Wii Balance Board
Center of Pressure Software

Kate Howell (Leader), Molly Krohn (Communicator),
Tony Schmitz (BWIG), Chris Fernandez (BSAC)

University of Wisconsin-Madison - Biomedical Engineering Department
Clients: Dr. Kurt Kaczmarek, Kathy Rust & Mitch Tyler
Advisor: Professor Thomas Yen

December 13, 2011

Table of Contents

1. Abstract

The quality of life is greatly affected by individuals suffering from balance disorders. Balance disorders can result from conditions such as neurodegeneration, Parkinson's disease, multiple Sclerosis and more. However, rehabilitation measures can be taken to ease the difficulty of daily activities of suffering from a balance disorder.

Rehabilitation of balance disorders utilize the center of pressure data to determine if the individual is balanced. Current methods in measuring the center of pressure are extremely expensive and inconvenient. Examples include force plates and the BEEP board. However, the Wii Balance Board has the potential to become a cheap, easily accessible solution for rehabilitation. The goal of this design project was to construct an interface to a computer and the Wii Balance Board.

The final design includes the Wii Balance Board connected to a computer via Matlab programming and Bluetooth technology. The interface provides real-time feedback of the user's center of pressure. The visual feedback displays a cursor, representing the center of pressure, and a target, or circle. The objective is to maintain the center of pressure (cursor) inside the target (circle) on the interface. A level of easy, medium, or hard can be selected to change the size of the cursor and circle. A larger cursor and circle represents the easiest level and is reduced in size as the level increases in difficulty. Thus, users of all conditions may use the program. This will allow rehabilitation in a cheap and easy manner for individuals suffering from balance disorders.

2. Background

2.1 Anatomy of Balance

Balancing the human body, in both static and dynamic environments, is achieved through collaboration between the body's visual, vestibular, and proprioceptive systems. The visual system is a component of the central nervous system that processes visual details and creates a sense of visual perception, in addition to carrying out non-image forming photoresponse functions [1]. Meanwhile, the vestibular system is a sensory system that acts as the leading contributor to the sense of balance and spacial orientation of the body. These physiological phenomena are a result of small fluid or jelly filled organs within the vestibular system; the semicircular canals which detect the direction and speed of rotation of the head, as well as the utricle and saccule which respond to linear acceleration of the body, as shown in Fig. 1. [2]. Finally, proprioception can be described as sense of relative position of different body parts and the strength of effort required to make a desired movement [4]. This sense can be broken down into two main categories: conscious proprioception, localized to the posterior column-medial lemniscus pathway in the cerebrum, and unconscious proprioception, localized to the dorsal spinocerebellar tract of the cerebellum [5].

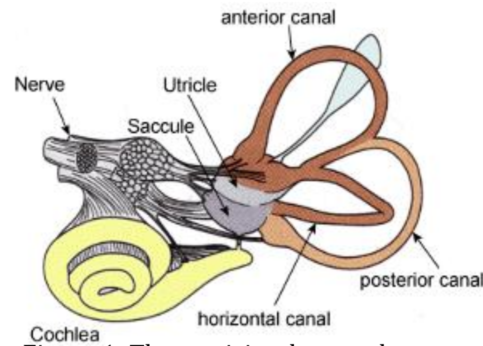


Figure 1: The semicircular canals (shown in brown), the utricle, and saccule of the Vestibule. [3]

2.2 Balance Disorders

Furthermore, it is the neurological processing and comparison between the afferent signals from all three of these unique systems that manifest in the correct balance and postural stability of the whole body. Much alike everything else belonging to the human body, disruption to any of these systems, or related central or peripheral nervous system constituents, can cause balance disorders that have very serious effects on the effected person's quality of life. Balance disorders can impair standing balance, increase the risk of falling, and significantly reduce aforementioned person's ability to perform simple day to day tasks like bathing, cooking, or getting dressed [6]. Further complications like vertigo, dizziness, confusion, disorientation, oscillopsia, nausea, blurred vision, lightheadedness, faintness, and/or fatigue commonly occur in patients affected by a balance disorder [2]. Furthermore, there are many potential causes for balance disorders; neurodegeneration related to age, Parkinson's disease, or multiple Sclerosis, neurotrama caused by stroke or traumatic brain injury, spinal cord injury, neurodevelopement disorders like spinocerebellar ataxia, neurostructural and circulatory problems, some autoimmune diseases, and even some infections. A recent survey provides an example of the startling prevalence and severity of this cause-effect relationship; 44% of patients who had suffered a stroke reported having limitations in most, if not all, activities of daily living which require the coordinated efforts of posture, head and limb control strategies [4].Taking into account that Stroke is the number one leading cause of disease burden[4], it becomes clear how comprehensive and widespread the effects, and even the causes, of balance disorders are in the world.

2.3 Diagnostic and Rehabilitative Stages of Balance Disorders

Related to biomedical engineering, there are various technologies proven efficacious in both the diagnostic and rehabilitative stages of coping with balance disorders. For example, only 10, 20 minute sessions of center of posture rehabilitation exercises on a Nintendo Wii Balance Board over a 4 week period produced an 11% mean improvement in Tinetti's Falls Efficacy Scale post-training compared to pre-training, reflecting greater confidence in the ability to perform functional tasks [7]. Similarly, most important factor in measuring the severity of a person's imbalance, using only commercially available methods, is a person's center of posture. For these purposes, the center of pressure is the resultant force on a force measuring device caused by the sum of a pressure field that causes no moment about that point i.e. the integrated vectorial pressure field. Upon analysis, this value can give detailed and quantifiable information about the user's postural stability and the associated sensory-motor behaviors. For example; dynamic center of pressure characteristics, such as movement direction and temporal ordering, can be obtained by considering the COP profile as fractional Brownian motion and consequently applying power spectral analysis or bridge detrended scaled window variance analysis methods [8]. Many systems that measure center of pressure are available on the market today and the NeuroCom® Balance Manager - Dynamic Posturography System is often seen as an industry standard among these systems.

2.4 Limitations in Present Center of Pressure Measurement Systems

However, there are currently many limitations to today's center of posture measurement systems. They are extremely expensive – ranging from $75,000 to $115,000 according to the Tactile Communication and Neurorehabilitation Laboratory at UW Madison – and are often exclusively located in large scale rehabilitation and healthcare centers. Additionally, these diagnostic tools have such busy schedules that they are only available to small to medium size healthcare providers by physician referral. Summing all of these logistical problems up, these center of posture measurement devices are geographically, chronologically, and fiscally difficult to acquire and use, both as a healthcare provider and as a patient.

Additionally, these powerful and accurate systems do not provide real-time center of pressure information to the user. In any kind of postural stability and balance training, instantaneous real-time feedback is an invaluable therapeutic tool. This is because it allows patients to self-evaluate and make deliberate efforts to correct postural deficits in real time. In cases where neurotrama or balance disorders could be permanent, it is essential to allow these patients an opportunity to learn to reorient themselves strictly through external environmental feedback, independent of any vestibular or proprioceptive cues.

3. Motivation

It is proven through rigorous clinical testing that center of posture rehabilitation systems are immensely beneficial to patient's short and long-term level of functioning with regards to balance disorders. Conversely, the affordability, availability, and complexity of these tools very greatly reduce the ability of these systems to be accessible to the optimal number of people who could benefit from them. Moreover, the lack of continuous real-time feedback does not give patients the ability to learn from the systems audiovisual outputs to reorient themselves to upright standing, balanced positions – a task their brains cannot accomplish independently due to balancing disorders. It is the culmination of these negative properties of current postural rehabilitative technology that motivates us to design a more utilitarian, affordable, simple to use, and transportable neurorehabilitative device geared specifically towards balance disorder rehabilitation and also includes audiovisual feedback to the user. In conclusion, a device that challenges the user's ability to balance, illicits a physical response beginning at a static level, removes the limiting traits of current center of posture measurement systems, and could be made available for use at home at a patients convenience would revolutionize the field of balance disorder rehabilitation.

4. Current Methods

4.1 Force Plates

Patients with balance disorders currently have to travel to small to medium-sized healthcare facilities to have access to a diagnostic force plate system. These large and high-tech systems do not allow for instant feedback of center of position. Real-time feedback is essential for therapeutic progress and recovery. These systems cost between $4,000-$80,000 which is unattainable for the average patient [9].

4.2 *NeuroCom*

The NeuroCom provides assessment and rehabilitation of balance and posture stability through dynamic tests that resemble daily life situations. There are multiple products NeroCom offers to assess a variety of disabilities. It tests visual stimulus, translation and rotation of force plate, blindfolded vs. visual displays, squatting and standing on one leg, and combinations of the

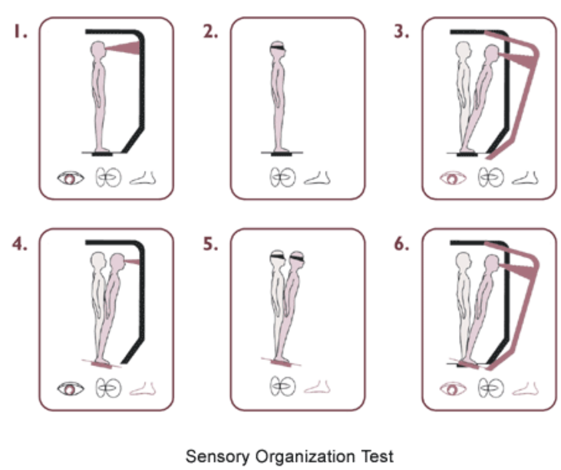


Figure 2: A description of the multiple tests that patients undergo while using the NeuroCom product. [10]

previous tests. They also supply games that force a patient to tilt there body back and forth to interact with the game. NeuroCom provides a great variety of options for rehabilitation, but lacks the convenience of practicing in your own home. The cost of one of the products is unaffordable to an average user.

5. Problem Statement

Currently, there is no inexpensive and widely accessible device that can measure the center of pressure for balance and postural rehabilitation training. Our goal is to design a software interface that receives information from the Wii Balance Board which will incorporate immediate auditory and visual feedback via headphones and a television screen.

5.1 Product Design Specifications

Our client has given us some specific requirements that they want us to follow in designing a rehabilitation tool for people of gait and posture disabilities. Our product must fit a few physical and operational characteristics. The device must withstand daily wear and tear in multiple home envirnments while lasting for up to five years. It must be able to hold 150 kg of vertical pressure while weighing less than 18 kg. The device must have an accuracy that is nearly equivalent to a clinical grade force plate. Our design must also follow some production and miscellaneous requirements. Our team is responsible for production of one finished product costing under $600 total. The device must capture signals of 40 Hz and to be low pass filtered that has a cut-off frequency between 4-12 Hz. The input should also be adjustable to +/- 10% increments. Lastly, the final product must be not be copying any current patents and must be easy and enjoyable for use by the patient.

5.2 Design Alternatives

The design requires multiple components including an instrument to measure the center of pressure or balance, a feedback system, and, depending on the instrument, a computer to interpret and record feedback. The focus of the brainstorm was to generate possible instruments that would be affordable, accurate and portable. After selection of the device then the other design components can be considered. There were three viable methods considered: a laboratory grade force plate, a Nintendo Wii Balance Board, and the Rolyan© BEEP(Balance Enhancement Exercise Program) Board.

5.2.1 Laboratory Grade Force Plate

Force plates are used for biomechanical purposes to study gait, balance, and locomotion because they can determine the center of pressure of an object. Figure 3 displays that force plates measure shear force components (Fx and Fy), a vertical force component (Fz) and the three moments corresponding to the axes. From these

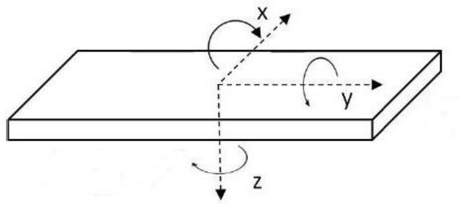


Figure 3: A force plate with the forces and moments it measures. [11]

inputs the center of pressure can be calculated. The measurements are done by four transducers, either strain gage or piezoelectric transducers, which as placed in the pedestals of the platform as seen in figure 4.

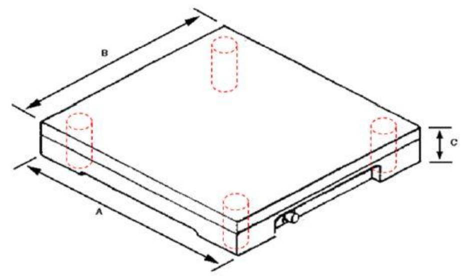


Figure 4: A force plate displayed where the four pedestals are with the transducers in them in. [11]

On average forces plates are 45 by 50 cm but can be purchased in variety of sizes. These plates may be portable or mounted models. When selecting a force plate for balance, to collect the best results a plate with the lowest capacity (meaning the highest sensitivity) should be used because the vertical force would not exceed body weight and the changes in the horizontal forces are small. This alternative would require an amplifier and computer with software to receive data. [12]

5.2.2 Nintendo Wii Balance Board

Nintendo released the Wii Balance Board in 2007 as an accessory to the Wii console. Now it is being used an instrumental tool in many research projects because of its affordability and accuracy. [13]The board operates similarly to a force plate by containing four transducers (strain gages) in each of the four cylindrical legs that can be seen in figure 5 that assess force distribution and the resultant movements in COP.

Figure 5: The dorsal side of a Wii Board displays the four legs each with a transducer. [13]

Figure 6: The Nintendo Wii Balance Board is shaped similar to a bathroom scale and is able to measure center of pressure. [13]

The board is rectangular in shape as seen in figure 6 and weighs 3.5 kilograms. It is capable of supporting up to 150 kg and is powered by four AA batteries allowing for 60 hours of operation. The board communicates using Bluetooth technology and is able to send information back at 60 signals per second. This alternative would require a computer with a built in Bluetooth connection or the addition of a Dongle to receive the feedback as well as a software program to run calculations. [14]

5.2.3 Rolyan® BEEP(Balance Enhancement Exercise Program) Board

The BEEP Board was designed to be used at home for balance rehabilitation. It is designed as a first class lever where an unbalance weight distribution will cause the platform to tilt as seen in figure 7 An audible beep signals when weight shifts away from center. The board has a height of 6.35 cm and measures 40.64 x 63.5 cm. The board has a maximum 300-lb. weight capacity. With this alternative, no exact measure of the center of location can be determined. The user does receive physical response related to their balance unlike the force plates. Another drawback is that it is a primary step function feedback system. Minute changes in performance would not be detected with this data. [15]

Figure 7: The BEEP Board is design after a first class lever where any shift in weight would produce an audible signal. [14]

5.3 Design Matrix

The traditional method of a force plate was compared to a Nintendo Wii Balance Board and a Rolyan® BEEP Board. To evaluate these options the following categories were considered: price, accuracy, mobility, scientific literature, accessibility, and stability/ergonomics. The full evaluation of designs is seen in table 1.

	Weight of Category	Wii Board	Clinical Grade	Rolyan® BEEP Board
Price	0.25	10	1	9
Accuracy	0.2	8.5	10	1
Mobility	0.2	9	2	10
Scientific Literature	0.15	10	10	2
Accessibility	0.1	10	3.5	8
Stability/Ergonomics	0.1	8	9	7
Total Points Awarded	10	9.3	5.4	6.25

Table 1 : The three design alternatives were evaluated and given the ratings above.

Price has the highest significance because that is the main part of the design problem. Systems have been designed with forces plates (See NeuroCam) that meets the needs of the clients' rehabilitation desires, but the price is the key limiting factor. Force plates range in price from $4,000 to $80,000. [12] A force plate to adequately meet the design needs has a cost of $20,000, thus giving the clinical grade force the lowest possible score when considering the design matrix. The Wii Balance Board can be found at most retail stores for $99.99 which is only a fraction of the price of a force plate. Retailers sell the Wii Balance Board as a bundle package including the Wii Fit Game. [13] The BEEP Board costs $185.00. [15]

Accuracy was also an influential factor. The device must be able to give accurate data in order to determine if the user center of pressure is balanced. The force plate exceeds these requirements since it sets the standard for all center of pressure devices and was awarded full points for this category. According to Clark and Bryant's work on testing the validity and reliability of the Wii Balance Board the board had excellent center of pressure path length test-retest reliability within device when testing subjects on a series of single and doubled leg standing tests. [16] BEEP Board only indicates when the user's center of pressure is not balanced. It does this test well, but it is unable to give more in depth information.

Mobility needed to be taken heavily under consideration since the clients' future goals are to have the devices in participants' homes to allow for daily exercise. Force plates can either come in mounted or portable models. Wii Balance Boards were designed to be mobile and useable on most surfaces. It weighs 3.5 kg and comes with stabilizing covers to be place over the legs. [13] The BEEP Board was also designed to be portable and weighs 2.2 kg. [15]

Scientific literature influences the decision because of the time and resource limitation. The clients expressed interest in building off and modifying previously completed work. Both the force plate and the Wii Balance Board have had extensive amounts of research complied about the devices and are publicly available. Minimal scientific research has been done with the BEEP Board since it is a very basic form of feedback.

Accessibility describes how easily these devices can be available. The Wii Balance Board can be purchased at any retail store, therefore giving it the highest accessibility. Both the force plate and the BEEP Board must be purchased through distribution companies. Force plates also often are customized and made when order is placed which makes them less readily available.

Stability and ergonomics were coupled into the same criteria group. The device will be used by participants whom have balance impairments. It must be able to withstand the force from a human standing and stepping on to the board as well has swaying movements. The Wii Board and the force plate are flat and do not move which is less hazardous. The force plate is capable of handling a larger load than the Wii Board. The motion from the level in the BEEP Board makes it less stable than the stationary boards.

6. Final Design

The interface created involves the Wii Balance Board communicating with a PC via Bluetooth and a computer program written in MATLAB. Specifically the PC being used during the design process did not have Bluetooth technology built into it, so a Belkin dongle was used to provide the wireless communication.

6.1 University of Colorado's Code: "bb_record"

Open source code from the University of Colorado's Neuromechanics Lab was utilized as the base of the code. The script of this code is named "bb_record" (balance board record). It has the capabilities to display the center of pressure collected by the Wii Balance Board in real time, as well as the capability to record the center of pressure data. To run the program, it firsts asks to calibrate the board by placing an object with a known weight on it and then entering the weight in kilograms into the appropriate window. The object is then removed, and the interface appears as seen in Fig. 8

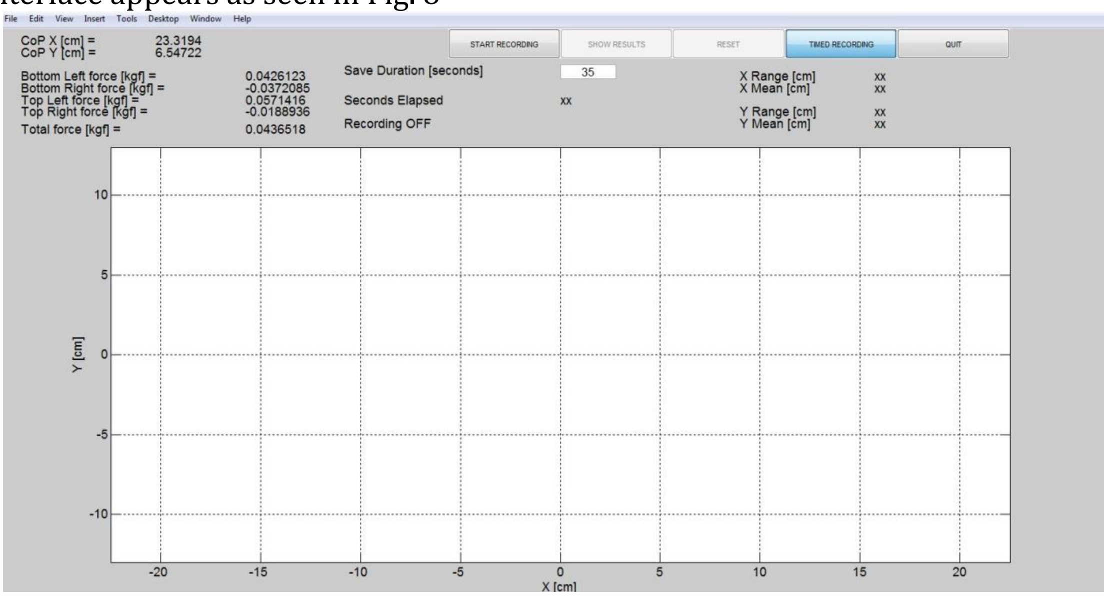


Figure 8: The displayed interface with no load on WBB using bb_record.

The interface lists the location of the center of pressure, the force sensed by each quadrant, and information regarding time for recording. The white grid, as seen in Fig. 8, represents the top platform of the Wii Balance Board. The x and y axes are the length and width, respectively, of the active part of the board with the origin located at the center of board. The center of the board is physically marked on the Wii Balance Board by lines differentiating the four panels. When the board senses a load, a red cursor appears in the white graph area. The size of the cursor is proportional to the amount of the load.

Data is recorded at a frequency around 86.956 Hz. There are two ways to record data. The user can manually control it by starting the program and stopping it when desired. Or, using the time duration feature, a set time can be established prior to use. This allows the user to set the duration of the recording, and the program will automatically stop after the time has elapsed. After data is recorded it can be viewed as a table of values or there is an image displayed as shown in Fig. 9 with the movements tracked and the recorded values plotted with a summary of average center of pressures.

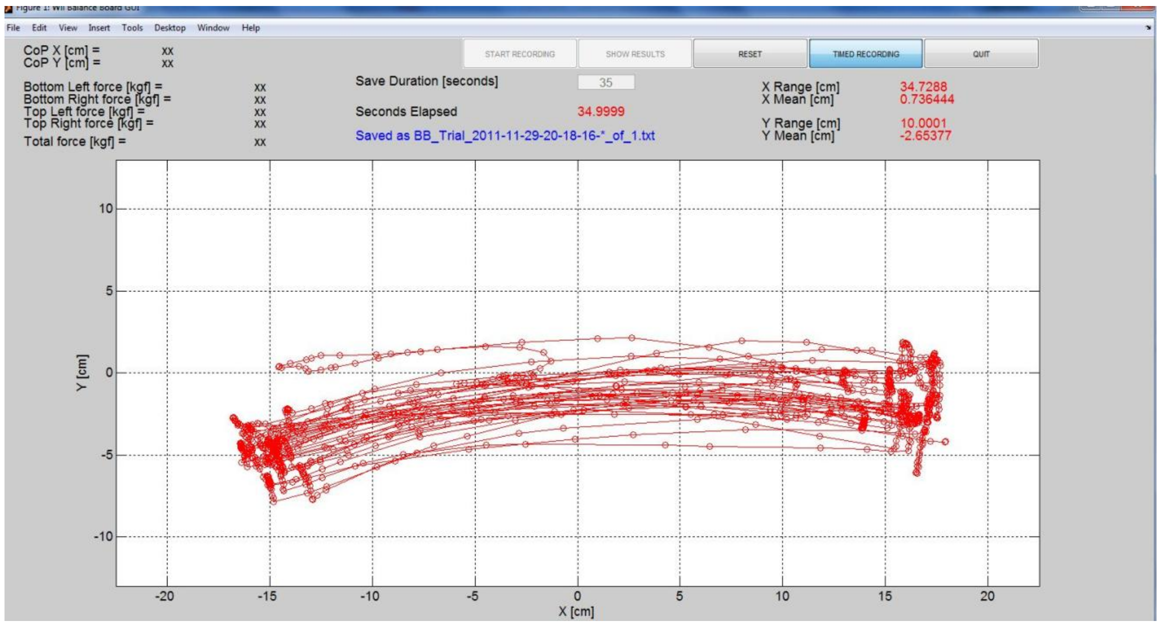


Figure 9: The image of the motion of CoP after recording a moving load on the WBB using “bb_record”.

6.2 Final Code

As stated prior, the CU Neuromechanic’s Lab code was modified to produce a final product for the clients. Several visual elements were modified to meet the requests of the clients. Since the current use of the program is for exercise purposes and not a diagnostic tool, the display of excess numbers may distract the users while viewing the CoP. Therefore, the display was modified to exclude all real time information except the total force and CoP movement. Also, the grid display was modified to eliminate all vertical and horizontal lines, except the two hairline guides along the x-axis and y-axis intersecting at the origin. To ease use, the cursor no longer changes with relation to the amount of force sensed; it remains a constant size corresponding to the level.

The final design consists of 3 levels of difficulty the user may select depending on condition including easy, medium, or hard.

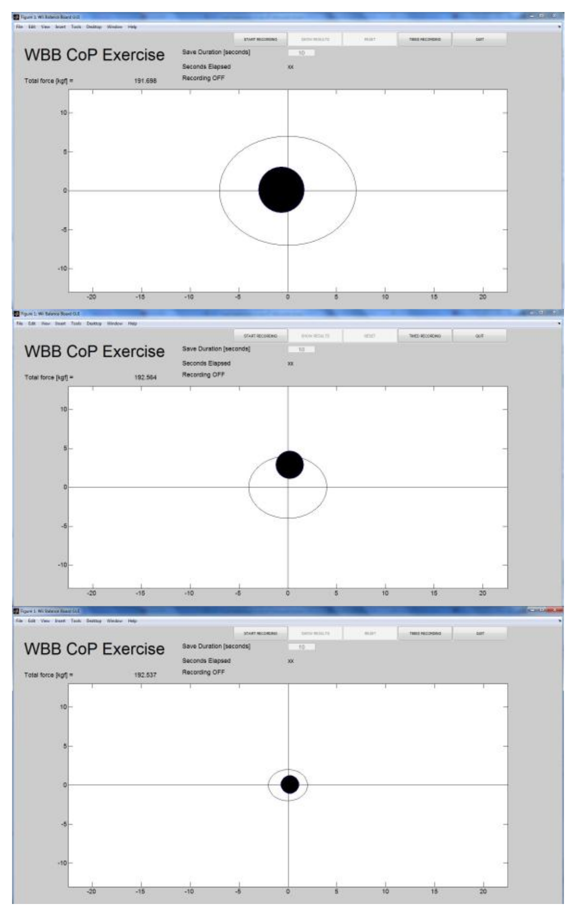


Figure 10: Different levels with the final interface. Easy (top), medium (middle), and hard (bottom). Levels vary by cursor size and target circle size.

Each level displays a target circle in which users aim to keep the cursor (indicating their center of pressure) inside, correlating to a stable balance distributed in both feet. When the program runs, it first requires calibration and then another window prompts the user to select the difficulty level. The levels vary by the target size and cursor size. The easy level consists of the largest target circle and cursor while the hard level has the smallest cursor and target circle as can be seen in Figure 10.

6.3 Budget

The materials needed for the design project included a computer, computer programming software and a Wii Balance Board. A connection between the Wii Balance Board and computer needs Bluetooth. A dongle (~$15) was required for some computers, including the Windows computer in the final design. The Wii Balance Board costs $99.99. If MATLAB and computers are available then budget runs around $115.00. If a license to MATLAB is required that increases the price of the system at least $500.00[17].

7. Testing

Analysis of the accuracy of the interface was performed utilizing comparisons between the physically measured CoP point on the balance board and the CoP as displayed by the interface via the Wii® Balance Board. A 4.54 kg weight was placed on 5 locations of the board as shown in Fig. 11. Three trials (or readings) were obtained from each point. The physical measurements of the CoP, made with a ruler, and readings of the CoP were obtained from the interface. The difference between the theoretical and experimental values was plotted, along with the standard deviation as shown in Figure 12. The standard deviation of (X CoP, YCoP) for each position are as followed: position 1: (0.316, 3.39E-08), position 2: (0.0185, 0.005), position 3: (0.0135, 0.0005), position 4: (0.0097, 2.15E-07), and position 5: (0.0141, 1.60E-09). The greatest difference between the experiment and theoretical CoP data values is less than 0.4 cm. Potential error with this data may result from inaccurate physical measurements of the CoP of the weight. However, as our results demonstrate, the interface is indeed reliable for accurate CoP display.

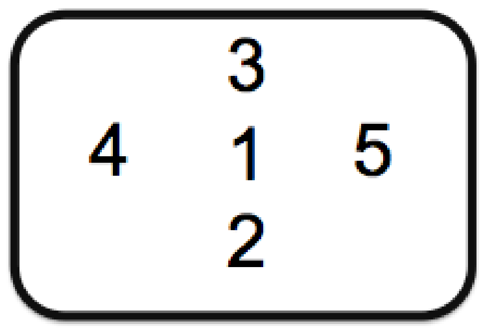


Figure 11: Aerial perspective of positions tested for CoP accuracy.

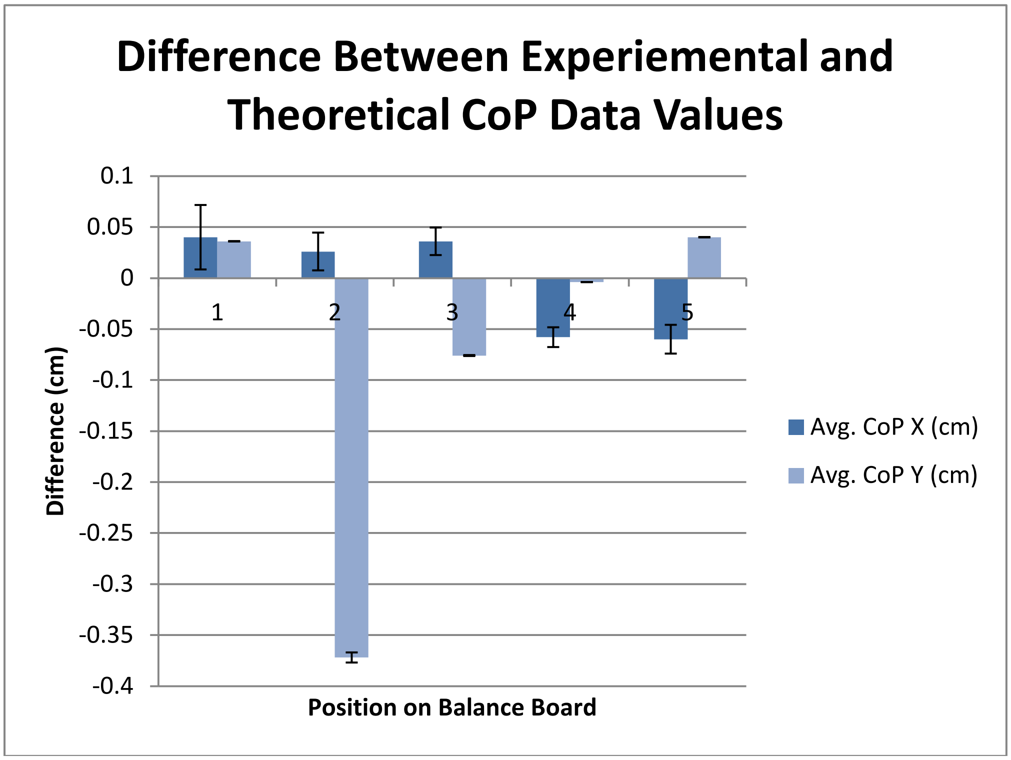


Figure 12: Analysis of Experimental and Theoretical CoP Data Values.

8. Future work

In the future, focus will be placed on improving the MatLab interface to satisfy our client's needs. Of most importance is providing an additional form of feedback, specifically audio, to the users. Since balancing with eyes closed is an important therapeutic procedure, having only visual feedback with the board is not enough. Another adjustment to the code is to allow feedback of time spent within and outside of the target circle in each difficultly level. This will allow for proper initial diagnosis of client's abilities and evaluation of progress made over time. A smaller addition to our project will be to include another option of placing a foam piece on top of the balance board for addition difficulty. Lastly, since our project lacks dual-product capability, a driver will need to be crafted to allow our code to run on Macintosh computer systems.

9. Conclusion

Balance disorders can severely affect the quality of life for some individuals. Previously, there was not a cheap, easily accessible rehabilitation method for enhancing the quality of life for these individuals. By using the Wii Balance Board and a computer, an interface was created to give real-time feedback to the individual to aid in the rehabilitation process. With this convenient trainer, progress in rehabilitation can be greater and faster compared to current methods.

10. Bibliography

[1]: HyperBrain Syllabus Chapter 7. (2010). *The Visual System*. Retrieved from University of Utah School of Medicine website: http://library.med.utah.edu/kw/hyperbrain/syllabus/syllabus7.html

[2]: National Institute of Deafness and Other Communication Disorders . (2009). *Balance Disorders*. Retrieved 2011, from National Institutes of Health website: http://www.nidcd.nih.gov/health/balance/pages/balance_disorders.aspx

[3]: Chagdes, J. (n.d.). Current Research. In *Nonlinear dynamics and control of human posture*. Retrieved from Purdue Research Foundation website: https://engineering.purdue.edu/~jchagdes/research.html

[4]: Mosby's Medical, Nursing and Allied Health Dictionary, Fourth Edition, Mosby-Year Book 1994, p. 1285

[5]: Swenson, R. (2006). Review of Clinical and Functional Neuroscience . In *Somatosensory Systems*. Retrieved from Dartmouth Medical School website: http://www.dartmouth.edu/~rswenson/NeuroSci/chapter_7A.html#Unconscious_sensation

[6]: Bongers, B., & Smith, S. (n.d.). *Interactivating Rehabilitation through Active Multimodal Feedback and Guidance*. Retrieved from University of Technology, Sydney, and Prince of Wales Medical Research Institute website: http://bertbon.home.xs4all.nl/downloads/Interactive%20Rehabilitation%20print.pdf

[7]: Assessing and training standing balance in older adults: A novel approach using the 'Nintendo Wii' Balance Board

[8]: Pan, N., Wu, J., Williams, K., & Wang, Y. (2006). Effects of Floor Coverings on Posture Steadiness and Locomotion Stability (NTC Project No. S04-CD03). Retrieved from National Textile Center Annual Report website: http://www.ntcresearch.org/pdf-rpts/AnRp06/S04-CD03-A6.pdf

[9] "Choosing a Force Plate | Standard Portable Force Plates for Research and Analysis of Gait, Balance, and Sports Performance | AMTI Products." *AMTI | Multi-Axis Force Plates, Force Sensors, and Testing Machines | Watertown, MA*. Web. 25 Oct. 2011. <http://www.amti.biz/fps-guide.aspx>.

[10] "Sensory Organization Test SOT: NeuroCom Protocols: NeuroCom - Products."*NeuroCom, a Division of Natus - Balance Specialists*. Web. 25 Oct. 2011. <http://resourcesonbalance.com/neurocom/protocols/sensoryImpairment/SOT.aspx>.

[11] *Force Plate User Manual*. Bertec Corporation: Copyright 2007

[12] *AMTI: Force and Motion.* (n.d.) Date accessed online: October 16, 2011) http://www.amti.biz/fps-guide.aspx

[13] Nintendo Wii. (2011) Date accessed online: October 19, 2011) http://www.nintendo.com/wii/console/accessories/balanceboard

[14]Okamoto et al. "Storage Medium Storing a Load Detecting Program and Load Detecting Apparatus." United States Patent Application Publication (Pub. No. US 2009/0093305 A1). April 9, 2009. (Date accessed online: October 16, 2011) http://www.freepatentsonline.com/20090093305.pdf

[15] *Patterson Medical*. (n.d.) Date accessed online: October 17, 2011) http://pattersonmedical.com/app.aspx?cmd=get_product&id=100487

[16]Clark A., et al. (2010) Validity and reliability of the Nintendo Wii Balance Board for assessment of standing balance. Gait & Posture 31. 307-310

[17] Mathworks. (2011) Date accessed online: December 12, 2011 http://www.mathworks.com/store/productIndexLink.do

11. Appendix

11.1 Design Specifications

1. Physical and Operational Characteristics
    - a. *Performance requirements*: The device must be able to withstand normal wear and tear from daily use. The software and feedback must be easily understandable by its users.
    - b. *Safety*: The device must be able to withstand a person standing and shifting weight that has balance and gait disorders.
    - c. *Accuracy and Reliability*: Using intraclass correlation coefficients, the COP path length test-retest reliability within the device should be between 0.66-0.94 and when compared to laboratory grade force plate, the device should be between 0.77-0.89.
    - d. *Life in Service*: The device must run 20 minutes daily for the length of study which could last between three months to one year long.
    - e. *Shelf Life*: The device should be functional for at least five years of use. The Wii Balance Board requires four AA batteries to power 60 hours of use.
    - f. *Operating Environment*: The device will be operated in a variety of different household environments from bedrooms and family rooms, to garages and basements.
    - g. *Ergonomics*: The device must be able to withstand a maximum weight of 150 kg.
    - i. *Weight*: The device must be able to be moved by an average sized adult, so it needs to weigh less than 18 kg.
    - j. *Materials*: The system will include a Wii Balance Board or an external weight shift sensing device, Bluetooth capable computer and accessories, and a system for audio output.
    - k. *Aesthetics*, *Appearance, and Finish*: The audio output needs to produce non-irritable noises and the visual output needs to be easily understood in relation to their current position.
2. Production Characteristics
    - a. *Quantity*: The design team is only responsible for producing one product, but the client may use multiple balance systems to distribute to all of the research participants.
    - b. *Target Product Cost*: The entire system should cost a total of under $600.

3. Miscellaneous
    - b. *Customer*: The client wants a device that captures signals at 40 Hz and to be low pass filtered with a cut-off frequency of 4-12 Hz. Input gain should be adjustable to +/- 10% increments.
    - c. *Patient-related concerns*: The device must be easy and enjoyable to use for the patient.
    - d. *Competition*: The Wii Balance Board has already been used for physical rehabilitation with video game-like displays of feedback and auditory feedback via personal headphones. Currently there are no patents impeding our prospects of a unique design aligning to our client's wanting.
    Virtual Wiihab from the article "Lean on Wii: Physical Rehabilitation With Virtual Reality and Wii Peripherals" by F. Anderson, M. Annett, and W. Bischof. As well as eBaViR from the article "Effectivness of a Wii balance board-based system (eBaViR) for balance rehabilitation: a pilot randomized clinical trial in patients with acquired brain injury" by J. Gil-Gomez

11.2 Final Code

```
% bbrecord is a script that runs a user friendly interface for viewing
% center of pressure data from a Wii Balance Board
%
% It uses the WiiLab patch that includes balance board support,
% http://klab.wikidot.com/wii-proj)
%
% System requirements: Windows XP & Matlab R2007a
%
% Authors of bbrecord: Members of the University of Colorado, Boulder
% Neuromechanics Lab (PI Alaa Ahmed)

% versions 1_1-1_3: Andrew Kary
% version 1_0: Helen J. Huang, Sergio Perez

%% Setting the path

global bb f go flag_saving connected quitgui collecting filename ...
    deletedfile enable_results reset C sensor_avg_upright saveduration ...
    h_timerinput h_startbutton h_timerbutton flag_stop flag_timed...
    flag_start starttoggle Upper_time drivename h_resultsbutton;

% housekeeping
if ~exist('connected','var') || isempty(connected)
    clear all;
    delete first.mat;
end
close all; clc; clear first

% find the path for this m file
thisone = mfilename('fullpath');
backslashes = find(thisone == '\');
thispath = thisone(1:(backslashes(end)-1));
cd (thispath) % make sure that we are in the directory of this file

% check if this is the first time that bbrecord has been run
if ~exist('first.mat','file') || isempty(drivename)
    first = 1;
else
    load first;
end



if first == 1

    % find the address of the directory
    [stat,drive] = fileattrib;

    % find all of the slashes
    slashes = find('\' == drive.Name);

    % find the address of the folder that the directory is nested inside of
    drivename = drive.Name( 1 : slashes(end)-1 );

    % add the path of the home folder, CU_Wii
    addpath([drivename '\CU_Wii'])

    % add the path containing the Wii balance board function we wrote
    addpath([drivename '\CU_Wii\WiiBBFunctions'])

    % add the path containing simple graphics functions of WiiLAB
```

```
    addpath([drivename '\WiiLAB\WiiLAB_Matlab\EG111-H'])

    % add the path containing the Wiimote functions of WiiLAB
    addpath([drivename '\WiiLAB\WiiLAB_Matlab\WiimoteFunctions'])

    % creat sound player
    [Y,FS,NBITS] = auread('bloop.au');
    player = audioplayer(Y, FS);

    save first first
end


%% Make sure that MATLAB talks to the Balance Board

if ~exist('connected','var') || (isempty(connected)) || (connected == 0)

    % from WiiLAB wiimote.m
    bb = Wiimote();
    bb.Connect(); % connect to balance board

    connected = 1;

end

%% !! PARAMETERS TO SET !!

if first == 0
    % set path for saving data
    pathname = [drivename '\CU_Wii\Data'];

    % change to that directory
    cd(pathname);

    %answer = questdlg('Would you like to open the data folder?', ...
      %  'View Data Folder?', 'Yes', 'No','Yes');
   % switch answer,
    %    case 'Yes',
     %        winopen(pathname);
      %  case 'No',
       %     cal = 0;
   % end


end


%% Balance Board Data

% check that the balance board is REALLY connected
if bb.isConnected() > 0

    %     bb.wm.GetBalanceBoardCoGState(); % returns rough values for the
center-of-pressure into variable called 'cog' x,y [cm]. positive direction is
right, back
   %     bb.wm.GetBalanceBoardSensorState(); % returns values for the 4 sensors
into a variable called 'sensors' [no obvious units]
    %     bb.wm.GetBatteryState(); % 1 for full charge, 0 for absolutely empty

%% Calibrate
    if first == 0
```

```
        if ~exist('sensor_avg_upright','var') || isempty(sensor_avg_upright) ||
C(1) == 0
            cal = 1; % do this if the BB has not been calibrated
        else
            % do this if the BB has been calibrated
            answer = questdlg('Would you like to calibrate the Balance Board
now?', ...
                'Calibrate?', 'Yes', 'No','Yes');
            switch answer,
                case 'Yes',
                    cal = 1;
                case 'No',
                    cal = 0;
            end
        end

        if cal == 1

            [C, quitgui, sensor_avg_upright ] = OneWeightCalibrate (bb); %
calibrate
%              save C C

        end

    end

    %% Set Level of Difficulty
    diffanswer = questdlg('Which level of difficulty would you like?', ...
               'Level of Difficulty', 'Easy', 'Medium', 'Hard', 'Easy');
            switch diffanswer,
                case 'Easy',
                   leveldiffradius = 7;
                   cursorsize = 100;
                case 'Medium',
                    leveldiffradius = 4;
                    cursorsize = 60;
                case 'Hard',
                    leveldiffradius = 2;
                    cursorsize = 40;
            end

    %% Create GUI figure for displaying BB data

if first == 0

    saveduration = 10; % How long do you want to save the data for by default?
    cycletime = 1/30 ; % how long should each iteration take? [seconds]
    Upper_time = 3600; % longest allowable trial [s]

    screensize = get(0,'ScreenSize');
    % figure('units','normalized','outerposition',[0 0 1 1])
    f= figure('Visible','off','color',[0.8 0.8
0.8],'units','normalized','outerposition',[0 0 1 1],'Name','Wii Balance Board
GUI');
    % maxfig(f,1);

    subplot('position', [.1  .8  .8  .2]);
    set(gca, 'visible','off','Units', 'normalized');

    x1 = -.1;
    x2 = .05;
    x3 = .15;
    x4 = .26;
```

```
    x5 = .7;
    x6 = 0.5;

    % COP Text
    %x_copDispLbl = text(x1,0.9,'CoP X [cm] = ','fontsize', 13);
   % y_copDispLbl = text(x1,0.8,'CoP Y [cm] = ','fontsize', 13);
   % x_copDisp = text(x2, 0.9, 'xx','fontsize', 13);
   % y_copDisp = text(x2, 0.8, 'xx','fontsize', 13);

    % Force sensor values
    %bb_BLDispLbl = text(x1, 0.6, 'Bottom Left force [kgf] = ','fontsize', 13);
    %bb_BRDispLbl = text(x1, 0.5, 'Bottom Right force [kgf] = ','fontsize',
13);
    %bb_TLDispLbl = text(x1, 0.4, 'Top Left force [kgf] = ','fontsize', 13);
    %bb_TRDispLbl = text(x1, 0.3, 'Top Right force [kgf] = ','fontsize', 13);
    bb_TotalispLbl = text(x1, 0.15, 'Total force [kgf] = ','fontsize', 13);

    %Group Project Title
    bb_BLDispLbl = text(x1, 0.6, 'WBB CoP Exercise','fontsize',35);

   % bb_BLDisp = text(x3, 0.6, 'xx','fontsize', 13);
   % bb_BRDisp = text(x3, 0.5, 'xx','fontsize', 13);
   % bb_TLDisp = text(x3, 0.4, 'xx','fontsize', 13);
   % bb_TRDisp = text(x3, 0.3, 'xx','fontsize', 13);
    bb_Totaldisp = text(x3, 0.15, 'xx','fontsize', 13);

    % Display save duration time
   text(x4, 0.65, 'Save Duration [seconds]', 'fontsize', 13);


    % Text indicating save status
    % save_filename = text(x4, 0.5, ['Save filename: ' filename], 'fontsize',
    % 13);
    savestatustext = text(x4, 0.2, 'Recording OFF', 'fontsize', 13);
    savenametext = text(x4, 0.15, 'xx', 'fontsize', 13, 'visible', 'off');

    text(x4, 0.4, 'Seconds Elapsed', 'fontsize', 13);
    save_duration_text = text(x6, 0.4, 'xx', 'fontsize', 13);

    y0 = .7;
    y1 = .6;
    y2 = .5;
    y3 = .3;
    y4 = .2;

    % Text indicating range
   %text(x5, y1, 'X Range [cm]', 'fontsize', 13);
    %text(x5, y3, 'Y Range [cm]', 'fontsize', 13);
    %text(x5, y2, 'X Mean [cm]', 'fontsize', 13);
    %text(x5, y4, 'Y Mean [cm]', 'fontsize', 13);

    %xrange = text(0.85, y1, 'xx', 'fontsize', 13);
    %xmean = text(0.85, y2, 'xx', 'fontsize', 13);

    %yrange = text(0.85, y3, 'xx', 'fontsize', 13);
    %ymean = text(0.85, y4, 'xx', 'fontsize', 13);


    % Flags for controlling condition statements
    flag_saving = 0;
    flag_start = 0;
    flag_stop = 0;
    flag_timed = 0;
```

```
    % Initialize variables
    reset = 1;

    % creates gui buttons
    h_startbutton = start_button_toggle; % for starting to save data
    h_quitbutton = quit_button; % for quitting the program
    h_resultsbutton = results_button; % for pausing the program
    h_resetbutton = reset_button; % for resetting the program
    h_timerbutton = timer_button; % for deciding whether to use the automatic
timer or the manual timer

    h_timerinput = timer_input(x6, 0.915, saveduration); % make the timer input
box

    tStart = tic; % Start timer

end

%% Get BB Data
    if first == 0

        % initialize
        quitgui = 0; % 1 for quit

        while quitgui == 0

            % get rid of old data
            if reset == 1

               % set(xrange, 'String', 'xx', 'Color', 'k' );
               % set(xmean, 'String', 'xx', 'Color', 'k' );
               % set(yrange, 'String', 'xx', 'Color', 'k' );
               % set(ymean, 'String', 'xx', 'Color', 'k' );
                set(h_startbutton, 'Enable', 'on')
                set(h_resetbutton, 'Enable', 'off')
                set(save_duration_text, 'String', 'xx', 'color', 'k');
                set(savestatustext, 'Color', 'k', 'String', 'Recording OFF');
                set(savenametext,'visible','off');

                clear first
                if exist('first','file')
                    delete first.mat
                end

                reset = 0; % don't need to reset anymore
                timestartsaving = 0; currentsaveduration = 0;
                iter = 0; % loop iteration number
                go = 1; % set go = 1 to execute loop, after reaching
saveduration, go set to 0, stopping the loop
                collecting =1;
                trials = 0; % number of trials so far
                color = 'k';
                deletedfile = 0;
                enable_results = 0;
                starttoggle = 1;
                clear answer
                datalog = zeros(2*Upper_time/cycletime,8);
                fclose all;

                if flag_timed == 1
                    set(h_timerinput,'Enable','on');
                end
```

```
                set(f,'Visible','on'); % show the figure
                tStart = tic; % reset stopwatch

            end

            while go == 1
                iter = iter + 1; %increment iteration counter

                data.bb.time(iter) = toc(tStart); % find the elapsed time now
                cog = bb.wm.GetBalanceBoardCoGState(); % find the center of
pressure
                sensors = (bb.wm.GetBalanceBoardSensorState()-
sensor_avg_upright)./C; % find the force on each sensor, calibrated

                % fix iteration duration
                t_el = toc(tStart);
                if t_el/iter < cycletime
                    pausetime = (cycletime*iter) - t_el;
                    pause (pausetime)
                end

                % save data when triggered
                if flag_saving % this condition only executes the following
when we want to save data
                    if flag_start
                        % this condition saves a single cycle of time to
                        % timestartsaving. Otherwise, timestartsaving would be
                        % written over and over and over.

                        timestartsaving = toc(tStart);
                        iterstartsaving = iter;
                        set(savestatustext, 'Color', 'r', 'String', 'Recording
ON');


                        starttype = get(h_startbutton, 'Style');

                        switch starttype
                            case 'pushbutton'
                                set(h_startbutton, 'Enable', 'off')
                                starttoggle = 0;
                            case 'togglebutton'
                                starttoggle = 1;
                        end


                        flag_start = 0;
                        closefile = 1;
                        trials = trials+1;

                        color = 'k';

                    end
                    currentsaveduration = toc(tStart) - timestartsaving;
                    set(save_duration_text, 'String', currentsaveduration);
                    datalog(iter,:) = [iter,data.bb.time(iter),cog(1),-
cog(2),sensors(1),sensors(2),sensors(3),sensors(4)];
                end

                % change text on GUI
                %set(x_copDisp, 'String', cog(1));
                %set(y_copDisp, 'String', -cog(2));
                %set(bb_BLDisp, 'String', sensors(1));
```

```
                %set(bb_BRDisp, 'String', sensors(2));
                %set(bb_TLDisp, 'String', sensors(3));
                %set(bb_TRDisp, 'String', sensors(4));
                weight = sum(sensors);
                set(bb_Totaldisp, 'String', weight);

                data.bb.copx(iter) = cog(1);
                data.bb.copy(iter) = -cog(2);
                data.bb.BLforce(iter) = sensors(1);
                data.bb.BRforce(iter) = sensors(2);
                data.bb.TLforce(iter) = sensors(3);
                data.bb.TRforce(iter) = sensors(4);

                % plot the CoP

                size = cursorsize;
                ang = 0:0.01:2*pi;
                xp =  leveldiffradius*cos(ang);
                yp =  leveldiffradius*sin(ang);
                xvert = [0 0];
                yvert = [-13 13];
                xhor = [-23 23];
                yhor = [ 0 0];



                if weight > 5
                    subplot('position',[.1  .1  .8  .7]);


                    plot(xvert,yvert, 'k-',xhor,yhor, 'k-', xp, yp, 'k',
cog(1), -cog(2),'bo', 'MarkerFaceColor',color, 'MarkerSize', size);
                    axis([-22.5 22.5 -13 13]);


                    %xlabel('X [cm]')
                    %ylabel('Y [cm]')
                    set(gca, 'fontsize', 13); grid off;
                else

                    subplot('position',[.1  .1  .8  .7]);
                    linethick = 6;

                    plot(xvert,yvert, 'k-', xhor,yhor, 'k-',xp, yp, 'k', 100,
100, 'wo', 'MarkerFaceColor','w', 'MarkerSize', size);
                    axis([-22.5 22.5 -13 13]);


                    %xlabel('X [cm]')
                   % ylabel('Y [cm]')
                    set(gca, 'fontsize', 13); grid off;
                end

                set(0,'CurrentFigure',f)

                % stop saving data
                if ((currentsaveduration >= saveduration) == 1) && (starttoggle
== 0) && flag_saving == 1
                    flag_stop = 1;
                end

                if (flag_stop == 1)
```

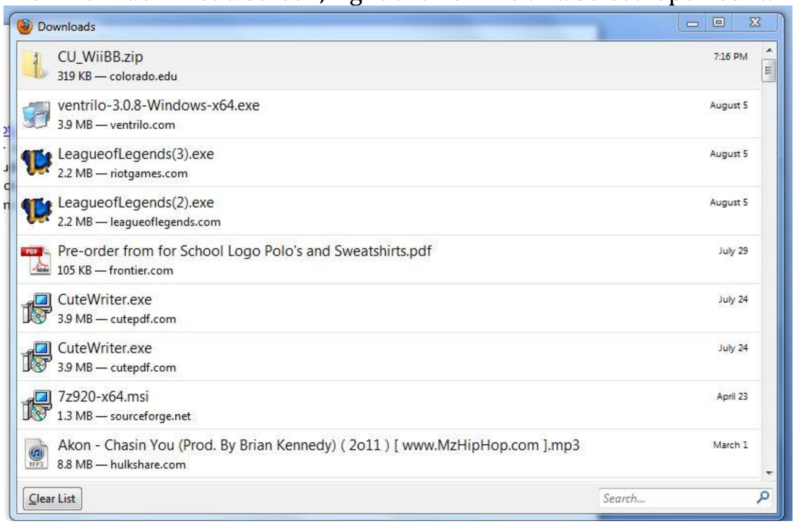
```
                    % do this at the end of a recording period
                    flag_saving = 0;
                    flag_stop = 0;

                    flag_timed = get(h_timerbutton,'Value');

                    set(h_timerbutton,'Enable','on');
                    set(savestatustext, 'Color', 'k', 'String', 'Recording
OFF');

                    color = 'k';

                    if closefile,
                        fclose all;
                        closefile = 0;
                        play(player); % go "ding"
                    end

                    % do this at the end of two recording periods
                    if trials == 1
                        set(h_resultsbutton, 'Enable', 'on')
                        set(h_resetbutton, 'Enable', 'off')

                    else
                        set(h_startbutton, 'Enable', 'on');
                    end
                end




            end

%% Plot the traces
%%If we don't need to show the trace, delete down to 538.
            if quitgui == 0  && deletedfile == 0

                set(h_startbutton, 'Enable', 'off')

                % load the data
                frame = datalog(:,1);
                time = datalog(:,2);
                copx = datalog(:,3);
                copy = datalog(:,4);
                f1 = datalog(:,5);
                f2 = datalog(:,6);
                f3 = datalog(:,7);
                f4 = datalog(:,8);

                % find the indices corresponding to beginnings and ends
                start_indices = find( diff(frame) > 1 ) + 1;
                end_indices = find( diff(frame) < -1 );
                end_indices(length(end_indices)+1) = max(frame); % otherwise it
will miss the last end

                subplot('position',[.1  .1  .8  .7]);

                weight = (f1+f2+f3+f4);
                thresh = weight > 5;

                % change visible text
                set(x_copDisp, 'String', 'xx' );
                set(y_copDisp, 'String', 'xx' );
```

```
                set(bb_BLDisp, 'String', 'xx' );
                set(bb_BRDisp, 'String', 'xx' );
                set(bb_TLDisp, 'String', 'xx' );
                set(bb_TRDisp, 'String', 'xx' );
                set(bb_Totaldisp, 'String', 'xx' );

                % pick colors
                cc=hsv(length(start_indices));

                trace_ind = 1;
                while trace_ind <= length(start_indices)
                    xy = find(thresh( start_indices(trace_ind)+...
                        1:end_indices(trace_ind)) )+start_indices(trace_ind) ;
                    currentsaveduration = time(end_indices(trace_ind)) ...
                        - time(start_indices(trace_ind));

                    % plot figures
                    plot(copx(xy),copy(xy),'color',cc(trace_ind,:))
                    hold on
                    plot(copx(xy),copy(xy),'o','color',cc(trace_ind,:))
                    axis([-22.5 22.5 -13 13]);
                    set(gca, 'fontsize', 13);
                    grid on;
                    xlabel('X [cm]')
                    ylabel('Y [cm]')

                    % change visible text to X and Y ranges & means
                    set(xrange, 'String', range(copx(xy)), 'color',
cc(trace_ind,:) );
                    set(xmean, 'String', mean(copx(xy)), 'color',
cc(trace_ind,:) );
                    set(yrange, 'String', range(copy(xy)), 'color',
cc(trace_ind,:) );
                    set(ymean, 'String', mean(copy(xy)), 'color',
cc(trace_ind,:) );

                    % show the elapsed time for that trial
                    set(save_duration_text, 'String', currentsaveduration,
'color', cc(trace_ind,:));


                    save trace_ind trace_ind

                    ind = trace_ind;
                    if trace_ind == length(start_indices)
                        trace_ind = trace_ind + 1; % don't wait for the button
to be pressed
                    else
                        % wait for the button to be pressed
                        while trace_ind == ind
                            load trace_ind;
                            pause(.01)
                        end
                    end

                end

                % start fresh
                set(h_resetbutton, 'Enable', 'on')
                set(h_resultsbutton, 'String', 'SHOW RESULTS', 'enable', 'off')
                hold off

                title = 'BB_Trial_'; % not formatted in TeX
```

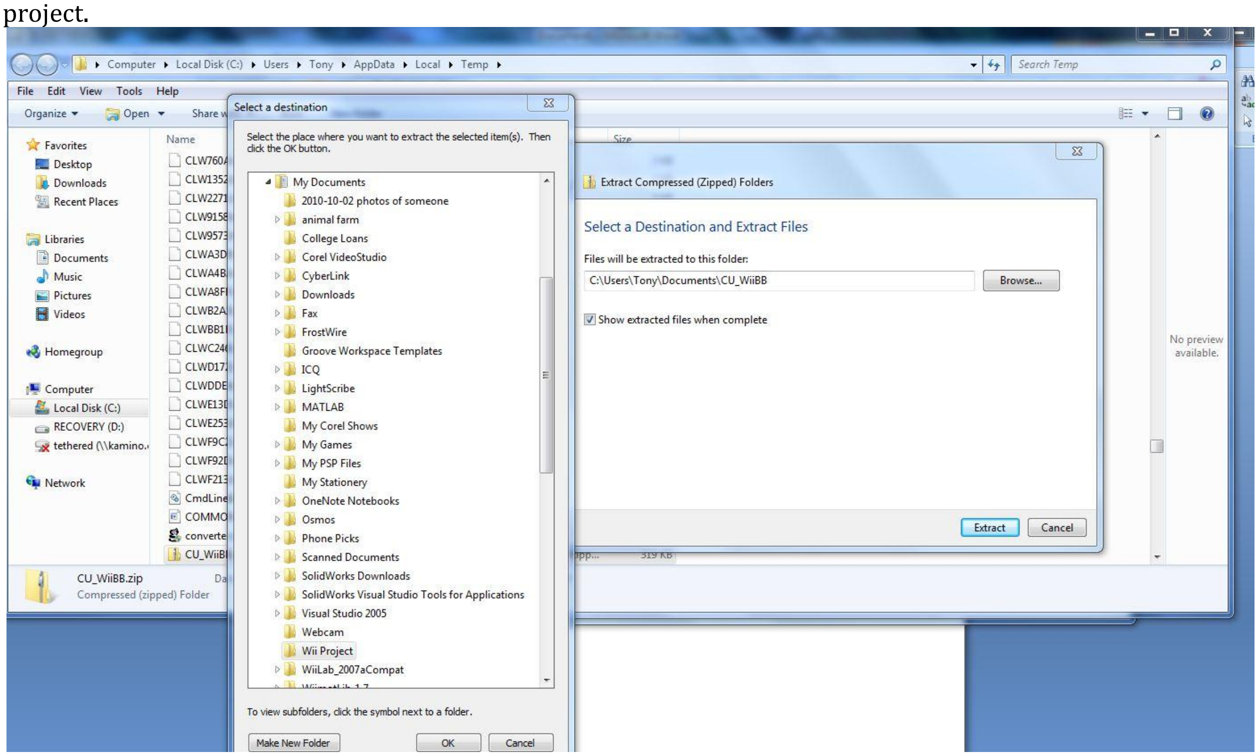
```
                date = datestr(now, 'yyyy-mm-dd-HH-MM-SS');

                % write the data for each trial to a separate file
                for ii = 1 : length(start_indices);

                    % pick the filename
                    filename = strcat(title,date,'-',num2str(ii),...
                        '_of_',num2str(length(start_indices)),'.txt');

                    % create and open the text file for read and append
                    % access
                    fid = fopen(filename,'a+');

                    % write column titles
                    fprintf(fid,'%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\n', ...
                        'Cycle', 'Time [s]', 'CoP X [cm]', 'CoP Y [cm]', ...
                        'Back Left Force [kgf]', 'Back Right Force [kgf]'...
                        , 'Front Left Force [kgf]', 'Front Right Force [kgf]');

                    % write data
                    for jj =
start_indices(ii):end_indices(ii);%1:end_indices(length(end_indices))
                        fprintf(fid,'%d\t%d\t%d\t%d\t%d\t%d\t%d\t%d\n',
datalog(jj,:));
                    end

                    % close the text file
                    fclose(fid);
                end

                % tell the user what the filename is
                stext1 = strcat({'Saved as '}, strcat(title,date,'-',...
                    '*','_of_',num2str(length(start_indices)),'.txt') );
                set(savestatustext, 'Color', 'b', 'String', stext1,
'interpreter', 'none');

                deletedfile = 1;

            end

            pause(.1) % this just makes debugging easier

        end

    else
        first = 0;
        save first first
        run (thisone)
        % this runs the script a second time. this is really a hack, but it
        % solves a mysterious problem. for an unknown reason, the script
        % would not work until the second time it was run.

    end

    close all

else
    error('BB is not connected. Try restarting MATLAB')
end

delete first.mat trace_ind.mat;
```

11.3 Installation

1. The website to download the files: http://spot.colorado.edu/~alaa/neuro_lab/cu_wii.html
2. It is important to check your operating system and computer type. Go to: Control – security : check for Windows 7 and 64 bit – use MATLAB 32 bit version (side note for UW students - make sure you authorize MATLAB)
3. Go to Colorado website and download zip file: CU_WiiBB.zip
4. Then from download screen, right click on file and select: open containing file.

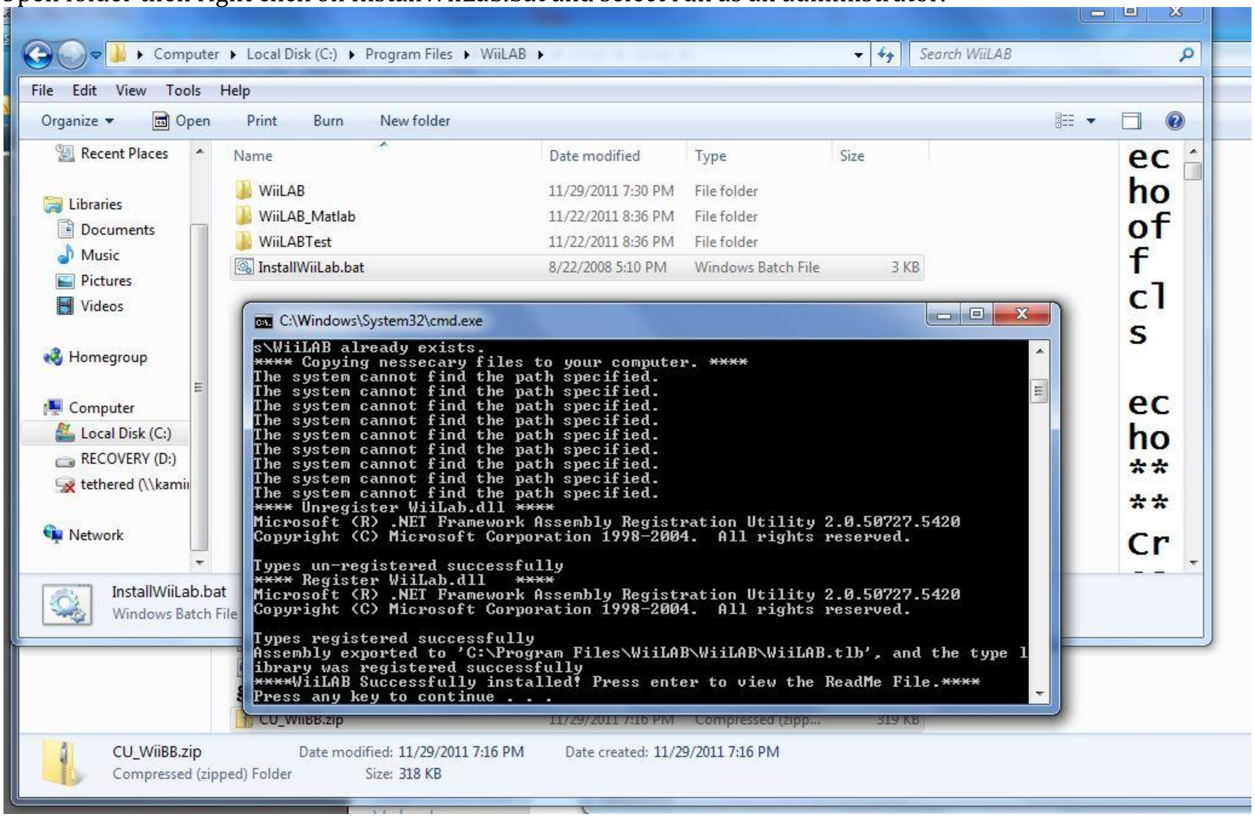


5. Right click on file, select: extract all. Then create new folder under my documents called Wii project.



6. Click Extract, 168 items should download
7. Copy Wii Lab Folder and paste it into C: drive and in program files

8. Open folder then right click on InstallWiiLab.bat and select run as an administrator.



9. Using a Bluetooth device, connect the Wii Balance Board. For our computer we have to go to control panel-devices and printers- and click add a device after hitting the red button by the batteries on the board
10. For our computer we had to go to add device, make sure the Wii balance board is not already a device. If it is delete, then go to add devices and see number 9.
11. Add pair without code.
12. Open MATLAB 2010 a 32 bit (Helen opened MATLAB 2011b).
13. At top of MATLAB, open fall (the …) to My documents-Wiiporjects-CU_Wii
14. Then open bbrecord_1_3, then run, as view data folder should appear (first ran this MATLAB had an error, reopened it and second time worked). After this runs the Wii Board should be connected aka blue light stops flashing and is constantly on
15. Initially click no to data file. Then it will ask you to calibrate the board, if possible use a precise weight we are using two 10 lb weights. The weight will need to be entered after you ok calibration the weight needs to be in kg.
16. It will ask you to get off board until program is loaded. An interface will appear as seen in Fig. 8.
17. Then you can step on and record data. (See Fig. 9)
18. Data is saved in Data folder – the code has the date and the time recorded

NOTE: We did NOT have to connect Wii Remote.