5 4.5];
Y1=[-25 25];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-25 25];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
axis([0 13 -25 25])
title('Uncorrected Torsional Phase vs. Test')
xlabel('Test')



figure('Name','Light vs. Dark Corrected Horizontal','NumberTitle','off');
tests=1:12;
subplot(2,1,1)
errorbar(tests,cf*gain_ch_light,cf*gain_light_se(:,3),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,cf*gain_ch_dark,cf*gain_dark_se(:,3),'k:*')
X1=[4.5 4.5];
Y1=[-12 30];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-12 30];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,25,'Day 1')
hold on
text(6,25,'Day 2')
hold on
text(10.5,25,'Day 3')
legend('Light','Dark','Location',([0.8 0.815 .03 .02]))
legend('boxoff')
axis([0 13 -12 30])
text(-1.8,9,{'H VOR';' [/s]'})
title('Corrected Horizontal VOR vs. Test')
```

```
for i=1:12
if phase_ch_light(i)<0,phase_ch_light(i)=phase_ch_light(i)+360;end
end
for i=1:12
if phase_ch_dark(i)<0,phase_ch_dark(i)=phase_ch_dark(i)+360;end
end


subplot(2,1,2)
errorbar(tests,phase_ch_light,phase_light_se(:,3),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,phase_ch_dark,phase_dark_se(:,3),'k:*')
hold on
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(-2,150,{'Phase';'  [∫]'})
X1=[4.5 4.5];
Y1=[-100 400];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-100 400];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
title('Corrected Horizontal Phase vs. Test')
xlabel('Test')
axis([0 13 0 500])




figure('Name','Light vs. Dark Corrected Torsional','NumberTitle','off');
tests=1:12;
subplot(2,1,1)
errorbar(tests,cf*gain_ct_light,cf*gain_light_se(:,4),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,cf*gain_ct_dark,cf*gain_dark_se(:,4),'k:*')
X1=[4.5 4.5];
Y1=[0 cf];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[0 cf];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,55,'Day 1')
```

```
hold on
text(6,55,'Day 2')
hold on
text(10.5,55,'Day 3')
text(-1.8,30,{'T VOR';' [∫/s]'})
legend('Light','Dark','Location',([0.8 0.82 .03 .02]))
legend('boxoff')
axis([0 13 0 cf])
text(13.2,0,'0')
text(13.2,10,'0.16')
text(13.2,20,'0.32')
text(13.2,30,'0.48')
text(13.2,40,'0.64')
text(13.2,50,'0.80')
text(13.2,60,'0.96')
axis([0 13 0 cf])
text(14.3, 37,'Gain', 'rotation',-90)
title('Corrected Torsional Amplitude vs. Test')

for i=1:12
if phase_ct_light(i)<0,phase_ct_light(i)=phase_ct_light(i)+360;end
end
for i=1:12
if phase_ct_dark(i)<0,phase_ct_dark(i)=phase_ct_dark(i)+360;end
end

for i=1:12
if phase_ct_light(i)>300,phase_ct_light(i)=phase_ct_light(i)-360;end
end
for i=1:12
if phase_ct_dark(i)>300,phase_ct_dark(i)=phase_ct_dark(i)-360;end
end


subplot(2,1,2)
errorbar(tests,phase_ct_light,phase_light_se(:,4),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,phase_ct_dark,phase_dark_se(:,4),'k:*')
hold on
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(-2,0,{'Phase';' [∫]'})
X1=[4.5 4.5];
Y1=[-25 25];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
```

```
Y2=[-25 25];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
axis([0 13 -25 25])
title('Corrected Torsional Phase vs. Test')
xlabel('Test')




figure('Name','Exp vs. Naive Uncorrected Horizontal','NumberTitle','off');
tests=1:12;
subplot(2,1,1)
errorbar(tests,cf*gain_uh_exp,cf*gain_exp_se(:,1),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,cf*gain_uh_nve,cf*gain_nve_se(:,1),'k:*')
X1=[4.5 4.5];
Y1=[-12 30];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-12 30];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,25,'Day 1')
hold on
text(6,25,'Day 2')
hold on
text(10.5,25,'Day 3')
legend('Exp','Nve','Location',([0.8 0.815 .03 .02]))
legend('boxoff')
axis([0 13 -12 30])
text(-1.8,9,{'H VOR';'  [∫/s]'})
title('Uncorrected Horizontal VOR vs. Test')

for i=1:12
if phase_uh_exp(i)<0,phase_uh_exp(i)=phase_uh_exp(i)+360;end
end
for i=1:12
if phase_uh_nve(i)<0,phase_uh_nve(i)=phase_uh_nve(i)+360;end
end


subplot(2,1,2)
errorbar(tests,phase_uh_exp,phase_exp_se(:,1),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,phase_uh_nve,phase_nve_se(:,1),'k:*')
hold on
```

```
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(-2,150,{'Phase';' [ʃ]'})
X1=[4.5 4.5];
Y1=[-100 400];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-100 400];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
title('Uncorrected Horizontal Phase vs. Test')
xlabel('Test')
axis([0 13 -100 500])
```

```
figure('Name','Exp vs. Naive Uncorrected Torsional','NumberTitle','off');
tests=1:12;
subplot(2,1,1)
errorbar(tests,cf*gain_ut_exp,cf*gain_exp_se(:,2),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,cf*gain_ut_nve,cf*gain_nve_se(:,2),'k:*')
X1=[4.5 4.5];
Y1=[0 cf];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[0 cf];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,55,'Day 1')
hold on
text(6,55,'Day 2')
hold on
text(10.5,55,'Day 3')
text(-1.8,30,{'T VOR';' [ʃ/s]'})
legend('Exp','Nve','Location',([0.8 0.83 .03 .02]))
legend('boxoff')
axis([0 13 0 cf])
text(13.2,0,'0')
text(13.2,10,'0.16')
text(13.2,20,'0.32')
text(13.2,30,'0.48')
text(13.2,40,'0.64')
```

```
text(13.2,50,'0.80')
text(13.2,60,'0.96')
axis([0 13 0 cf])
text(14.3, 37,'Gain', 'rotation',-90)
title('Uncorrected Torsional Amplitude vs. Test')

for i=1:12
if phase_ut_exp(i)<0,phase_ut_exp(i)=phase_ut_exp(i)+360;end
end
for i=1:12
if phase_ut_exp(i)>300,phase_ut_exp(i)=phase_ut_exp(i)-360;end
end
for i=1:12
if phase_ut_nve(i)<0,phase_ut_nve(i)=phase_ut_nve(i)+360;end
end
for i=1:12
if phase_ut_nve(i)>300,phase_ut_nve(i)=phase_ut_nve(i)-360;end
end


subplot(2,1,2)
errorbar(tests,phase_ut_exp,phase_exp_se(:,2),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,phase_ut_nve,phase_nve_se(:,2),'k:*')
hold on
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(-2,0,{'Phase';'  [∫]'})
X1=[4.5 4.5];
Y1=[-25 25];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-25 25];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
axis([0 13 -25 25])
title('Uncorrected Torsional Phase vs. Test')
xlabel('Test')




figure('Name','Exp vs. Naive Corrected Horizontal','NumberTitle','off');
```

```
tests=1:12;
subplot(2,1,1)
errorbar(tests,cf*gain_ch_exp,cf*gain_exp_se(:,3),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,cf*gain_ch_nve,cf*gain_nve_se(:,3),'k:*')
X1=[4.5 4.5];
Y1=[-12 30];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-12 30];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,25,'Day 1')
hold on
text(6,25,'Day 2')
hold on
text(10.5,25,'Day 3')
legend('Exp','Nve','Location',([0.8 0.823 .03 .02]))
legend('boxoff')
axis([0 13 -12 30])
text(-1.8,9,{'H VOR';' [∫/s]'})
title('Corrected Horizontal VOR vs. Test')

for i=1:12
if phase_ch_exp(i)<0,phase_ch_exp(i)=phase_ch_exp(i)+360;end
end

for i=1:12
if phase_ch_nve(i)<50,phase_ch_nve(i)=phase_ch_nve(i)+360;end
end


subplot(2,1,2)
errorbar(tests,phase_ch_exp,phase_exp_se(:,3),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,phase_ch_nve,phase_nve_se(:,3),'k:*')
hold on
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(-2,150,{'Phase';' [∫]'})
X1=[4.5 4.5];
Y1=[-100 400];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-100 400];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
title('Corrected Horizontal Phase vs. Test')
```

```
xlabel('Test')
axis([0 13 -100 500])



figure('Name','Exp vs. Naive Corrected Torsional','NumberTitle','off');
tests=1:12;
subplot(2,1,1)
errorbar(tests,cf*gain_ct_exp,cf*gain_exp_se(:,4),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,cf*gain_ct_nve,cf*gain_nve_se(:,4),'k:*')
X1=[4.5 4.5];
Y1=[0 cf];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[0 cf];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,55,'Day 1')
hold on
text(6,55,'Day 2')
hold on
text(10.5,55,'Day 3')
text(-1.8,30,{'T VOR';'  [∫/s]'})
legend('Exp','Nve','Location',([0.8 0.83 .03 .02]))
legend('boxoff')
axis([0 13 0 cf])
text(13.2,0,'0')
text(13.2,10,'0.16')
text(13.2,20,'0.32')
text(13.2,30,'0.48')
text(13.2,40,'0.64')
text(13.2,50,'0.80')
text(13.2,60,'0.96')
axis([0 13 0 cf])
text(14.3, 37,'Gain', 'rotation',-90)
title('Corrected Torsional Amplitude vs. Test')

for i=1:12
if phase_ct_exp(i)<0,phase_ct_exp(i)=phase_ct_exp(i)+360;end
end
for i=1:12
if phase_ct_exp(i)>300,phase_ct_exp(i)=phase_ct_exp(i)-360;end
end

for i=1:12
```

```
if phase_ct_nve(i)<0,phase_ct_nve(i)=phase_ct_nve(i)+360;end
end
for i=1:12
if phase_ct_nve(i)>300,phase_ct_nve(i)=phase_ct_nve(i)-360;end
end


subplot(2,1,2)
errorbar(tests,phase_ct_exp,phase_exp_se(:,4),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,phase_ct_nve,phase_nve_se(:,4),'k:*')
hold on
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(-2,0,{'Phase';'  [∫]'})
X1=[4.5 4.5];
Y1=[-25 25];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-25 25];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
axis([0 13 -25 25])
title('Corrected Torsional Phase vs. Test')
xlabel('Test')



figure('Name','Light vs. Dark Uncorrected Horizontal Day','NumberTitle','off');
days=1:3;
subplot(2,1,1)
errorbar(days,cf*gain_light_day(:,1),cf*gain_light_day_se(:,1),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,cf*gain_dark_day(:,1),cf*gain_dark_day_se(:,1),'k:*')
text(0.1,7.5,{'H VOR';'  [∫/s]'})
legend('Light','Dark','Location',([0.8 0.83 .03 .02]))
legend('boxoff')
title('Uncorrected Horizontal Amplitude vs. Day')
axis([0.5 3.5 0 15])

for i=1:3
if phase_light_day(i,1)<0,phase_light_day(i,1)=phase_light_day(i,1)+360;end
end
for i=1:3
if phase_light_day(i,1)>300,phase_light_day(i,1)=phase_light_day(i,1)-360;end
```

```
end

for i=1:3
if phase_dark_day(i,1)<0,phase_dark_day(i,1)=phase_dark_day(i,1)+360;end
end
for i=1:3
if phase_dark_day(i,1)>300,phase_dark_day(i,1)=phase_dark_day(i,1)-360;end
end


subplot(2,1,2)
errorbar(days,phase_light_day(:,1),phase_light_day_se(:,1),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_dark_day(:,1),phase_dark_day_se(:,1),'k:*')
title('Uncorrected Horizontal Phase vs. Day')
xlabel('Days')
text(0.1,250,{'Phase';'  [∫/s]'})
axis([0.5 3.5 150 350])

%%%%%%%%

figure('Name','Light vs. Dark Uncorrected Torsional Day','NumberTitle','off');
days=1:3;
subplot(2,1,1)
errorbar(days,cf*gain_light_day(:,2),cf*gain_light_day_se(:,2),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,cf*gain_dark_day(:,2),cf*gain_dark_day_se(:,2),'k:*')
text(0.1,25,{'T VOR';'  [∫/s]'})
legend('Light','Dark','Location',([0.81 0.78 .03 .02]))
legend('boxoff')
title('Uncorrected Torsional Amplitude vs. Day')
text(3.53,10,'0.16')
text(3.53,15,'0.24')
text(3.53,20,'0.32')
text(3.53,25,'0.40')
text(3.53,30,'0.48')
text(3.53,35,'0.56')
text(3.53,40,'0.62')
text(3.8,29,'Gain', 'rotation',-90)
axis([0.5 3.5 10 40])

for i=1:3
if phase_light_day(i,2)<0,phase_light_day(i,2)=phase_light_day(i,2)+360;end
end
for i=1:3
if phase_light_day(i,2)>300,phase_light_day(i,2)=phase_light_day(i,2)-360;end
```

```
end

for i=1:3
if phase_dark_day(i,2)<0,phase_dark_day(i,2)=phase_dark_day(i,2)+360;end
end
for i=1:3
if phase_dark_day(i,2)>300,phase_dark_day(i,2)=phase_dark_day(i,2)-360;end
end


subplot(2,1,2)
errorbar(days,phase_light_day(:,2),phase_light_day_se(:,2),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_dark_day(:,2),phase_dark_day_se(:,2),'k:*')
title('Uncorrected Torsional Phase vs. Day')
xlabel('Days')
text(0.1,2.5,{'Phase';'  [∫/s]'})
axis([0.5 3.5 -5 10])


%%%%%%

figure('Name','Light vs. Dark Corrected Horizontal Day','NumberTitle','off');
days=1:3;
subplot(2,1,1)
errorbar(days,cf*gain_light_day(:,3),cf*gain_light_day_se(:,3),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,cf*gain_dark_day(:,3),cf*gain_dark_day_se(:,3),'k:*')
text(0.1,7.5,{'H VOR';'  [∫/s]'})
legend('Light','Dark','Location',([0.8 0.83 .03 .02]))
legend('boxoff')
title('Corrected Horizontal Amplitude vs. Day')
axis([0.5 3.5 0 15])

for i=1:3
if phase_light_day(i,3)<0,phase_light_day(i,3)=phase_light_day(i,3)+360;end
end
for i=1:3
if phase_light_day(i,3)>300,phase_light_day(i,3)=phase_light_day(i,3)-360;end
end

for i=1:3
if phase_dark_day(i,3)<0,phase_dark_day(i,3)=phase_dark_day(i,3)+360;end
end
for i=1:3
if phase_dark_day(i,3)>300,phase_dark_day(i,3)=phase_dark_day(i,3)-360;end
```

```
end

subplot(2,1,2)
errorbar(days,phase_light_day(:,3),phase_light_day_se(:,3),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_dark_day(:,3),phase_dark_day_se(:,3),'k:*')
title('Corrected Horizontal Phase vs. Day')
xlabel('Days')
text(0.1,250,{'Phase';' [∫/s]'})
axis([0.5 3.5 150 350])

%%%%%%%

figure('Name','Light vs. Dark Corrected Torsional Day','NumberTitle','off');
days=1:3;
subplot(2,1,1)
errorbar(days,cf*gain_light_day(:,4),cf*gain_light_day_se(:,4),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,cf*gain_dark_day(:,4),cf*gain_dark_day_se(:,4),'k:*')
text(0.1,25,{'T VOR';' [∫/s]'})
legend('Light','Dark','Location',([0.8 0.76 .03 .02]))
legend('boxoff')
title('Corrected Torsional Amplitude vs. Day')
text(3.53,10,'0.16')
text(3.53,15,'0.24')
text(3.53,20,'0.32')
text(3.53,25,'0.40')
text(3.53,30,'0.48')
text(3.53,35,'0.56')
text(3.53,40,'0.62')
text(3.8,29,'Gain', 'rotation',-90)
axis([0.5 3.5 10 40])

for i=1:3
if phase_light_day(i,4)<0,phase_light_day(i,4)=phase_light_day(i,4)+360;end
end
for i=1:3
if phase_light_day(i,4)>300,phase_light_day(i,4)=phase_light_day(i,4)-360;end
end

for i=1:3
if phase_dark_day(i,4)<0,phase_dark_day(i,4)=phase_dark_day(i,4)+360;end
end
for i=1:3
if phase_dark_day(i,4)>300,phase_dark_day(i,4)=phase_dark_day(i,4)-360;end
```

```
end


subplot(2,1,2)
errorbar(days,phase_light_day(:,4),phase_light_day_se(:,4),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_dark_day(:,4),phase_dark_day_se(:,4),'k:*')
title('Corrected Torsional Phase vs. Day')
xlabel('Days')
text(0.1,2.5,{'Phase';' [∫/s]'})
axis([0.5 3.5 -5 10])

%%%%%%%

figure('Name','Experienced vs. Naive Uncorrected Horizontal Day','NumberTitle','off');
days=1:3;
subplot(2,1,1)
errorbar(days,cf*gain_exp_day(:,1),cf*gain_exp_day_se(:,1),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,cf*gain_nve_day(:,1),cf*gain_nve_day_se(:,1),'k:*')
text(0.1,7.5,{'H VOR';' [∫/s]'})
legend('Exp','Naive','Location',([0.8 0.83 .03 .02]))
legend('boxoff')
title('Uncorrected Horizontal Amplitude vs. Day')
axis([0.5 3.5 0 15])

for i=1:3
if phase_exp_day(i,1)<0,phase_exp_day(i,1)=phase_exp_day(i,1)+360;end
end
for i=1:3
if phase_exp_day(i,1)>300,phase_exp_day(i,1)=phase_exp_day(i,1)-360;end
end

for i=1:3
if phase_nve_day(i,1)<0,phase_nve_day(i,1)=phase_nve_day(i,1)+360;end
end
for i=1:3
if phase_nve_day(i,1)>300,phase_nve_day(i,1)=phase_nve_day(i,1)-360;end
end


subplot(2,1,2)
errorbar(days,phase_exp_day(:,1),phase_exp_day_se(:,1),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_nve_day(:,1),phase_nve_day_se(:,1),'k:*')
title('Uncorrected Horizontal Phase vs. Day')
```

```
xlabel('Days')
text(0.1,250,{'Phase';'  [∫/s]'})
axis([0.5 3.5 150 350])

%%%%%%%%

figure('Name','Experienced vs. Naive Uncorrected Torsional Day','NumberTitle','off');
days=1:3;
subplot(2,1,1)
errorbar(days,cf*gain_exp_day(:,2),cf*gain_exp_day_se(:,2),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,cf*gain_nve_day(:,2),cf*gain_nve_day_se(:,2),'k:*')
text(0.1,25,{'T VOR';'  [∫/s]'})
legend('Exp','Naive','Location',([0.8 0.67 .03 .02]))
legend('boxoff')
title('Uncorrected Torsional Amplitude vs. Day')
text(3.53,10,'0.16')
text(3.53,15,'0.24')
text(3.53,20,'0.32')
text(3.53,25,'0.40')
text(3.53,30,'0.48')
text(3.53,35,'0.56')
text(3.53,40,'0.62')
text(3.8,29,'Gain', 'rotation',-90)
axis([0.5 3.5 10 40])

for i=1:3
if phase_exp_day(i,2)<0,phase_exp_day(i,2)=phase_exp_day(i,2)+360;end
end
for i=1:3
if phase_exp_day(i,2)>300,phase_exp_day(i,2)=phase_exp_day(i,2)-360;end
end

for i=1:3
if phase_nve_day(i,2)<0,phase_nve_day(i,2)=phase_nve_day(i,2)+360;end
end
for i=1:3
if phase_nve_day(i,2)>300,phase_nve_day(i,2)=phase_nve_day(i,2)-360;end
end


subplot(2,1,2)
errorbar(days,phase_exp_day(:,2),phase_exp_day_se(:,2),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_nve_day(:,2),phase_nve_day_se(:,2),'k:*')
title('Uncorrected Torsional Phase vs. Day')
```

```
xlabel('Days')
text(0.1,2.5,{'Phase';'  [∫/s]'})
axis([0.5 3.5 -5 10])

%%%%%%

figure('Name','Experienced vs. Naive Corrected Horizontal Day','NumberTitle','off');
days=1:3;
subplot(2,1,1)
errorbar(days,cf*gain_exp_day(:,3),cf*gain_exp_day_se(:,3),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,cf*gain_nve_day(:,3),cf*gain_nve_day_se(:,3),'k:*')
text(0.1,7.5,{'H VOR';'  [∫/s]'})
legend('Exp','Naive','Location',([0.8 0.83 .03 .02]))
legend('boxoff')
title('Corrected Horizontal Amplitude vs. Day')
axis([0.5 3.5 0 15])

for i=1:3
if phase_exp_day(i,3)<0,phase_exp_day(i,3)=phase_exp_day(i,3)+360;end
end
for i=1:3
if phase_exp_day(i,3)>300,phase_exp_day(i,3)=phase_exp_day(i,3)-360;end
end

for i=1:3
if phase_nve_day(i,3)<0,phase_nve_day(i,3)=phase_nve_day(i,3)+360;end
end
for i=1:3
if phase_nve_day(i,3)>300,phase_nve_day(i,3)=phase_nve_day(i,3)-360;end
end


subplot(2,1,2)
errorbar(days,phase_exp_day(:,3),phase_exp_day_se(:,3),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_nve_day(:,3),phase_nve_day_se(:,3),'k:*')
title('Corrected Horizontal Phase vs. Day')
xlabel('Days')
text(0.1,250,{'Phase';'  [∫/s]'})
axis([0.5 3.5 150 350])

%%%%%%

figure('Name','Experienced vs. Naive Corrected Torsional Day','NumberTitle','off');
days=1:3;
```

```
subplot(2,1,1)
errorbar(days,cf*gain_exp_day(:,4),cf*gain_exp_day_se(:,4),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,cf*gain_nve_day(:,4),cf*gain_nve_day_se(:,4),'k:*')
text(0.1,25,{'T VOR';' [∫/s]'})
legend('Exp','Naive','Location',([0.8 0.65 .03 .02]))
legend('boxoff')
title('Corrected Torsional Amplitude vs. Day')
text(3.53,10,'0.16')
text(3.53,15,'0.24')
text(3.53,20,'0.32')
text(3.53,25,'0.40')
text(3.53,30,'0.48')
text(3.53,35,'0.56')
text(3.53,40,'0.62')
text(3.8,29,'Gain', 'rotation',-90)
axis([0.5 3.5 10 40])

for i=1:3
if phase_exp_day(i,4)<0,phase_exp_day(i,4)=phase_exp_day(i,4)+360;end
end
for i=1:3
if phase_exp_day(i,4)>300,phase_exp_day(i,4)=phase_exp_day(i,4)-360;end
end

for i=1:3
if phase_nve_day(i,4)<0,phase_nve_day(i,4)=phase_nve_day(i,4)+360;end
end
for i=1:3
if phase_nve_day(i,4)>300,phase_nve_day(i,4)=phase_nve_day(i,4)-360;end
end


subplot(2,1,2)
errorbar(days,phase_exp_day(:,4),phase_exp_day_se(:,4),'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_nve_day(:,4),phase_nve_day_se(:,4),'k:*')
title('Corrected Torsional Phase vs. Day')
xlabel('Days')
text(0.1,2.5,{'Phase';' [∫/s]'})
axis([0.5 3.5 -5 10])

tests=1:12;
subplot(4,1,1)
plot(tests,(abs(gain_uh_light-gain_uh_dark))*cf)
axis([0 13 0 60])
```

```
set(gca,'XTickMode','manual')
set(gca,'XTick',[])
set(gca,'YTickMode','manual')
set(gca,'YTick',[0 60])
ylabel('Difference')
xlabel('Tests')
title('UH Difference [∫/s] vs. Test')
subplot(4,1,2)
plot(tests,(abs(gain_ut_light-gain_ut_dark))*cf)
axis([0 13 0 60])
set(gca,'XTickMode','manual')
set(gca,'XTick',[])
set(gca,'YTickMode','manual')
set(gca,'YTick',[0 60])
ylabel('UH Differences [∫/s]')
xlabel('Tests')
ylabel('Difference')
title('UT Difference [∫/s] vs. Test')
subplot(4,1,3)
plot(tests,(abs(gain_ch_light-gain_ch_dark))*cf)
axis([0 13 0 60])
set(gca,'XTickMode','manual')
set(gca,'XTick',[])
set(gca,'YTickMode','manual')
set(gca,'YTick',[0 60])
ylabel('UH Differences [∫/s]')
xlabel('Tests')
ylabel('Difference')
title('CH Difference [∫/s] vs. Test')
subplot(4,1,4)
plot(tests,(abs(gain_ct_light-gain_ct_dark))*cf)
axis([0 13 0 60])
set(gca,'XTickMode','manual')
set(gca,'XTick',[])
set(gca,'YTickMode','manual')
set(gca,'YTick',[0 60])
ylabel('UH Differences [∫/s]')
xlabel('Tests')
ylabel('Difference')
title('CT Difference [∫/s] vs. Test')

figure('Name','Regression Lines','NumberTitle','off');
tests=1:12;
errorbar(tests,cf*gain_ut_dark,cf*gain_dark_se(:,2),'k:*')
hold on
x=1:4;
```

```
plot(x,(-5.5646*x)+42.9009)
hold on
x=4:5;
plot(x,(15.6400*x)-40.5584)
hold on
x=5:8;
plot(x,(-0.5056*x)+40.1457)
hold on
x=8:9;
plot(x,(0.4997*x)+31.2956)
hold on
x=9:12;
plot(x,(-1.9192*x)+53.6474)
ylabel('Uncorrected Torsional Amplitude')
xlabel('Tests')
legend('Actual Dark','Fitted Dark')
legend('boxoff')
axis([0 13 0 60])

figure('Name','Beginning vs. End Trends','NumberTitle','off');
tests=[1 12];

subplot(2,2,1)
plot(tests,gain_uh_light(tests)*cf,'--','Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_uh_dark(tests)*cf,'k-.')
hold on
plot(tests,gain_uh_exp(tests)*cf,'Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_uh_nve(tests)*cf,'k:')
legend('Light','Dark','Exp','Nve');legend('boxoff')
axis([0 13 0 15])
set(gca,'XTickMode','manual')
set(gca,'XTick',[1 12])
text
title('Unorrected Horizontal')
ylabel('Amplitude')

subplot(2,2,2)
plot(tests,gain_ch_light(tests)*cf,'--','Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_ch_dark(tests)*cf,'k-.')
hold on
plot(tests,gain_ch_exp(tests)*cf,'Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_ch_nve(tests)*cf,'k:')
```

```
axis([0 13 0 15])
set(gca,'XTickMode','manual')
set(gca,'XTick',[1 12])
title('Corrected Horizontal')
ylabel('Amplitude')

subplot(2,2,3)

plot(tests,gain_ut_light(tests)*cf,'--','Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_ut_dark(tests)*cf,'k-.')
hold on
plot(tests,gain_ut_exp(tests)*cf,'Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_ut_nve(tests)*cf,'k:')
axis([0 13 0 40])
set(gca,'XTickMode','manual')
set(gca,'XTick',[1 12])
title('Uncorrected Torsional')
ylabel('Amplitude')

subplot(2,2,4)
plot(tests,gain_ct_light(tests)*cf,'--','Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_ct_dark(tests)*cf,'k-.')
hold on
plot(tests,gain_ct_exp(tests)*cf,'Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_ct_nve(tests)*cf,'k:')
axis([0 13 0 40])
set(gca,'XTickMode','manual')
set(gca,'XTick',[1 12])
title('Corrected Torsional')
ylabel('Amplitude')

end
```

Somatosensory Bar Plots

```
clear all

load /Volumes/Untitled/Motion_Experience/SSBar_Excel/SSBar3.mat

cd /Volumes/Untitled/Analysis_Programs/

q=input('Show graphs? [Y/N]:','s');

%RPO=subject 1, AAR=subject 2, DMM=subject 3, FK=subject 4, MM=SUbject 5,
%JK=subject 6 SZ=subject 7, RCT=subject 8

%y_complex_all(subject,days,tests,movements,:1 or 2)
%1 equals bar, 2=device


%Calculates the non-nan length of each MOVEMENT, e.g. calculates the number
%of used cycles in the MOVEMENT
for subject=1:8
   for days=1:3
      for tests=1:4
         for movements=1:2
            new_length(subject,days,tests,movements)=10-
length(find(isnan(y_complex_all(subject,days,tests,movements,:,1))));
         end
      end
   end
end

%Calculates the average complex value for each MOVEMENT
for subject=1:8
   for days=1:3
      for tests=1:4
         for movements=1:2
            average_bar(subject,((days-1)*8)+((tests-
1)*2)+movements)=nansum(squeeze(y_complex_all(subject,days,tests,movements,:,1)))/
new_length(subject,days,tests,movements);
            average_device(subject,((days-1)*8)+((tests-
1)*2)+movements)=nansum(squeeze(y_complex_all(subject,days,tests,movements,:,2)))/
new_length(subject,days,tests,movements);
         end
      end
   end
end
```

```
%Andrew,Jen,Suzanne
for subject=1:8
   for movements=1:2:23


complex_bar(subject,round(movements/2))=((average_bar(subject,movements)/average_
device(subject,movements))+(average_bar(subject,movements+1)/average_device(subjec
t,movements+1)))/2;

   end
end


complex_bar(4,6)=average_bar(4,11)/average_device(4,11);
complex_bar(4,12)=average_bar(4,24)/average_device(4,24);
complex_bar(5,9)=average_bar(5,17)/average_device(5,17);
complex_bar(7,6)=average_bar(7,11)/average_device(7,11);



%Calculates light/dark, exp/naive gain, phase, amp for each TEST
for tests=1:12

gain_bar_light(tests)=abs((complex_bar(3,tests)+complex_bar(4,tests)+complex_bar(5,te
sts)+complex_bar(6,tests))/4);

phase_bar_light(tests)=180/pi*angle((complex_bar(3,tests)+complex_bar(4,tests)+compl
ex_bar(5,tests)+complex_bar(6,tests))/4);


gain_bar_exp(tests)=abs((complex_bar(3,tests)+complex_bar(4,tests)+complex_bar(1,tes
ts)+complex_bar(2,tests))/4);

phase_bar_exp(tests)=180/pi*angle((complex_bar(3,tests)+complex_bar(4,tests)+comple
x_bar(1,tests)+complex_bar(2,tests))/4);

   if tests==12

gain_bar_dark(tests)=abs((complex_bar(1,tests)+complex_bar(2,tests)+complex_bar(8,te
sts))/3);

phase_bar_dark(tests)=180/pi*angle((complex_bar(1,tests)+complex_bar(2,tests)+compl
ex_bar(8,tests))/3);
```

```
gain_bar_nve(tests)=abs((complex_bar(8,tests)+complex_bar(5,tests)+complex_bar(6,tes
ts))/3);

phase_bar_nve(tests)=180/pi*angle((complex_bar(8,tests)+complex_bar(5,tests)+comple
x_bar(6,tests))/3);
  else

gain_bar_dark(tests)=abs((complex_bar(1,tests)+complex_bar(2,tests)+complex_bar(7,te
sts)+complex_bar(8,tests))/4);

phase_bar_dark(tests)=180/pi*angle((complex_bar(1,tests)+complex_bar(2,tests)+compl
ex_bar(7,tests)+complex_bar(8,tests))/4);


gain_bar_nve(tests)=abs((complex_bar(7,tests)+complex_bar(8,tests)+complex_bar(5,tes
ts)+complex_bar(6,tests))/4);

phase_bar_nve(tests)=180/pi*angle((complex_bar(7,tests)+complex_bar(8,tests)+comple
x_bar(5,tests)+complex_bar(6,tests))/4);
  end
end

%Light Standard Error
for tests=1:12


A_real=[real(complex_bar(3,tests)),real(complex_bar(4,tests)),real(complex_bar(5,tests))
,real(complex_bar(6,tests))];

A_imag=[imag(complex_bar(3,tests)),imag(complex_bar(4,tests)),imag(complex_bar(5,t
ests)),imag(complex_bar(6,tests))];

     A_complex = complex(A_real, A_imag);
     A_mean = mean(A_complex, 2);
     A_amp = abs(A_mean);



     A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
     cd /Volumes/Untitled/Analysis_Programs/
     display = 0;    % 1:display, 0:no
     e = [];         % e = []: standard error, e = 1: standard deviation
     A_std = f_covellips_wg(A_complex,e,display);

     gain_light_se(tests) = A_std(1);
```

```
    phase_light_se(tests) = A_std(2) * 180/pi;
end

%Dark Standard Error
for tests=1:12
    if tests==12

A_real=[real(complex_bar(1,tests)),real(complex_bar(2,tests)),real(complex_bar(8,tests))
];

A_imag=[imag(complex_bar(1,tests)),imag(complex_bar(2,tests)),imag(complex_bar(8,t
ests))];
    else

A_real=[real(complex_bar(1,tests)),real(complex_bar(2,tests)),real(complex_bar(7,tests))
,real(complex_bar(8,tests))];

A_imag=[imag(complex_bar(1,tests)),imag(complex_bar(2,tests)),imag(complex_bar(7,t
ests)),imag(complex_bar(8,tests))];
    end
    A_complex = complex(A_real, A_imag);
    A_mean = mean(A_complex, 2);
    A_amp = abs(A_mean);



    A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
    cd /Volumes/Untitled/Analysis_Programs/
    display = 0;   % 1:display, 0:no
    e = [];        % e = []: standard error, e = 1: standard deviation
    A_std = f_covellips_wg(A_complex,e,display);

    gain_dark_se(tests) = A_std(1);
    phase_dark_se(tests) = A_std(2) * 180/pi;

end

%Exp Standard Error
for tests=1:12


A_real=[real(complex_bar(3,tests)),real(complex_bar(4,tests)),real(complex_bar(1,tests))
,real(complex_bar(2,tests))];

A_imag=[imag(complex_bar(3,tests)),imag(complex_bar(4,tests)),imag(complex_bar(1,t
ests)),imag(complex_bar(2,tests))];
```

```
    A_complex = complex(A_real, A_imag);
    A_mean = mean(A_complex, 2);
    A_amp = abs(A_mean);



    A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
    cd /Volumes/Untitled/Analysis_Programs/
    display = 0;   % 1:display, 0:no
    e = [];        % e = []: standard error, e = 1: standard deviation
    A_std = f_covellips_wg(A_complex,e,display);

    gain_exp_se(tests) = A_std(1);
    phase_exp_se(tests)  = A_std(2) * 180/pi;

end

%Naive Standard Error
for tests=1:12
    if tests==12

A_real=[real(complex_bar(8,tests)),real(complex_bar(5,tests)),real(complex_bar(6,tests))
];

A_imag=[imag(complex_bar(8,tests)),imag(complex_bar(5,tests)),imag(complex_bar(6,t
ests))];
    else

A_real=[real(complex_bar(7,tests)),real(complex_bar(8,tests)),real(complex_bar(5,tests))
,real(complex_bar(6,tests))];

A_imag=[imag(complex_bar(7,tests)),imag(complex_bar(8,tests)),imag(complex_bar(5,t
ests)),imag(complex_bar(6,tests))];
    end
    A_complex = complex(A_real, A_imag);
    A_mean = mean(A_complex, 2);
    A_amp = abs(A_mean);

    fprintf('%d %f\n',tests,A_amp-gain_bar_nve(tests))

    A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
    cd /Volumes/Untitled/Analysis_Programs/
    display = 0;   % 1:display, 0:no
    e = [];        % e = []: standard error, e = 1: standard deviation
    A_std = f_covellips_wg(A_complex,e,display);
```

```
    gain_nve_se(tests) = A_std(1);
    phase_nve_se(tests)  = A_std(2) * 180/pi;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for days=1:3
  for tests=1:4

    one(tests,days)=complex_bar(1,((days-1)*4)+tests);
    two(tests,days)=complex_bar(2,((days-1)*4)+tests);
    three(tests,days)=complex_bar(3,((days-1)*4)+tests);
    four(tests,days)=complex_bar(4,((days-1)*4)+tests);
    five(tests,days)=complex_bar(5,((days-1)*4)+tests);
    six(tests,days)=complex_bar(6,((days-1)*4)+tests);
    seven(tests,days)=complex_bar(7,((days-1)*4)+tests);
    eight(tests,days)=complex_bar(8,((days-1)*4)+tests);

  end
end



for days=1:3

gain_light_day(days)=abs((nansum(three(:,days))+nansum(four(:,days))+nansum(five(:,d
ays))+nansum(six(:,days)))/16);

phase_light_day(days)=180/pi*angle((nansum(three(:,days))+nansum(four(:,days))+nans
um(five(:,days))+nansum(six(:,days)))/16);


gain_exp_day(days)=abs((nansum(three(:,days))+nansum(four(:,days))+nansum(one(:,da
ys))+nansum(two(:,days)))/16);

phase_exp_day(days)=180/pi*angle((nansum(three(:,days))+nansum(four(:,days))+nansu
m(one(:,days))+nansum(two(:,days)))/16);

  if days==3

gain_dark_day(days)=abs((nansum(one(:,days))+nansum(two(:,days))+nansum(seven(1:
3,days))+nansum(eight(:,days)))/15);

phase_dark_day(days)=180/pi*angle((nansum(one(:,days))+nansum(two(:,days))+nansu
m(seven(1:3,days))+nansum(eight(:,days)))/15);
```

```
gain_nve_day(days)=abs((nansum(seven(1:3,days))+nansum(eight(:,days))+nansum(five
(:,days))+nansum(six(:,days)))/15);

phase_nve_day(days)=180/pi*angle((nansum(seven(1:3,days))+nansum(eight(:,days))+n
ansum(five(:,days))+nansum(six(:,days)))/15);
  else

gain_dark_day(days)=abs((nansum(one(:,days))+nansum(two(:,days))+nansum(seven(:,d
ays))+nansum(eight(:,days)))/16);

phase_dark_day(days)=180/pi*angle((nansum(one(:,days))+nansum(two(:,days))+nansu
m(seven(:,days))+nansum(eight(:,days)))/16);


gain_nve_day(days)=abs((nansum(seven(:,days))+nansum(eight(:,days))+nansum(five(:,
days))+nansum(six(:,days)))/16);

phase_nve_day(days)=180/pi*angle((nansum(seven(:,days))+nansum(eight(:,days))+nans
um(five(:,days))+nansum(six(:,days)))/32);
  end
end



%Light Standard Error

for days=1:3

  A_real=[real(three(:,days))',real(four(:,days))',real(five(:,days))',real(six(:,days))'];

A_imag=[imag(three(:,days))',imag(four(:,days))',imag(five(:,days))',imag(six(:,days))'];

  A_complex = complex(A_real, A_imag);
  A_mean = mean(A_complex, 2);
  A_amp = abs(A_mean);
  A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
  cd /Volumes/Untitled/Analysis_Programs/
  display = 0;   % 1:display, 0:no
  e = [];        % e = []: standard error, e = 1: standard deviation
  A_std = f_covellips_wg(A_complex,e,display);

  gain_light_day_se(days) = A_std(1);
  phase_light_day_se(days)  = A_std(2) * 180/pi;
```

```
end


%Dark Standard Error

for days=1:3
  if days==3
  A_real=[real(one(:,days))',real(two(:,days))',real(seven(1:3,days))',real(eight(:,days))'];

A_imag=[imag(one(:,days))',imag(two(:,days))',imag(seven(1:3,days))',imag(eight(:,days
))'];
  else
  A_real=[real(one(:,days))',real(two(:,days))',real(seven(:,days))',real(eight(:,days))'];

A_imag=[imag(one(:,days))',imag(two(:,days))',imag(seven(:,days))',imag(eight(:,days))']
;
  end
  A_complex = complex(A_real, A_imag);
  A_mean = mean(A_complex, 2);
  A_amp = abs(A_mean);
  A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
  cd /Volumes/Untitled/Analysis_Programs/
  display = 0;   % 1:display, 0:no
  e = [];        % e = []: standard error, e = 1: standard deviation
  A_std = f_covellips_wg(A_complex,e,display);

  gain_dark_day_se(days) = A_std(1);
  phase_dark_day_se(days)  = A_std(2) * 180/pi;

end


%Exp Standard Error

for days=1:3

  A_real=[real(one(:,days))',real(two(:,days))',real(three(:,days))',real(four(:,days))'];

A_imag=[imag(one(:,days))',imag(two(:,days))',imag(three(:,days))',imag(four(:,days))'];

  A_complex = complex(A_real, A_imag);
  A_mean = mean(A_complex, 2);
  A_amp = abs(A_mean);
  A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
  cd /Volumes/Untitled/Analysis_Programs/
  display = 0;   % 1:display, 0:no
```

```
   e = [];        % e = []: standard error, e = 1: standard deviation
   A_std = f_covellips_wg(A_complex,e,display);

   gain_exp_day_se(days) = A_std(1);
   phase_exp_day_se(days)  = A_std(2) * 180/pi;

end



%Naive Standard Error

for days=1:3
   if days==3
   A_real=[real(five(:,days))',real(six(:,days))',real(seven(1:3,days))',real(eight(:,days))'];

A_imag=[imag(five(:,days))',imag(six(:,days))',imag(seven(1:3,days))',imag(eight(:,days)
)'];
   else
   A_real=[real(five(:,days))',real(six(:,days))',real(seven(:,days))',real(eight(:,days))'];

A_imag=[imag(five(:,days))',imag(six(:,days))',imag(seven(:,days))',imag(eight(:,days))'];
   end
   A_complex = complex(A_real, A_imag);
   A_mean = mean(A_complex, 2);
   A_amp = abs(A_mean);
   A_ph  = 180/pi*atan2(imag(A_mean),real(A_mean));
   cd /Volumes/Untitled/Analysis_Programs/
   display = 0;   % 1:display, 0:no
   e = [];        % e = []: standard error, e = 1: standard deviation
   A_std = f_covellips_wg(A_complex,e,display);

   gain_nve_day_se(days) = A_std(1);
   phase_nve_day_se(days)  = A_std(2) * 180/pi;

end



if q=='Y'
```

```
figure('Name','Light vs. Dark SS Bar','NumberTitle','off');
subplot(2,1,1)
tests=1:12;
errorbar(tests,gain_bar_light,gain_light_se,'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,gain_bar_dark,gain_dark_se,'k:*')
legend('Light','Dark','Location',([0.21 0.82 .03 .02]))
legend('boxoff')
X1=[4.5 4.5];
Y1=[0 2.5];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[0 2.5];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,2.2,'Day 1')
hold on
text(6,2.2,'Day 2')
hold on
text(10.5,2.2,'Day 3')
ylabel('SS Bar Gain')
axis([0 13 0 2.5])
title('SS Bar Gain vs. Test')

for i=1:12
if phase_bar_light(i)<0,phase_bar_light(i)=phase_bar_light(i)+360;end
end
for i=1:12
if phase_bar_dark(i)<0,phase_bar_dark(i)=phase_bar_dark(i)+360;end
end

subplot(2,1,2)
errorbar(tests,phase_bar_light,phase_light_se,'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,phase_bar_dark,phase_dark_se,'k:*')
hold on
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
ylabel('SS Bar Phase')
X1=[4.5 4.5];
Y1=[-10 30];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-10 30];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
xlabel('Tests')
axis([0 13 0 30])
```

```
figure('Name','Exp vs. Nve SS Bar','NumberTitle','off');
subplot(2,1,1)
errorbar(tests,gain_bar_exp,gain_exp_se,'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,gain_bar_nve,gain_nve_se,'k:*')
legend('Exp','Nve','Location',([0.8 0.83 .03 .02]))
legend('boxoff')
X1=[4.5 4.5];
Y1=[0 2.5];
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[0 2.5];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
text(2,2.2,'Day 1')
hold on
text(6,2.2,'Day 2')
hold on
text(10.5,2.2,'Day 3')
ylabel('SS Bar Gain')
axis([0 13 0 2.5])
title('SS Bar Gain vs. Test')

for i=1:12
if phase_bar_exp(i)<0,phase_bar_exp(i)=phase_bar_exp(i)+360;end
end
for i=1:12
if phase_bar_nve(i)<0,phase_bar_nve(i)=phase_bar_nve(i)+360;end
end

subplot(2,1,2)
errorbar(tests,phase_bar_exp,phase_exp_se,'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(tests+0.1,phase_bar_nve,phase_nve_se,'k:*')
hold on
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
ylabel('SS Bar Phase')
X1=[4.5 4.5];
Y1=[-10 30];
```

```
line(X1,Y1,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
X2=[8.5 8.5];
Y2=[-10 30];
line(X2,Y2,'Color',[0.7 0.7 0.7],'LineWidth',1,'LineStyle',':')
xlabel('Tests')
axis([0 13 0 30])




% figure('Name','Gains','NumberTitle','off');
% subplot(2,1,1)
% errorbar(tests,gain_bar_light,gain_light_se,'o-','Color',[0.5 0.5 0.5])
% hold on
% errorbar(tests,gain_bar_dark,gain_dark_se,'k:*')
% legend('Light','Dark')
% legend('boxoff')
%
%
% subplot(2,1,2)
% errorbar(tests,gain_bar_exp,gain_exp_se,'o-','Color',[0.5 0.5 0.5])
% hold on
% errorbar(tests,gain_bar_nve,gain_nve_se,'k:*')
% legend('Exp','Nve')
% legend('boxoff')




figure('Name','Light vs. Dark SS Bar Day','NumberTitle','off');
days=1:3;
subplot(2,1,1)
errorbar(days,gain_light_day,gain_light_day_se,'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,gain_dark_day,gain_dark_day_se,'k:*')
```

```
text(0.1,0.75,{' SS Bar';'Gain [∫/s]'})
legend('Light','Dark','Location',([0.23 0.86 .03 .02]))
legend('boxoff')
title('SS Bar Gain vs. Day')
axis([0.5 3.5 0 2.5])

for i=1:3
if phase_light_day(i)<0,phase_light_day(i)=phase_light_day(i)+360;end
end
for i=1:3
if phase_light_day(i)>300,phase_light_day(i)=phase_light_day(i)-360;end
end

for i=1:3
if phase_dark_day(i)<0,phase_dark_day(i)=phase_dark_day(i)+360;end
end
for i=1:3
if phase_dark_day(i)>300,phase_dark_day(i)=phase_dark_day(i)-360;end
end


subplot(2,1,2)
errorbar(days,phase_light_day,phase_light_day_se,'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_dark_day,phase_dark_day_se,'k:*')
title('SS Bar Phase vs. Day')
xlabel('Days')
text(0.1,20,{'Phase';'  [∫]'})
axis([0.5 3.5 0 30])





figure('Name','Experienced vs. Naive SS Bar Day','NumberTitle','off');
days=1:3;
subplot(2,1,1)
errorbar(days,gain_exp_day,gain_exp_day_se,'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,gain_nve_day,gain_nve_day_se,'k:*')
text(0.1,0.75,{' SS Bar';'Gain [∫/s]'})
legend('Exp','Naive','Location',([0.23 0.83 .03 .02]))
```

```
legend('boxoff')
title('SS Bar Gain vs. Day')
axis([0.5 3.5 0 2.5])


for i=1:3
if phase_exp_day(i)<0,phase_exp_day(i)=phase_exp_day(i)+360;end
end
for i=1:3
if phase_exp_day(i)>300,phase_exp_day(i)=phase_exp_day(i)-360;end
end

for i=1:3
if phase_nve_day(i)<0,phase_nve_day(i)=phase_nve_day(i)+360;end
end
for i=1:3
if phase_nve_day(i)>300,phase_nve_day(i)=phase_nve_day(i)-360;end
end


subplot(2,1,2)
errorbar(days,phase_exp_day,phase_exp_day_se,'o-','Color',[0.5 0.5 0.5])
hold on
errorbar(days,phase_nve_day,phase_nve_day_se,'k:*')
title('SS Bar Phase vs. Day')
xlabel('Days')
text(0.1,20,{'Phase';'   [∫]'})
axis([0.5 3.5 0 30])

%%%%%%%%%%%%%%%%%

figure('Name','Start vs. End Trends','NumberTitle','off');
tests=[1 12];
errorbar(tests,gain_bar_light(tests),gain_light_se(tests))
plot(tests,gain_bar_light(tests),'--','Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_bar_dark(tests),'k-.')
hold on
plot(tests,gain_bar_exp(tests),'Color',[0.5 0.5 0.5])
hold on
plot(tests,gain_bar_nve(tests),'k:')
hold on
axis([0 13 0 1])
legend('Light','Dark','Exp','Nve','Location',('best'));legend('boxoff')
set(gca,'XTickMode','manual')
set(gca,'XTick',[1 12])
```

```
text
xlabel('Tests')
ylabel('Gain')
title('SS Bar vs. Test')




figure('Name','Start vs. End Trends','NumberTitle','off');
tests=[1 12];
errorbar(tests,gain_bar_light(tests),gain_light_se(tests),'--','Color',[0.5 0.5 0.5])
hold on
errorbar(tests,gain_bar_dark(tests),gain_dark_se(tests),'k-.')
hold on
errorbar(tests,gain_bar_exp(tests),gain_exp_se(tests),'Color',[0.5 0.5 0.5])
hold on
errorbar(tests,gain_bar_nve(tests),gain_nve_se(tests),'k:')
hold on
axis([0 13 0 1])
legend('Light','Dark','Exp','Nve','Location',('best'));legend('boxoff')
set(gca,'XTickMode','manual')
set(gca,'XTick',[1 12])
text
xlabel('Tests')
ylabel('Gain')
title('SS Bar vs. Test')
end


%Looks at the individual cycles to identify 'dirty' values

% for subject=7;
%   for days=3;
%     for tests=4;
%       for movements=1:2;
%         figure(subject)
%         subplot(6,4,movements+ ((tests-1)*2) + ((days-1)*8))
%         plot(squeeze(y_complex_all(subject,days,tests,movements,:,1)),'+')
%         hold on
%         plot(average_bar(subject,((days-1)*8)+((tests-1)*2)+movements),'r+')
%       end
%     end
%   end
% end
```

```
%Stats

for subject=1:8
   for tests=1:12
      total_bar(((subject-1)*12)+tests,1)=real(complex_bar(subject,tests));
      total_bar(((subject-1)*12)+tests,2)=imag(complex_bar(subject,tests));
   end
end

%Compares Experienced and Naive, Light vs. Dark
for z=84:95
   total_bar(z,1)=total_bar(z+1,1);
   total_bar(z,2)=total_bar(z+1,2);
end

total_bar_final(1:95,1)=total_bar(1:95,1);
total_bar_final(1:95,2)=total_bar(1:95,2);

grouping1(1:24)={'Dark'};
grouping1(25:72)={'Light'};
grouping1(73:96)={'Dark'};

grouping2(1:48)={'Experienced'};
grouping2(49:96)={'Naive'};

all=[1:96];
erase=[84];

grouping1=grouping1(setxor(all,erase));
grouping2=grouping2(setxor(all,erase));


[d,p,stats]=manova1(total_bar_final,grouping1)
[d,p,stats]=manova1(total_bar_final,grouping2)% Questionable

plot(total_bar_final(1:48),'g+')

junk=abs(complex(total_bar_final(:,1),total_bar_final(:,2)));
junk2=180/pi*angle(complex(total_bar_final(:,1),total_bar_final(:,2)));


%First vs. Last Total
individ_bar=[complex_bar(1,1),complex_bar(2,1),complex_bar(3,1),complex_bar(4,1),co
mplex_bar(5,1),complex_bar(6,1),complex_bar(7,1),complex_bar(8,1),complex_bar(1,12
```

```
),complex_bar(2,12),complex_bar(3,12),complex_bar(4,12),complex_bar(5,12),complex
_bar(6,12),complex_bar(7,12),complex_bar(8,12)];

for z=1:length(individ_bar)
   individual_stats_bar(z,1)=real(individ_bar(z));
   individual_stats_bar(z,2)=imag(individ_bar(z));
end

individual_grouping(1:8)={'first'};
individual_grouping(9:16)={'last'};

[d,p,stats]=manova1(individual_stats_bar,individual_grouping)


%First vs. Last Light
all=[1:16];
erase=[1 2 7 8 9 10 15 16];

individ_bar_demo=individ_bar(setxor(all,erase));

clear z
for z=1:length(individ_bar_demo)
   individ_bar_light(z,1)=real(individ_bar_demo(z));
   individ_bar_light(z,2)=imag(individ_bar_demo(z));
end

individual_light(1:4)={'first'};
individual_light(5:8)={'last'};

[d,p,stats]=manova1(individ_bar_light,individual_light)

%First vs. Last Dark
all=[1:16];
erase=[3 4 5 6 11 12 13 14];

individ_bar_demo=individ_bar(setxor(all,erase));

clear z
for z=1:length(individ_bar_demo)
   individ_bar_dark(z,1)=real(individ_bar_demo(z));
   individ_bar_dark(z,2)=imag(individ_bar_demo(z));
end

individual_dark(1:4)={'first'};
individual_dark(5:8)={'last'};
```

```
[d,p,stats]=manova1(individ_bar_dark,individual_dark)

%First vs. Last Experienced
all=[1:16];
erase=[5 6 7 8 13 14 15 16];

individ_bar_demo=individ_bar(setxor(all,erase));

clear z
for z=1:length(individ_bar_demo)
   individ_bar_exp(z,1)=real(individ_bar_demo(z));
   individ_bar_exp(z,2)=imag(individ_bar_demo(z));
end

individual_exp(1:4)={'first'};
individual_exp(5:8)={'last'};

[d,p,stats]=manova1(individ_bar_exp,individual_exp)

%First vs. Last Naive
all=[1:16];
erase=[1 2 3 4 9 10 11 12];

individ_bar_demo=individ_bar(setxor(all,erase));

clear z
for z=1:length(individ_bar_demo)
   individ_bar_nve(z,1)=real(individ_bar_demo(z));
   individ_bar_nve(z,2)=imag(individ_bar_demo(z));
end

individual_nve(1:4)={'first'};
individual_nve(5:8)={'last'};

[d,p,stats]=manova1(individ_bar_nve,individual_nve)


%Light vs. Dark at start
junk=[complex_bar(3,1),complex_bar(4,1),complex_bar(5,1),complex_bar(6,1),complex
_bar(1,1),complex_bar(2,1),complex_bar(7,1),complex_bar(8,1)];

for q=1:8
   junk_final(q,1)=real(junk(q))
   junk_final(q,2)=imag(junk(q))
end
```

```
junk_grouping(1:4)={'Light'}';
junk_grouping(5:8)={'Dark'}';


[d,p,stats]=manova1(junk_final,junk_grouping)

%Light vs. Dark at end

junk=[complex_bar(3,12),complex_bar(4,12),complex_bar(5,12),complex_bar(6,12),com
plex_bar(1,12),complex_bar(2,12),complex_bar(7,12),complex_bar(8,12)];

for q=1:8
  junk_final(q,1)=real(junk(q))
  junk_final(q,2)=imag(junk(q))
end

junk_grouping(1:4)={'Light'}';
junk_grouping(5:8)={'Dark'}';


[d,p,stats]=manova1(junk_final,junk_grouping)

for subjects=1:8
  for tests=1:12
    new_junk(((subjects-1)*12)+tests) =abs(complex_bar(subjects,tests));
  end
end
```

References

Ahn SC. Short-term vestibular responses to repeated rotations in pilots. *Aviat Space Environ Med* 74: 285-287, 2003.
Anastasio TJ. A pattern correlation model of vestibulo-ocular reflex habituation. *Neural Netw* 14: 1-22, 2001.
Angelaki DE, and Dickman JD. Gravity or translation: central processing of vestibular signals to detect motion or tilt. *J Vestib Res* 13: 245-253, 2003.
Angelaki DE, and Hess BJ. Visually induced adaptation in three-dimensional organization of primate vestibuloocular reflex. *J Neurophysiol* 79: 791-807, 1998.
Angelaki DE, McHenry MQ, Dickman JD, Newlands SD, and Hess BJ. Computation of inertial motion: neural strategies to resolve ambiguous otolith information. *J Neurosci* 19: 316-327, 1999.
Angelaki DE, McHenry MQ, and Hess BJ. Primate translational vestibuloocular reflexes. I. High-frequency dynamics and three-dimensional properties during lateral motion. *J Neurophysiol* 83: 1637-1647, 2000.
Angelaki DE, Wei M, and Merfeld DM. Vestibular discrimination of gravity and translational acceleration. *Ann N Y Acad Sci* 942: 114-127, 2001.
Aron A, and Aron E. *Statistics for Psychology*. Prentice Hall, 1999.
Benson AJ, Spencer MB, and Stott JR. Thresholds for the detection of the direction of whole-body, linear movement in the horizontal plane. *Aviat Space Environ Med* 57: 1088-1096, 1986.
Bringoux L, Marin L, Nougier V, Barraud PA, and Raphel C. Effects of gymnastics expertise on the perception of body orientation in the pitch dimension. *J Vestib Res* 10: 251-258, 2000.
Clement G, Moore ST, Raphan T, and Cohen B. Perception of tilt (somatogravic illusion) in response to sustained linear acceleration during space flight. *Exp Brain Res* 138: 410-418, 2001.
Coats AC, and Smith SY. Body position and the intensity of caloric nystagmus. *Acta Otolaryngol* 63: 515-532, 1967.
Curthoys IS. The role of ocular torsion in visual measures of vestibular function. *Brain Res Bull* 40: 399-403; discussion 403-395, 1996.
Dichgans J, Held R, Young LR, and Brandt T. Moving visual scenes influence the apparent direction of gravity. *Science* 178: 1217-1219, 1972.
Einstein A. Über das Relativitätsprinzip und die aus demselben gezogenen Folgerungen. *Jahrb Radioakt* 4: 411-462, 1908.
Faulstich BM, Onori KA, and du Lac S. Comparison of plasticity and development of mouse optokinetic and vestibulo-ocular reflexes suggests differential gain control mechanisms. *Vision Res* 44: 3419-3427, 2004.
Fernandez C, and Goldberg JM. Physiology of peripheral neurons innervating otolith organs of the squirrel monkey. I. Response to static tilts and to long-duration centrifugal force. *J Neurophysiol* 39: 970-984, 1976a.
Fernandez C, and Goldberg JM. Physiology of peripheral neurons innervating otolith organs of the squirrel monkey. II. Directional selectivity and force-response relations. *J Neurophysiol* 39: 985-995, 1976b.

Fernandez C, and Goldberg JM. Physiology of peripheral neurons innervating otolith organs of the squirrel monkey. III. Response dynamics. *J Neurophysiol* 39: 996-1008, 1976c.
Graybiel A, and Brown R. The delay in visual reorientation following exposure to a change in direction of resultant force on a human centrifuge. *J Gen Psychol* 45: 143-150, 1951.
Green AM, Shaikh AG, and Angelaki DE. Sensory vestibular contributions to constructing internal models of self-motion. *J Neural Eng* 2: S164-179, 2005.
Haslwanter T, Jaeger R, Mayr S, and Fetter M. Three-dimensional eye-movement responses to off-vertical axis rotations in humans. *Exp Brain Res* 134: 96-106, 2000.
Johnson RA, and Wichern DW. *Applied Multivariate Statistical Analysis.* Englewood Cliffs, NJ: Prentice-Hall, 1982.
Jones GM. Plasticity in the adult vestibulo-ocular reflex arc. *Philos Trans R Soc Lond B Biol Sci* 278: 319-334, 1977.
Jones GM, and Young LR. Subjective detection of vertical acceleration: a velocity-dependent response? *Acta Otolaryngol* 85: 45-53, 1978.
Lee MY, Kim MS, and Park BR. Adaptation of the horizontal vestibuloocular reflex in pilots. *Laryngoscope* 114: 897-902, 2004.
Lichtenberg BK, Young LR, and Arrott AP. Human ocular counterrolling induced by varying linear accelerations. *Exp Brain Res* 48: 127-136, 1982.
Mayne R. *A systems concept of the vestibular organs.* Berlin: Springer-Verlag, 1974, p. 493-580.
Merfeld DM. Spatial orientation in the squirrel monkey : an experimental and theoretical investigation. In: *Aeronautics and Astronautics.* Cambridge: Massachusetts Institute of Technology, 1990.
Merfeld DM, Haburcakova C, Gong W, and Lewis RF. Chronic vestibulo-ocular reflexes evoked by a vestibular prosthesis. *IEEE Trans Biomed Eng* 54: 1005-1015, 2007.
Merfeld DM, Park S, Gianna-Poulin C, Black FO, and Wood S. Vestibular perception and action employ qualitatively different mechanisms. I. Frequency response of VOR and perceptual responses during Translation and Tilt. *J Neurophysiol* 94: 186-198, 2005a.
Merfeld DM, Park S, Gianna-Poulin C, Black FO, and Wood S. Vestibular perception and action employ qualitatively different mechanisms. II. VOR and perceptual responses during combined Tilt&Translation. *J Neurophysiol* 94: 199-205, 2005b.
Merfeld DM, and Young LR. The vestibulo-ocular reflex of the squirrel monkey during eccentric rotation and roll tilt. *Exp Brain Res* 106: 111-122, 1995.
Merfeld DM, Zupan L, and Peterka RJ. Humans use internal models to estimate gravity and linear acceleration. *Nature* 398: 615-618, 1999.
Merfeld DM, and Zupan LH. Neural processing of gravitoinertial cues in humans. III. Modeling tilt and translation responses. *J Neurophysiol* 87: 819-833, 2002.
Merfeld DM, Zupan LH, and Gifford CA. Neural processing of gravito-inertial cues in humans. II. Influence of the semicircular canals during eccentric rotation. *J Neurophysiol* 85: 1648-1660, 2001.
Miles FA, and Lisberger SG. Plasticity in the vestibulo-ocular reflex: a new hypothesis. *Annu Rev Neurosci* 4: 273-299, 1981.

Minor LB, and Goldberg JM. Influence of static head position on the horizontal nystagmus evoked by caloric, rotational and optokinetic stimulation in the squirrel monkey. *Exp Brain Res* 82: 1-13, 1990.
Oman CM. Motion sickness: a synthesis and evaluation of the sensory conflict theory. *Can J Physiol Pharmacol* 68: 294-303, 1990.
Paige GD. Caloric responses after horizontal canal inactivation. *Acta Otolaryngol* 100: 321-327, 1985.
Paige GD, and Sargent EW. Visually-induced adaptive plasticity in the human vestibulo-ocular reflex. *Exp Brain Res* 84: 25-34, 1991.
Paige GD, Telford L, Seidman SH, and Barnes GR. Human vestibuloocular reflex and its interactions with vision and fixation distance during linear and angular head movement. *J Neurophysiol* 80: 2391-2404, 1998.
Paige GD, and Tomko DL. Eye movement responses to linear head motion in the squirrel monkey. I. Basic characteristics. *J Neurophysiol* 65: 1170-1182, 1991.
Park S, Gianna-Poulin C, Black FO, Wood S, and Merfeld DM. Roll rotation cues influence roll tilt perception assayed using a somatosensory technique. *J Neurophysiol* 96: 486-491, 2006.
Peterka RJ, Gianna-Poulin CC, Zupan LH, and Merfeld DM. Origin of orientation-dependent asymmetries in vestibulo-ocular reflexes evoked by caloric stimulation. *J Neurophysiol* 92: 2333-2345, 2004.
Telford L, Seidman SH, and Paige GD. Dynamics of squirrel monkey linear vestibuloocular reflex and interactions with fixation distance. *J Neurophysiol* 78: 1775-1790, 1997.
Watanabe S, Hattori K, and Koizuka I. Flexibility of vestibulo-ocular reflex adaptation to modified visual input in human. *Auris Nasus Larynx* 30 Suppl: S29-34, 2003.
Wei M, and Angelaki DE. Cross-axis adaptation of the translational vestibulo-ocular reflex. *Exp Brain Res* 138: 304-312, 2001.
Wilson V, and Melvill Jones G. *Mammalian Vestibular Physiology*. New York: Plenum, 1979.
Wood SJ. Human otolith-ocular reflexes during off-vertical axis rotation: effect of frequency on tilt-translation ambiguity and motion sickness. *Neurosci Lett* 323: 41-44, 2002.
Woodard D, Parker D, and von Gierke H. Effects of a visual-vestibular stimulus on the vestibulo-ocular reflex. *Aviat Space Environ Med* 58: A198-202, 1987.
Young L. *Perception of the body in space: mechanisms*. Bethesda, MD: 1984, p. 1023-1066.
Young LR, and Meiry JL. *A revised dynamic otolith model*. Pensacola, FL: NASA, 1967, p. 363-368.
Young LR, Oman CM, Watt DG, Money KE, and Lichtenberg BK. Spatial orientation in weightlessness and readaptation to earth's gravity. *Science* 225: 205-208, 1984.
Young LR, Sienko KH, Lyne LE, Hecht H, and Natapoff A. Adaptation of the vestibulo-ocular reflex, subjective tilt, and motion sickness to head movements during short-radius centrifugation. *J Vestib Res* 13: 65-77, 2003.
Zupan LH, and Merfeld DM. Human ocular torsion and perceived roll responses to linear acceleration. *J Vestib Res* 15: 173-183, 2005.

Zupan LH, and Merfeld DM. Neural processing of gravito-inertial cues in humans. IV. Influence of visual rotational cues during roll optokinetic stimuli. *J Neurophysiol* 89: 390-400, 2003.

Zupan LH, Peterka RJ, and Merfeld DM. Neural processing of gravito-inertial cues in humans. I. Influence of the semicircular canals following post-rotatory tilt. *J Neurophysiol* 84: 2001-2015, 2000.

"Because it is there."

--George Mallory

"A fathom deep in sleep I lie
With old desires, restrained before,
To clamor lifeward with a cry,
As dark flies out the graying door;
And so in quest of creeds to share
I seek assertive day again…
But old monotony is there:
Endless avenues of rain."

--F. Scott Fitzgerald

"Quenton, you can do very nearly anything. Haven't you figured that out yet?"

--Bruce Denton

Lord knows I can't change.
Won't you fly high free bird?

--Lynard Skynard