Restoring function after neuromuscular trauma: novel paradigms for enhancing volitional control of neural activity in untethered environments and vHAB, a gamified therapy platform

Tyler Libey

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2016

Reading Committee:

Eberhard E. Fetz, Chair

Eric Chudler

Steve Perlmutter

Program Authorized to Offer Degree:

Bioengineering

University of Washington

Abstract

Restoring function after neuromuscular trauma: novel paradigms for enhancing volitional control

of neural activity in untethered environments and vHAB, a gamified therapy platform

Tyler Libey

Chair of the Supervisory Committee:
Professor Eberhard E. Fetz, Ph.D.
Physiology & Biophysics, Bioengineering

Individuals suffering from neuromuscular traumas, such as stroke and spinal cord injury, face limited options in regaining motor function after injury. Emerging methods in brain-machine interfaces aim to assist individuals by directly connecting brain signals with assistive technologies. Human BMI studies are limited, leaving most studies to rely on models. Monkey studies are typically conducted in constrained booths under controlled conditions using brain areas that modulate with motor behavior. These approaches present challenges in translating to human populations with varying motor mappings, residual movement abilities, and cortical damage. These methods require years of work before seeing widespread adoption, leaving traditional methods as the leading treatment for most patients. Here, we present four novel approaches to these problems.

First, we present a rewarding brain-stimulation paradigm in unconstrained primates. To reward increases in cell activity, we sought sites in nucleus accumbens (NAc) whose stimulation

reinforced operant responding. Spikes occurring above baseline rates triggered electrical pulses to the reinforcement site. In-booth conditioning produced increases in neuron firing in 48 of 58 cells. In-cage conditioning produced significant increases in 21 of 33 sessions.

Second we developed a feeder and behavior monitoring system specifically designed for BMI applications. This system interfaces existing untethered recording equipment with a hub that controls a cage-mounted feeder. The behavior monitoring system uses a depth camera to provide real-time, easy to analyze, gross movement data streams. In a proof-of-concept experiment we demonstrated robust control and learning of the system over 14 sessions.

Next we describe the development of a gamified therapy system, vHAB, designed to enhance traditional recovery methods. vHAB consists of engaging games based on therapy tasks, automated analytics that describe patient progress, and carefully selected commodity hardware. Developing versions of the vHAB system were used at multiple pilot sites and two studies.

Finally, we relate initial promising results of the use of the prefrontal cortex as a non-motor area for BMI applications. Single neurons and local field potentials of the prefrontal cortex were conditioned in a constrained environment with three monkeys. Initial results were limited due to hardware complications, but monkeys showed signs of learning the task.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

and most importantly, Becky Schaefer deserves a great thanks for all of the work she does in running our lab. Her patience, kindness, and expertise often goes underappreciated, but none of this work would have been possible without her.

From the vHAB project, I must thank an additional set of individuals. vHAB was a collaborative project from the beginning and was inherently grander than any individual. Brian Mogen, Lars Crawford, and Dimitrios C. Gklezakos aided in the inception, development and testing in all the vHAB sections presented herein. Further, Tanner Dixon assisted in early development and Victoria Chambers provided insights into end-user populations. The Center for Sensorimotor Neural Engineering and Dr. Lise Johnson provided the environment for vHAB to be created. Deborah Harper and Josh Patrick provided daily support in the assurance of the project's success. Comotion and Ryan Buckmaster aided in establishing a clear path for vHAB to reach people in need. I would like to thank our group of project advisors, Dr. Jared Olson, Dr. Eric Fogel, and Timothy Rich, O.T. for their guidance and continued support.

I would like to thank my mom and grandma, who showed great patience through limited calls and visits over the past five years. Finally, I thank my girlfriend for her support and forgiveness through the stressful days and late nights. Her ability to inspire me yet keep me focused, is heavily responsible for this work.

DEDICATION

To Loki and Nymie.

"All of the true things I am about to tell you are shameless lies."

\- Kurt Vonnegut

Chapter 1. INTRODUCTION

1.1   NEUROMUSCULAR TRAUMA

Stroke, spinal cord injury, traumatic brain injuries, and neurodegenerative diseases such as Parkinson's, ALS, and cerebral palsy take away a person's ability to interact with the world. Collectively, these states and general traumas resulting in neuromuscular damage, will be referred to herein as neuromuscular traumas. While neuromuscular traumas have dramatically different causes, internal pathologies, external presentations, and treatment options, they have a one key thing in common: specific motor functions are deficient or absent entirely. This impacts the person's ability to do simple activities of daily living, such as going to the bathroom, drinking a cup of coffee, or holding hands with their loved ones. Multiple fields are constantly making progress towards better disease prevention, triage, and recovery techniques, but the quantity of these impacted individuals is continuously increasing. In this work, we aim to advance the intervention methods and treatment options available to patients as they recover from a neuromuscular trauma and try to regain motor function.

Many factors determine a patient's recovery and treatment options. First and foremost is the trauma's physiologic impact. Patients with intact motor pathways normally undergo physical and occupational therapy to regain motor function. These therapies suffer from a wide range of problems including low adherence, crude metrics of recovery, and low accessibility to persons in rural areas [1]. Traumas resulting in complete loss of neural pathways, such as a complete spinal cord injury or hemorrhagic stroke, may severely limit a patient's ability to participate in rehabilitation. These patients may someday be able to benefit from a brain-machine interface (BMI), a device designed to work around the broken neurological pathways by interacting with remaining neural signals. These devices can use these remaining signals to control assistive

devices or stimulate the body below the injury site to utilize the patient's own motor function. A complete review of BMI systems and their applications can be found in Green and Kalaska 2011 [2].

1.2   ADVANCING THE FIELD

Wide application of BMI technology is primarily limited by the invasiveness of the implants required to record robust neural signals. Further, most work studying emerging BMI systems, focus on a constrained animal model. Robust use of BMI systems will require the elimination of false positives when being used in simple activities of daily life. False positives can occur with a BMI system when the patient is not attending to the system, yet produces the control signals that activate the system. False positives in BMI systems are not often studied, mostly because the technology is not yet robust enough for long term human applications. However, as the technology advances, we must have models that allow for studying BMI systems in a natural environment.

Here we present two alternatives to the traditional constrained animal model. In Chapter 2 we explore the use of rewarding brain stimulation as an alternative to food reward. This method allowed for the development of a fully autonomous BMI system that could train an animal to modulate brain signals without being tethered to recording equipment. In Chapter 3, we describe the design of two modalities that can enhance the study of a freely behaving animal: 1) a wireless, cage mounted feeder that is controlled by the monkey's brain signals, and 2) an automated method for analyzing the monkey's movement while it is in the cage without the need for additional implants. These modalities were then used in a traditional BMI training paradigm, showing their efficacy in studying new BMI systems.

BMI systems traditionally utilize motor areas of the brain to record control signals, as modulations in these areas are typically related to motor actions. However, in the case of damage to these areas, control signals may not be available. In Chapter 5 we present preliminary results that use the prefrontal cortex to control a BMI system.

1.3   PROVIDING A SOLUTION NOW

BMI systems have tremendous potential for impacting the lives of patients with neuromuscular trauma. However, wide-scale adoption of these technologies may not happen for many years. In the meantime, one of a patient's best options is to undergo traditional therapy exercises. In Chapter 4, we present vHAB, a gamified therapy and assessment platform for patients with neuromuscular trauma. vHAB was developed to address key issues with traditional therapy, such as adherence and assessing recovery. First, we describe the vHAB system and how it was developed. Next, we present the preliminary results of vHAB being used by real patients in multiple environments. It is our hope that both vHAB and BMI systems will work in harmony someday to help patients regain the ability to interact with the world, and get back to doing the things they love.

1.4   REFERENCES

1.   Shaughnessy, M., B.M. Resnick, and R.F. Macko, *Testing a model of post-stroke exercise behavior.* Rehabil Nurs, 2006. **31**(1): p. 15-21.

2.   Green, A.M. and J.F. Kalaska, *Learning to move machines with the mind.* Trends Neurosci, 2011. **34**(2): p. 61-75.

Chapter 2. OPERANT CONDITIONING OF NEURAL ACTIVITY IN FREELY
BEHAVING MONKEYS WITH INTRACRANIAL REINFORCEMENT

2.1   INTRODUCTION

Volitional control of neural activity is critical for reliable and robust control of brain-machine interfaces (BMI). Indeed, BMIs can be seen as a form of neurofeedback that allows the user to see the consequences of neural activity and change that activity to optimize control of the external device [1]. However, BMI control is only a subset of the possible range of volitional control of neural activity that can be explored directly with operant conditioning. Traditional techniques for operant conditioning of behavior in monkeys have limited the scope of investigation to specific tasks, using food reward and visual feedback delivered in a training booth. Constrained, task-related movements differ from natural behavior, and correlations between neural activity and movement established under particular task conditions may not hold under non-task conditions [2] [3] [4]. The vast majority of nonhuman primate research involving trained behavior has employed rewards in the form of food or water [5-7], further limiting the circumstances in which neural activity was explored. Here we present a novel mechanism for rewarding neural activity during natural behavior using a closed-loop system delivering neurally contingent brain stimulation reward (BSR).

Olds & Milner [8] [9] demonstrated that rats would press bars and navigate mazes for BSR, which could reinforce operant responding as effectively as more conventional food and liquid rewards. Later work by David Hiatt attempted to condition increases in single-unit activity using burst-triggered BSR in rats [10]. As candidates for conditioning he sought cells in hippocampus, cerebellum, midbrain and superior colliculus that were not movement related.

Recently, BSR was used to elicit rate increases in prefrontal cortex neurons of freely behaving rats [11]. The ability of freely moving rats to differentially control small groups of cortical neurons was demonstrated with food reward and continuous auditory feedback [12].

Several studies have explored the efficacy of BSR in non-human primates. In a freely behaving chimpanzee Delgado et al deployed wireless closed-loop stimulation of reticular formation sites contingent on oscillations in amygdala field potentials [13]. The triggering neural oscillations disappeared after a day of activity-dependent stimulation, indicating that this form of stimulation was aversive. Later work showed that monkeys will perform simple bar-press tasks for BSR in several structures, including the orbitofrontal cortex, lateral hypothalamus, amygdala, medio-dorsal nucleus of the thalamus and nucleus accumbens [14-18].

An interesting open question is whether monkeys can learn to control activity of single neurons with intracranial electrical stimulation as the sole source of reinforcement. This would allow operant conditioning to be performed during prolonged periods of free behavior, providing extended time and behavioral range to learn volitional control of neural response patterns. BSR would enable delivery of reinforcement that is temporally more precise than food or water rewards, and less disruptive of ongoing behavior. In this study we sought to operantly condition activity of motor cortex neurons and electromyographic (EMG) activity of proximal limb muscles, using activity-contingent BSR at sites confirmed to sustain behavior. To compare the effects of the environment, we conditioned these activities both in the training booth and as the monkeys moved freely about their home cage.

2.2   METHODS

2.2.A   *Subjects and Training*

We used 3 male Macaca nemestrina monkeys P, D and J (4-6 years old, weight 6.0, 5.6 and 4.0 kg). All surgical, training and handling procedures were approved by the University of Washington Institutional Animal Care and Use Committee.

Prior to surgeries, monkeys were trained to perform a 1D center-out force-target-tracking (FTT) task in which isometric wrist torque controlled the position of a cursor on a screen. When the cursor entered a target and remained inside for the required time (1 second or less) a fruit sauce reward signaled completion of the trial. Target placement on the screen determined the required direction and magnitude of flexion or extension torque about the wrist. Peripheral targets were presented in random order with equal probability. Training was complete when monkeys moved directly from center to each target, and held it inside for at least 1 second. During experiments, the FTT task was performed daily to elicit task-related cell firing in motor cortex.

2.2.B   *Surgery and Implantation*

Cranial microwires and arrays of up to 16 cannulae were implanted in each monkey. The microwire arrays [19] were positioned to advance along layer V in the caudal bank of the precentral gyrus, where somata of many force-correlated cells (including corticomotoneuronal (CM) cells) have been identified [20], [21]. The cannulae were positioned stereotaxically to guide subsequent stimulating electrodes to the nucleus accumbens (NAc). Cannula-length stylets were placed in all guide tubes and the protruding surface of the array was sealed in silastic. The open space between craniotomy and array was packed with antibiotic-infused gelfoam. An

acrylic base around the implantation site and surrounding cranial screws formed the base for a cylindrical titanium chamber enclosing the microwire cannula arrays and Neurochip [22]. Rhodes SNEX-100 concentric bipolar electrodes were inserted subsequently into the cannulae after cold-sterilization of the chamber interior and electrodes with cidex.

To identify potential intracranial reinforcement electrode implant sites, we co-registered a magnetic resonance image (MRI) and digitized brain atlas data [23] to determine the stereotaxic coordinates of prospective midbrain reinforcement loci (Figure 2.1). Monkey P underwent MRI scanning prior to surgical implantation. Monkeys D and J were of similar size as atlas subjects, so MRIs were not deemed necessary. We selected coronal image slices located +3 mm rostral from the anterior commissure that contained the largest cross-section of the Nucleus Accumbens (NAc). Stereotaxic coordinates of the target locus were measured relative to medial-lateral center and ear-bar zero. A straight-line diagonal path to the target locus (center of NAc) that was 15 degrees lateral right with respect to the ventral dorsal axis in the right hemisphere avoided major blood vessels and regions governing autonomic function. To address the possibility of positioning error of entry sites, we implanted an array of 16 parallel cannulae spaced 1 to 1.5 mm apart in a 10×10 mm grid centered at the best point of entry. Thus, in cases of slight angle misalignment or entry location, the target locus might still be reachable by an electrode inserted in one of the neighboring cannulae. Following implantation, unused cannulae were occluded with stylets and sealed with silastic to block potential cranial infection.

Figure 2.1. Co-registration of cranial X-ray, MRI and brain atlas images. A coronal map of the M. nemestrina brain was morphed and superimposed on an MRI of Monkey P's brain. Both are positioned over an X-ray image showing the chamber and housed hardware. NAc shown in red.

In monkey P, in addition to cranial implant procedures, we implanted pairs of EMG wires in three proximal muscles of the monkey's right arm: the biceps brachii, triceps brachii and lateral deltoid. Muscle activity was first operantly conditioned to verify efficacy of BSR in free behavior. The EMG wires were routed subcutaneously around the shoulder, up the back and neck and terminated in connectors located inside the cranial chamber for signal processing by the Neurochip.

2.2.C   *Verification of brain stimulation reward*

To identify intracranial brain sites whose stimulation sustains operant responding, we compared response rates occurring during reinforcement (R) and visual feedback-only (FO)

blocks in a FTT task. During R blocks, each completed flexion or extension target hold triggered BSR. In FO blocks no stimulation was delivered, regardless of task performance, but the FTT task could be performed. R and FO blocks were interleaved with non-reinforcing (NR) blocks in which neither feedback nor reward were available. Stimulation consisted of trains of symmetric biphasic square-wave current pulses. A low-frequency tone during R blocks served as a discriminatory stimulus (in addition to FTT task auditory cues for target acquisition). Candidate sites were considered to be "positively reinforcing" when monkeys performed wrist FTT at a significantly greater rate during R blocks than during FO blocks.

2.2.D *Rate-contingent spike-triggered (RCST) stimulation*

Validated BSR sites were used to operantly condition cortical cell and muscle activity in two different settings: a traditional in-booth setting using rack mounted equipment for recording and stimulation and an in-cage setting using the Neurochip system (Figure 2.2A, B). The Neurochip (NC) employs an autonomous, battery-powered computer chip programmed to detect and reward cell and muscle activity while monkeys moved freely about their cages [24]. It discriminated cortical cell or EMG activity patterns using dual time-amplitude window discrimination and delivered stimuli contingent on discriminated events in real time. The Neurochip2- HV is a second-generation version with improved capabilities for storage, processing and stimulus range [22]. Alternating R/NR reinforcement schedules were used to distinguish the effects of BSR in the operant conditioning paradigm. FO blocks were not used during these experiments. The in-booth experiments utilized audio and visual feedback to distinguish between the periods, whereas the in-cage experiments relied solely on audio feedback. The in-booth experiments lasted between 1 and 6 hours, while the in-cage free-behavior sessions lasted considerably longer: 3-20 hours.

9
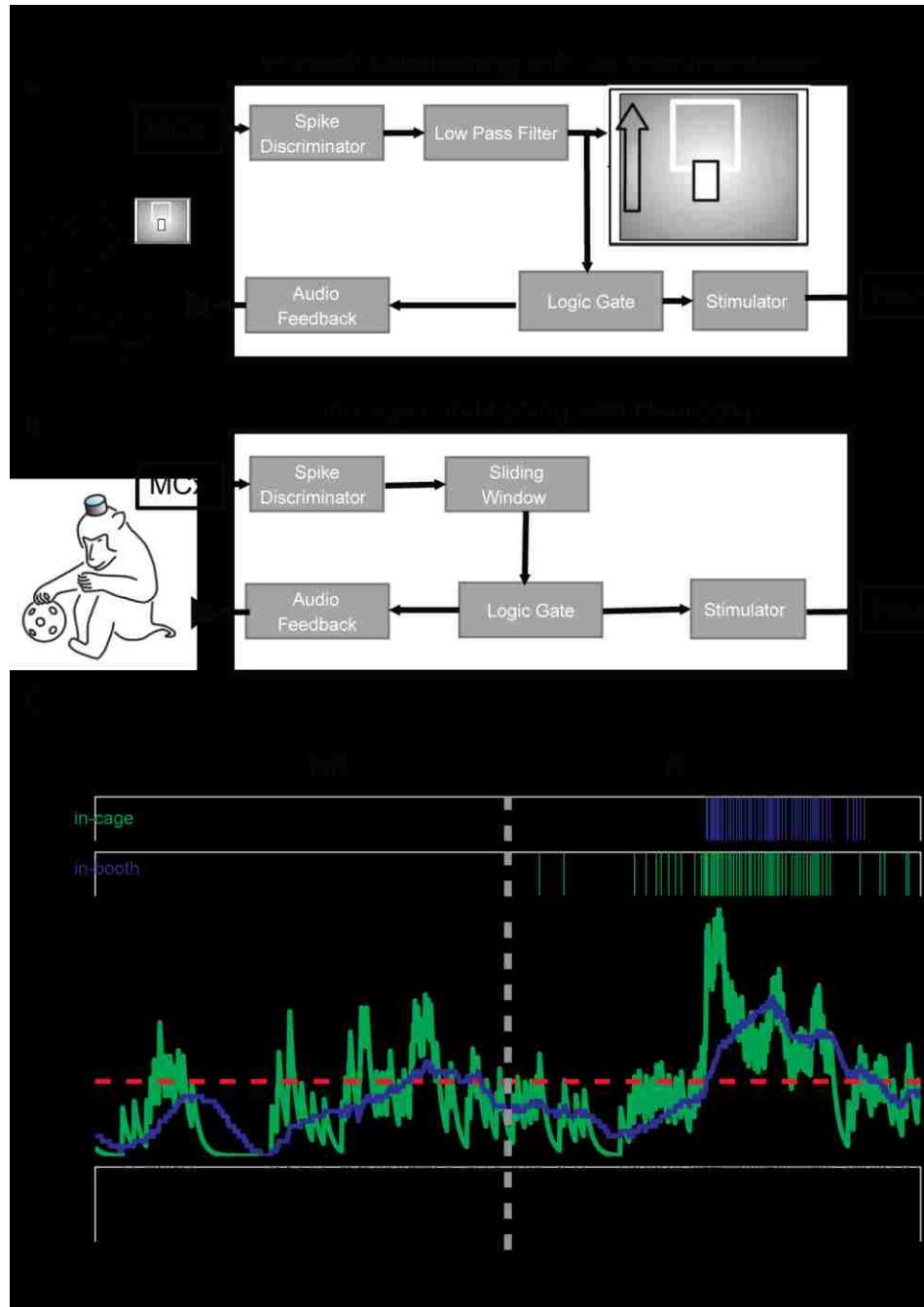
Figure 2.2. Experimental conditions. **A.** Schematic of unit conditioning in booth. Activity of motor

cortex (MC) cell generated pulses that were low-pass filtered and controlled cursor position on a screen.

A logic gate triggered pulses when firing rate exceeded a threshold (green traces in C). Pulses stimulated

nucleus accumbens (NAc) and auditory feedback tones.   **B.** Schematic of unit conditioning during free

behavior in cage. The Neurochip was programmed to detect spikes and compile a running average of

rate; when this exceeded threshold, pulses triggered stimuli to NAc (blue traces in C) and auditory clicks. **C.** Conversion of NC spike events (bottom) to NAc stimuli (top) as firing rates exceeded threshold (red dashes) for in-booth and in-cage conditioning (green and blue, respectively)

During alternating R/NR conditioning, we approximated instantaneous firing rate in real-time using two methods, depending on the environment (Figure 2.2). For most in-booth sessions, spikes were discriminated with a time-amplitude window and each spike event triggered a 1 ms wide square pulse. The pulse train output (Fig. 2C bottom) was low-pass filtered ($\tau = 50$ ms) and amplified using an analog leaky integrator. These operations produced a continuous signal (Figure 2.2C, green trace) that controlled cursor movements on the display in front of the animal, providing visual feedback of rate relative to target (Figure 2.2A). When the activity-controlled cursor entered the target, all subsequent in-target spike events triggered stimulation of the reinforcement site. Stimulation events were often also used to trigger auditory clicks. We initially set the target position just above baseline firing rate, and gradually raised its position over the course of conditioning to elicit higher spike rates. Targets were presented only during R periods of the alternating R/NR task.

For in-cage sessions (Figure 2.2B), we pre-programmed the NC to perform a real-time sliding window operation to estimate instantaneous spike rate (Figure 2.2.C, blue trace). The NC counted the number of spike events within a 500 ms wide moving window that advanced every 10 ms. The NC delivered spike-triggered stimuli on spike events that occurred when this estimated rate exceeded a threshold frequency (Figure 2.2C, red dashed line). Threshold was determined from force target-tracking or in-booth R/NR task response averages that revealed baseline and maximum firing rates of the particular cell. Typically, in-cage stimulation thresholds were set at 75% of the observed maximum firing rate of the candidate cell. In later

11

sessions, the NC governed operant conditioning sessions both in the training booth and cage, to directly compare the effects of environment.

Prior to conditioning, durations of alternating R and NR periods were randomly selected, with replacement, from uniform distributions spanning 1 to 2 min for R and 3 to 5 min for NR. We employed random period durations, within limits, to reduce the monkeys' ability to anticipate transitions in the reinforcement schedule.

2.2.E *Data Analyses*

2.2.E.1   *Time series analysis detects acquisition of the alternating R/NR task*

To determine whether firing rates during R and NR periods were significantly different, we computed serial correlation (i.e., auto-correlation) and von Neumann ratio test statistics on the average rates during R and NR [25]. We also analyzed the changes in rates around the transitions between R and NR periods, using peri-transition snippets of spike trains extracted 75 seconds prior to and 75 seconds following each transition. The snippets were combined into peri-event spike histograms (bin width = 50 ms) and consolidated into a single dense train that was convolved with a Gaussian kernel to approximate instantaneous spike rate around the transition. Changes in firing rate were identified as being significantly different in comparison to shuffled spike rates.

2.2.E.2   *Confidence intervals for time-averaged rates*

To compute confidence intervals for time averaged means, we developed a non-parametric bootstrap method based on the Poissonian property of independent interspike intervals (ISIs) [26]. First, for each R and NR period during conditioning, we calculated $n - 1$

ISIs from the list of n spike events that occurred and then sorted ISI values in ascending order. Second, from the ordered set, ISIs were randomly drawn with replacement until their cumulative sum just surpassed the duration of the actual interval containing the observed spike events. Third, the number of ISIs comprising the cumulative sum was divided by the period duration to provide a simulated time average rate value. Repeating steps one through three a total of N times generated a bootstrap distribution of time-averaged rate values from which confidence intervals were determined. To validate structure in spike histograms, we quantitatively evaluated their statistical significance by computing time-varying rate estimates from event-relative spike timestamps convolving an appropriate kernel [25]. We then computed confidence intervals about these point density estimates of instantaneous firing rate [27]. To gauge the validity of spike rate approximates, and to detect statistically significant changes, we computed 95% confidence intervals about kernel-smoothed estimates of firing rate from peri-event spike trains.

2.2.E.3   Spike event shuffling as a control measure

To evaluate whether modulations in rate traces were non-spurious, robust and recurring, we contrasted traces from the observed sequence of spike events against a collection of rate traces computed after those spike events had been shuffled. Rate traces from shuffled sets were computed using identical procedures and parameters, as was performed on the original sets so that traces from observed and shuffled trains could be directly compared. Domains in which the observed set diverged outside the confidence interval of the shuffled set indicated that such features in peri-transition spike activity could not be explained as random fluctuation, and therefore arose as a consequence of the conditioning paradigm. Reshuffling the observed set of spike trains $R = 199$ times generated a distribution of shuffled train rate traces from which average and confidence intervals could be computed.

13

2.3   RESULTS

2.3.A   *Accumbens stimulation reinforces target-tracking behavior*

We tested the efficacy of candidate reinforcing sites by measuring the monkeys' rate of responding in a manual force target-tracking (FTT) task on which they had been trained with applesauce reward. At effective sites trains of stimuli (25 1-mA-pulses at 50 Hz) delivered upon completion of 1-second force holds reinforced further responding. As shown in Figure 2.3A, the monkey responded at regular rates during reinforcement (R) periods when target completions triggered trains of brain stimulation reward (BSR). Response rates during R periods were significantly higher ($p < 0.001$) compared to interleaved periods during which only feedback was presented and no stimulation was delivered (FO-periods). At the onset of R periods response rates often returned quickly to those of the previous R-period. As a comparison, FTT task response rates for applesauce reward typically ranged between 10 and 13 responses per minute for the three monkeys.
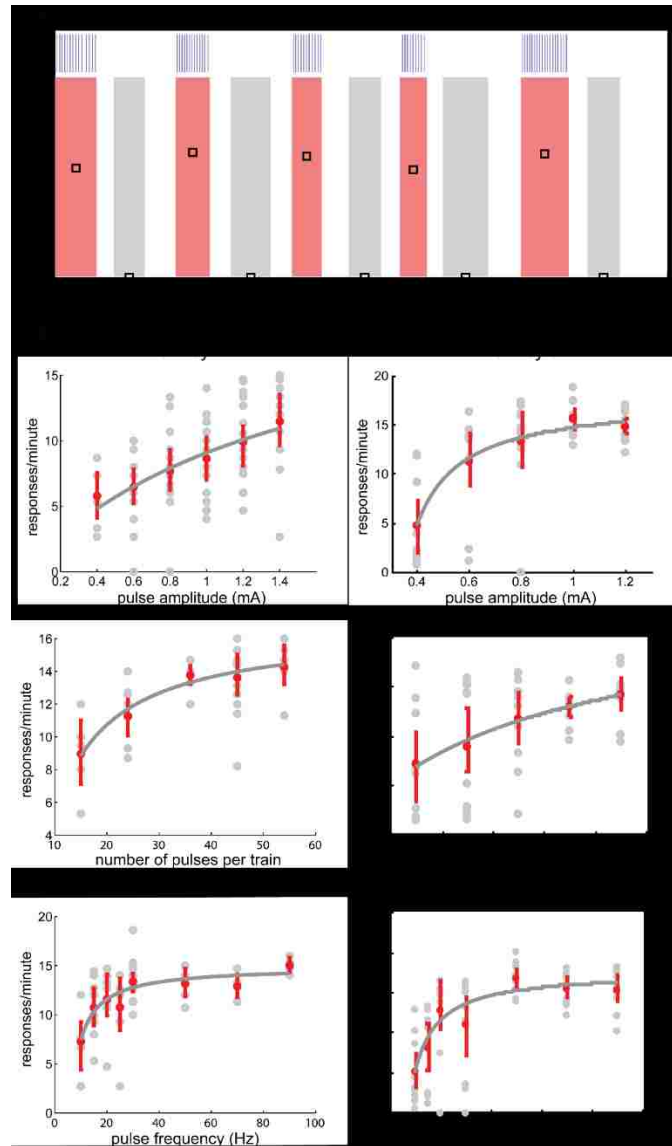
Figure 2.3. NAc stimulation reinforces target-tracking behavior. **A**. Average response rates (black squares) during reinforced (R, pink) and feedback only (FO, gray) periods of the wrist force target-tracking task. Clear intervals are non-reinforced periods. Blue ticks (top) mark trial completions. During R periods, each completed flexion or extension hold triggered behaviorally-reinforcing brain stimulation (BSR). **B.** Rate of target-tracking responding increases monotonically as a function of increasing BSR intensity**.** Data from monkey P (left column) and monkey D (right). Force target-tracking response rates (gray circles) and response rate means and standard errors (red) are plotted as functions of one varied

stimulation parameter: current intensity (top), number of pulses (middle) or frequency (bottom). In each

case the other two parameters remained fixed, at 1 mA, 50 Hz or 25 pulses per train. Gray curves depict

response rates predicted by regression-fitted Law of Effect model using parameters in Table 2.1.

2.3.B   *Target-tracking rates as function of BSR parameters*

In order to determine appropriate stimulation parameters for conditioning cortical cell

activity, we documented rates of target-tracking responses for different values of three

parameters: current intensity, pulse frequency and number of pulses per stimulus train. Each of

these parameters was varied while the other two remained fixed. Fixed values were 1 mA for

current intensity, 50 Hz for pulse frequency and 25 pulses per train. For each varied parameter,

the values in the desired range were repeated 10 times, delivered in a randomized sequence, to

eliminate possible "history effects".

Figure 2.3B depicts target-tracking response rates as a function of each stimulus

parameter in monkeys P and D. In all cases, the response rates R as a function of the tested

stimulation parameter r were well characterized by nonlinear-regression-fitted curves of the Law

of Effect model:

$$R(r) = \frac{k(r - r_{th})}{(r - r_{th}) + r_e} \qquad\qquad \text{Eq. 2.1}$$

where r_{th} is the threshold level, or lowest value at which the stimulus parameter supported self-

stimulation, and r_e represents the aggregate reinforcement for all non-operant responses [28].

Table 2.1 summarizes fit statistics for each of the plots. The response curves indicate that ~80-

90% of maximal responding (horizontal asymptote of each plot) occurred for stimulation

parameters 1 mA and 50 Hz.

16

In subsequent cell and muscle conditioning experiments pulse amplitude was set to 1 mA. Bursts of elevated spike rates triggered pulse trains at frequencies approaching 50 Hz. For slowly firing cells (e.g. <10 Hz), multiple stimulus pulses (delivered at 50 Hz) were triggered for each RCST stimulus event.

Table 2.1 Single response Law of Effect model fit parameters and statistics. Parameters fit to single response Law of Effect model (Eq. 2.1.). The fit parameters are: k, the maximal response rate asymptote (responses/min.), r_e, the aggregate reinforcement for all non-operant responses, and r_{th}, the threshold level, or lowest value at which the varied stimulus parameter supported self-stimulation. MSE: mean squared error of the model fit using non-linear regression.

Varied parameter	Monkey	k	r_e	r_{th}	MSE
Pulse amplitude (mA)	P	22.45	1.47	0.00	0.15
	D	17.64	0.13	0.35	0.31
No. pulses per train	P	16.65	7.31	6.75	0.16
	D	23.72	37.00	0.00	0.04
Pulse frequency (Hz)	P	14.89	4.00	6.05	0.35
	D	17.74	6.86	7.32	1.46

2.3.C *Muscle activity reinforced during free behavior with BSR*

To confirm the efficacy of BSR sites during free behavior we tested in-cage conditioning using EMG activity of upper limb muscles as the operant. The time-amplitude window discriminator detected biphasic waveforms in the multi-unit EMG signal (Figure 2.4A right) and generated acceptance pulses whose frequency increased with intensity of muscle contraction. During reinforcement periods the mean rates of biceps EMG-generated pulses were significantly larger than during intervening non-reinforced periods (Figure 2.4A left), and the monkey was observed to flex his arm during reinforcement periods. With biceps conditioning these

17

differences were maintained for up to 20 hours of conditioning. Significant differences were also seen with triceps conditioning (Figure 2.4A).

The transitions between periods of reinforcement (R) and non-reinforcement (NR) showed further evidence of learning to perform the biceps responses. Separate averages around these transitions for the initial, middle and final third of the session (Figure 2.4B) show progressive changes in responding over the course of the conditioning session. For the NR-to-R transitions rate increases were comparatively low and gradual during the first 6 hours, moderate during the middle period, and greatest and fastest during the last 6 hours. Interestingly, the R-to-NR transitions exhibited a brief increase in responding after the cessation of reinforcement for the first and middle thirds of conditioning (arrow), and no such peak in the last third. Since the monkey had no discriminative stimulus to distinguish R and NR, this behavior is consistent with initial attempts to sustain reinforcement that drop out after sufficient experience with the transition. The raster plots in Figure 2.4C show color-coded rates for the individual transitions and their variability in more detail. These data confirm that BSR can effectively reinforce an operant, muscle activity, for long periods of time during free behavior.
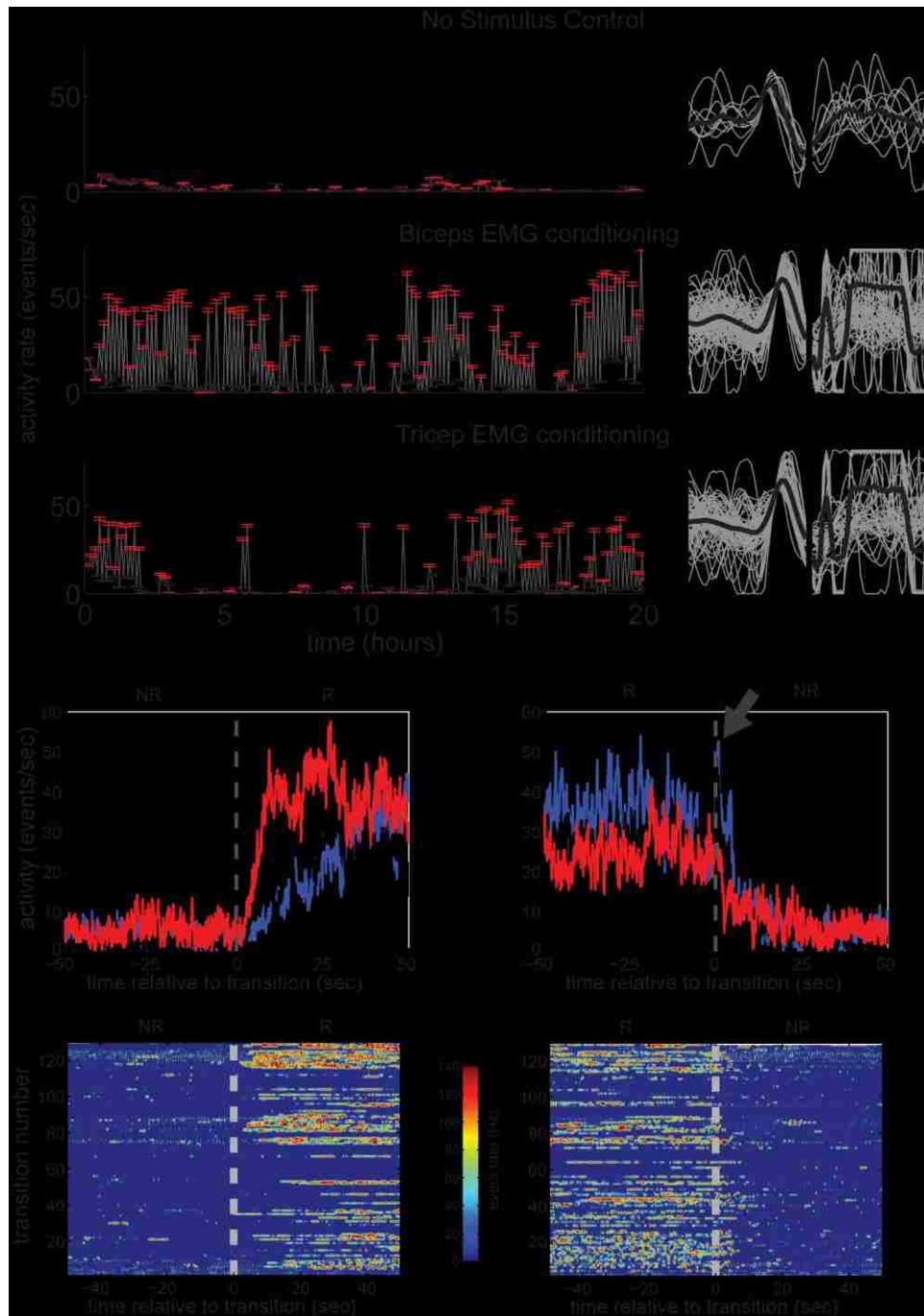
Figure 2.4. Muscle activity conditioning with NAc stimulation. A. NAc stimulation reinforces muscle activity in-cage. Top: Baseline session in which no stimulation was delivered during either R (red) or NR (black) periods. Each data-point indicates mean EMG activity over 5 minutes, and surrounding whiskers mark standard error boundaries. Middle and Bottom: EMG-contingent current pulses delivered to

accumbens during R- periods. Right: EMG potentials that triggered stimulation over each 20-hour session (gray), and their averages (black). Accepted biphasic patterns are followed by artifacts from triggered stimuli. **B.** Averages of biceps muscle activity surrounding NR-R schedule transitions (left) and R-NR transitions (right), shown separately for first third (blue) middle third (black) and last third (red) of the session. Shortly after transition in the blue and black traces, there is a brief transient increase in activity (black arrow) that is not present in the red trace, indicating that the monkey became more familiar with the task over time. **C.** Peri-transition biceps activity during the 20-hour conditioning session, showing NR-R (left) and R-NR (right) transitions. Ordinates count the transitions over the course of the 20-hour session. The color indicates the rate of biceps EMG activity (see scale).

2.3.D *Overview of cell conditioning sessions*

Table 2.2 summarizes results from all sessions in which cortical cell activity was conditioned with BSR for the three monkeys, categorized by environment: booth or cage. Given sufficient stability and unit isolation, we often conditioned the same cell over repeated sessions. The appropriate conditioning procedures were determined in about 70% of the total in-cage attempts. Attempts were deemed invalid for one or more of the following reasons: 1) Neurochip malfunction, 2) loss of action potential isolation, and 3) improper conditioning parameters.

2.3.E *Spike-triggered NAc stimulation reinforces increased motor cortex cell activity*

During reinforcement periods the monkeys received spike-triggered BSR when the instantaneous spike rate exceeded a pre-determined threshold. Table 2.3 summarizes conditioning parameters used for each of the sessions.

Table 2.2. Summary of all effects from cortical spike-triggered BSR conditioning attempts. Key: "+" symbols denote statistically significant increases during R-periods compared to NR-periods, "0"s mark cases where no significant changes were observed between R and NR activities and "−" indicate cases when NR-period spike rates were significantly greater than R-period rates. Often, the same cell underwent conditioning in multiple sessions. In this table, a given cell was tallied as generating a positive effect though it may have produced null or negative effects in other conditioning attempts. Cells that showed null (0) and negative (−) effects over sessions were categorized as null.

		in-booth				in-cage			
		effect type			% success	effect type			% success
		+	0	-		+	0	-	
Monkey P	cells:	7	2	1	70.0	--	--	--	--
	sessions:	7	4	4	46.7	--	--	--	--
Monkey D	cells:	21	3	2	80.8	3	1	0	75.0
	sessions:	38	12	4	70.4	3	2	0	60.0
Monkey J	cells:	20	2	0	90.9	6	2	1	66.7
	sessions:	42	14	1	73.7	18	9	1	64.3
Total	cells:	48	7	3	82.8	9	3	1	69.2
	sessions:	87	30	9	69.0	21	11	1	63.6

Figure 2.5A-C shows average motor cortex neuron spike rates during three representative conditioning sessions performed in the training booth. Robust increases in firing rates were observed during R-periods as compared to the intervening NR-periods, showing successful acquisition of the neural operant. In all plots, rates were significantly greater in R than NR periods as indicated by predominantly non-overlapping confidence intervals. Figure 2.5D shows an in-cage conditioning session in which monkey J moved freely about his home cage and the Neurochip2 delivered RCST accumbens stimulation in an alternating R/NR schedule over 8 hours. Average firing rates were statistically greater in R-period compared to NR periods;

however these differences were smaller than those observed for typical in-booth-conditioning sessions.

Table 2.3. Summary of conditioning parameters used for each example conditioning session. In the table above, we list session keys (in the format: Monkey:Session-Cell), dates, conditioning environments, equipment used, schedule durations, rate thresholds , trigger-to-stimulus pulse ratios, and sensory feedback used and text figures for each of the example sessions depicted in the results. In all sessions, BSR stimuli were delivered in 0.2 ms wide, biphasic square pulses at 1 mA intensity. The conditioning equipment used are the NC1: Neurochip, Rack: rack-mounted instrumentation, and NC2-HV: the High Voltage Neurochip2.The feedback types are: VC: visual cues in the form of a rate-guided computer cursor, AC1: auditory cues in the form of spike-triggered clicks produced by rack-mounted equipment, AC2: clicks on BSR pulse delivery generated by the Neurochip2-HV.

Session Key	Date	Environ-ment	Conditioning equipment	R-period duration (min)	NR-period duration (min)	Rate threshold (Hz)	Trigger-to-stim. pulse Ratio	Sensory Feedback	Figures
P:1-Bicep	11/24/2007	cage	NC1	5	5	none	1:1	none	4
P:2-Tricep	11/25/2007	cage	NC1	5	5	none	1:1	none	4
D:1-1	5/24/2010	booth	rack	1-2	3-5	20	1:1	VC, AC1	5A,9
D:2-2	9/3/2010	booth	rack	1-2	3-5	15	1:3	VC, AC1	5B,6A,9
J:1-1	11/10/2012	booth	rack	1-2	3-5	30	1:1	VC, AC1	5C,6B,9
J:2-1	11/28/2012	cage	NC2-HV	1	5	50	1:1	AC2	5D,6C,7,9
J:3-1	2/22/2013	booth	NC2-HV	2	5	60	1:1	AC2	8
J:3-1	2/22/2013	cage	NC2-HV	2	5	60	1:1	AC2	8
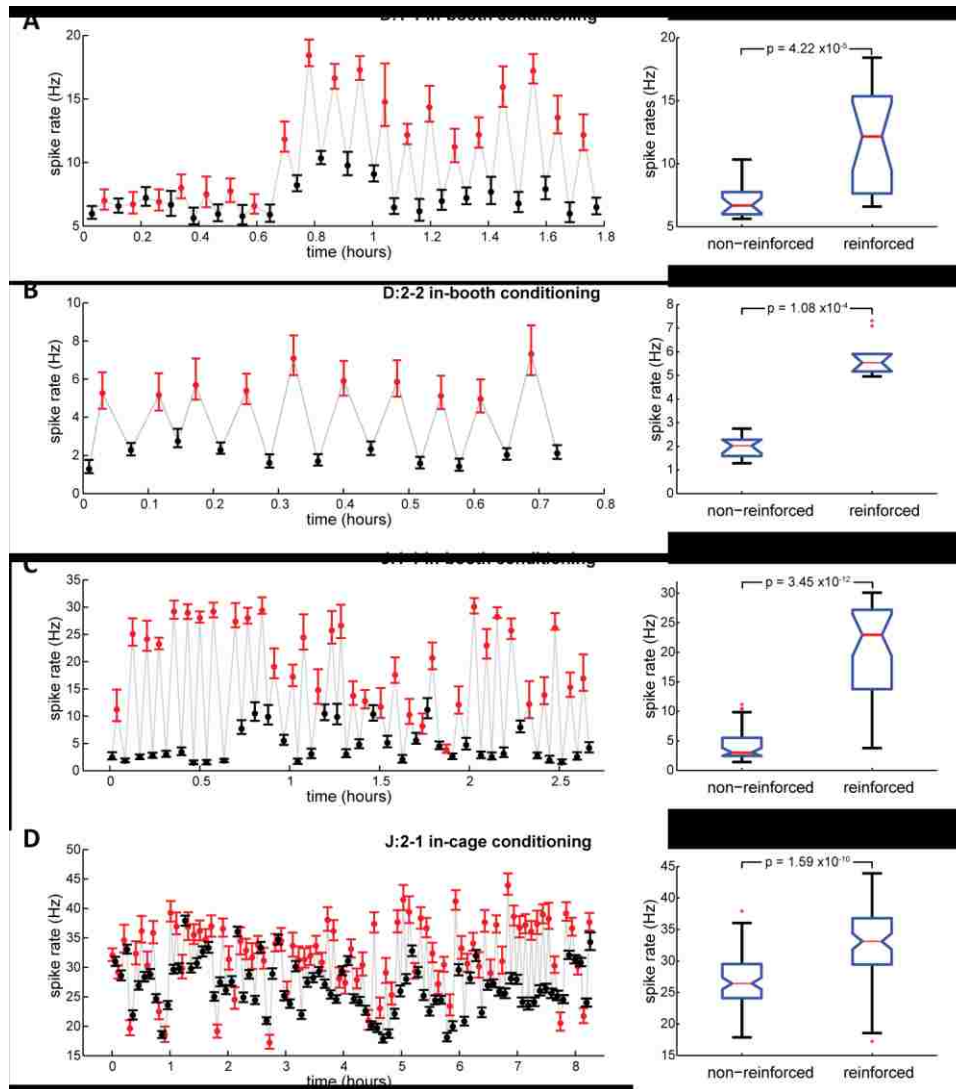
Figure 2.5. Response rates for cortical neurons during in-booth conditioning (A-C) and in-cage conditioning (D). A and B. initial and later sessions with monkey D. C. in-booth session with monkey J. D. in-cage session during free behavior with monkey J. Left: Points mark average rates during reinforced periods (red) when RCST stimulation was available and during non-reinforced periods (black). Bars denote 95% confidence intervals. Right. Statistics of cortical cell spike rate during R and NR periods shown as box plots. In each box, the central red line marks the distribution median, and blue box extremities depict upper (75%) and lower (25%) quartiles. Red crosses plot rates with values outside of

whisker boundaries. Notch height shows approximate limits of confidence intervals about their median at the 5% significance level.

The alternating rate patterns described above give rise to robust, statistically significant time series measures, namely serial correlation and von Neumann ratio [25]. Alternating rates are obvious from inspection of in-booth conditioning sessions, but are less apparent for the in-cage session. Serial correlation and von Neumann ratios measure pattern in time series from which statistical significance can be approximated through randomization and Monte Carlo approximation methods. These analysis techniques confirm significant patterns in these series of time-averages that might otherwise not be evident [25]

Box plots (Figure 2.5, right) summarize distributions of reinforced and non-reinforced time averages across each session. For both monkeys, NR distributions have lower medians and were less variable than the R group distributions. These differences are statistically significant in all four examples as assessed by Kruskal-Wallis and unpaired Student's t-tests.

2.3.F *Peri-transition activity patterns*

For further insight into behavioral mechanisms we documented the changes in firing rates associated with transitions between R and NR periods. Figure 2.6 shows histograms and smoothed rate traces of neuron spike trains during NR-R and R-NR transitions. For comparison, the overall average rates and 95% confidence intervals are illustrated by grey solid and dashed lines, respectively. Statistically significant deviations from chance occur where the red rate trace exceeds the "chance band." Two sets of peri-transition averages, one for monkey J and one for monkey D exemplify robust rate increases observed across NR-R transitions while the animals underwent RCST stimulation conditioning while under restraint in the training booth. In session

24

J1-1 (Fig. 6B), monkey J produced a four-fold increase in motor cortex cell spike rate and kept rates elevated, on average, for the full duration of reinforcement. During in-booth sessions, activity peaked early, usually within 10 seconds following the NR-R schedule transition, and then decayed over the remainder of each reinforced period. During in-cage conditioning activity peaked later in the R period. Spike activity dropped quickly following R-NR transitions both in-booth and in-cage. However, as shown in Figure 2.6C, NR spike activity tended to be more variable in the cage than in the booth.

Instrumentation in the training booth allowed us to record wrist torque during unit conditioning. In all examples, motor cortex neurons modulated their activity during dynamic and/or static phases of the force target tracking task. Peri-transition averages of the isometric torque signals show increased torques during R periods that accompanied spike rate increases and corresponding reduction of torque generation during NR with lower cortical spike rates (Figure 2.6A, B). Consistent with the parallel analysis of sequential time-averages (Figure 2.5, left), the increases in spike rates across NR-R transitions were greater in cells conditioned in-booth than in cells conditioned in-cage.
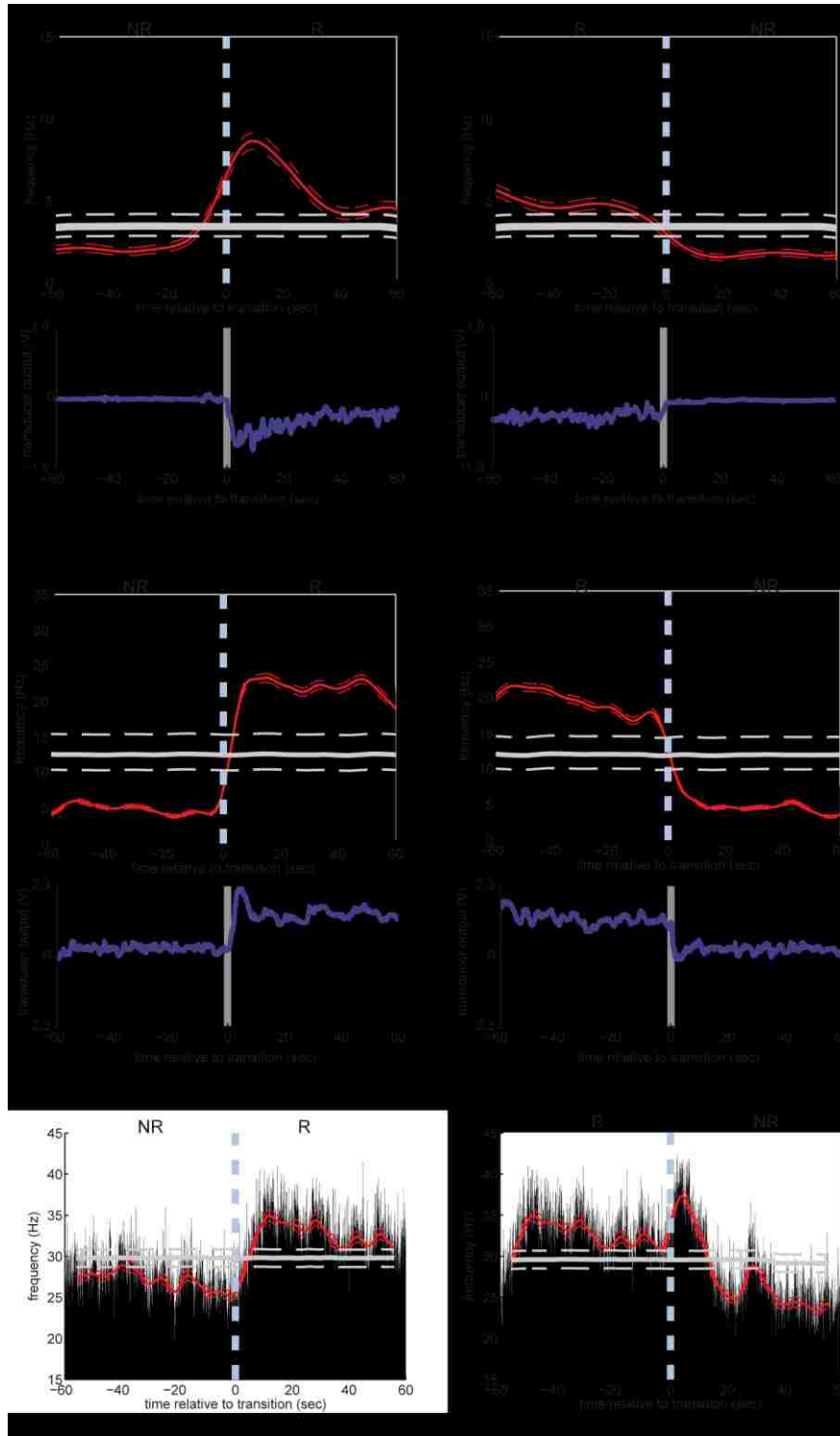
Figure 2.6. Peri-transition histograms of neuron spike activity during in-booth (A, B) and in-cage (C) conditioning. For each session, spikes occurring during two-minute intervals straddling NR-R

transitions (left) and R-NR transitions (right) are pooled and binned into histograms above. Light-blue dashed vertical lines at t = 0 mark onset and offset of activity-dependent BSR. The point-density estimate of spike rate (thick red line) and its 95% confidence band limits (dashed red lines) overlay corresponding histograms. Horizontal gray lines show averages (solid) of sweeps, and surrounding 95% confidence intervals (dashed) after spike shuffling. Red line shows smoothed firing rates where the smoothing kernel width was determined by overall spike rate [27]. The slow early rise in A is a result of low NR baseline rates followed by an abrupt increase. Bottom: averages of flexion-extension torques recorded concurrently. Depicted here are peri-transition activities from sessions: A. D2-2 and B. J: 1-1. C. Peri-transition histograms of spike activity conditioned in-cage during session J: 2-1. The NC2 delivered rate-contingent, spike-triggered BSR in the alternating R/NR schedule.

2.3.G *Rate changes of motor cortex cell spike activity conditioned in-cage*

As with EMG activity (Figure 2.4), for in-cage unit-conditioning the relative increases in BSR-reinforced spike activity were smallest, compared to NR-period activity, during the first third and greatest during the final third of the session (Figure 2.7). A transient increase in spike rate also followed R-NR transitions, when high-frequency spike bursts no longer triggered NAc stimulation. A similar post-extinction burst effect was seen in R-NR peri-transition averages of in-cage conditioned biceps activity (Figure 2.4B) of the first and middle third session averages. Unlike muscle conditioning however, the extinction burst in spike activity, though markedly reduced, did not completely disappear during the final third of the unit-conditioning session.
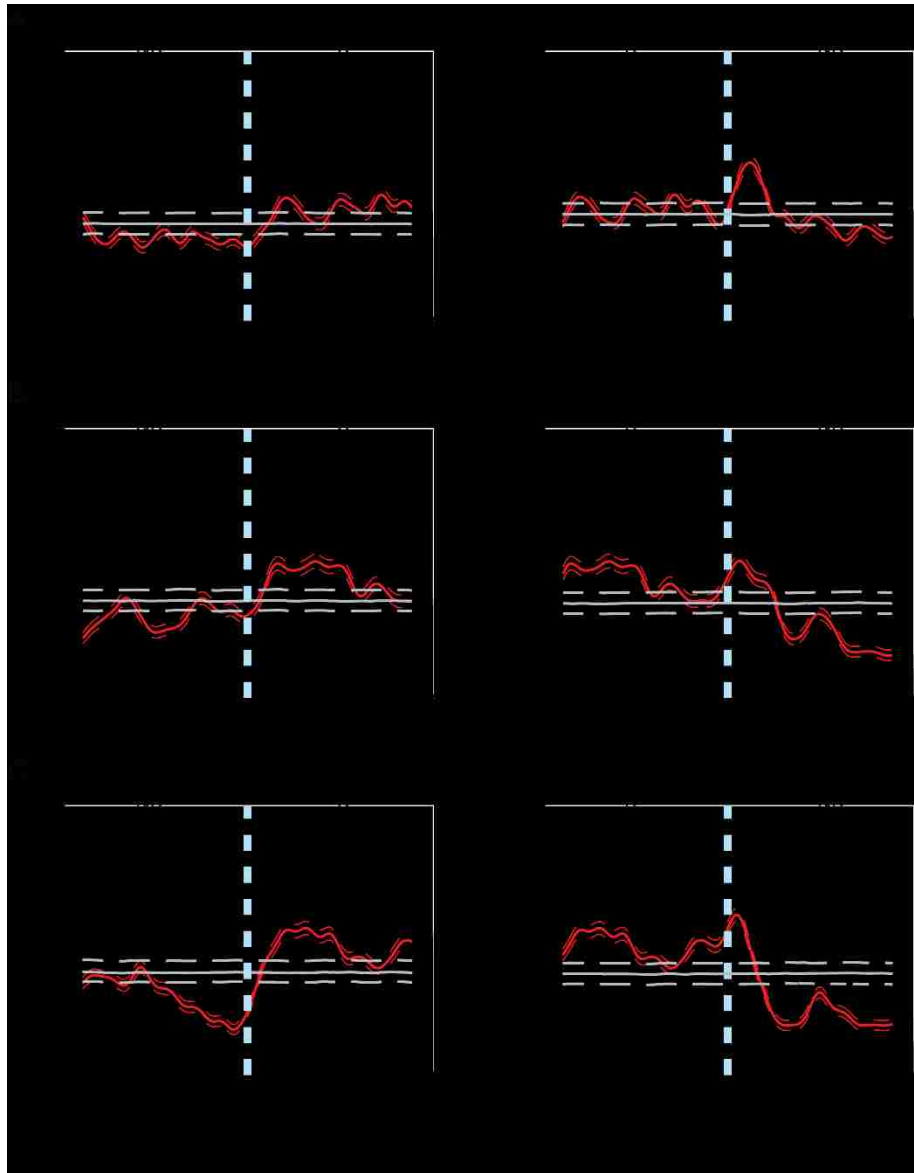
Figure 2.7. In-cage-conditioned spike activity grouped by first third (A), second third (B) and final third (C) of session. Spike activity from the example shown in Figure 2.6C.

2.3.H *Cell conditioned in both environments reveals superior efficacy of in-booth conditioning*

The above evidence suggests that greater conditioning effects were obtained during in-booth conditioning with restraint and visual feedback than during in-cage sessions with free behavior. This could have been due to the slight difference in reinforcement paradigms (Figure

2.2) as well as environment. For a definitive comparison, we conditioned the same cell, using identical conditioning parameters, both in the training booth and as monkey J moved freely about his cage. Figure 2.8 shows rates when spikes from a motor cortex neuron triggered NAc stimulation during elevated firing rates. Stimulation was available during 2-min R periods alternating with 5-min NR periods. During the first hour, the monkey underwent unit conditioning while he moved freely about his cage; he was then transferred within 6 minutes to his training booth and restrained. The Neurochip delivered identical conditioning stimulation in both environments. During reinforcement periods, single 1mA biphasic pulses were delivered to NAc on each event that exceeded 30 counts within a 500-ms-wide sliding window updated every 10 ms. Figure 2.8A plots cell spike activity as time-averaged rates. Horizontal dashed lines show group means of reinforced and non-reinforced intervals for each environment (red and black respectively). The Neurochip generated an auditory click on each stimulation pulse event to provide a discriminative stimulus. No visual feedback was provided in either environment.
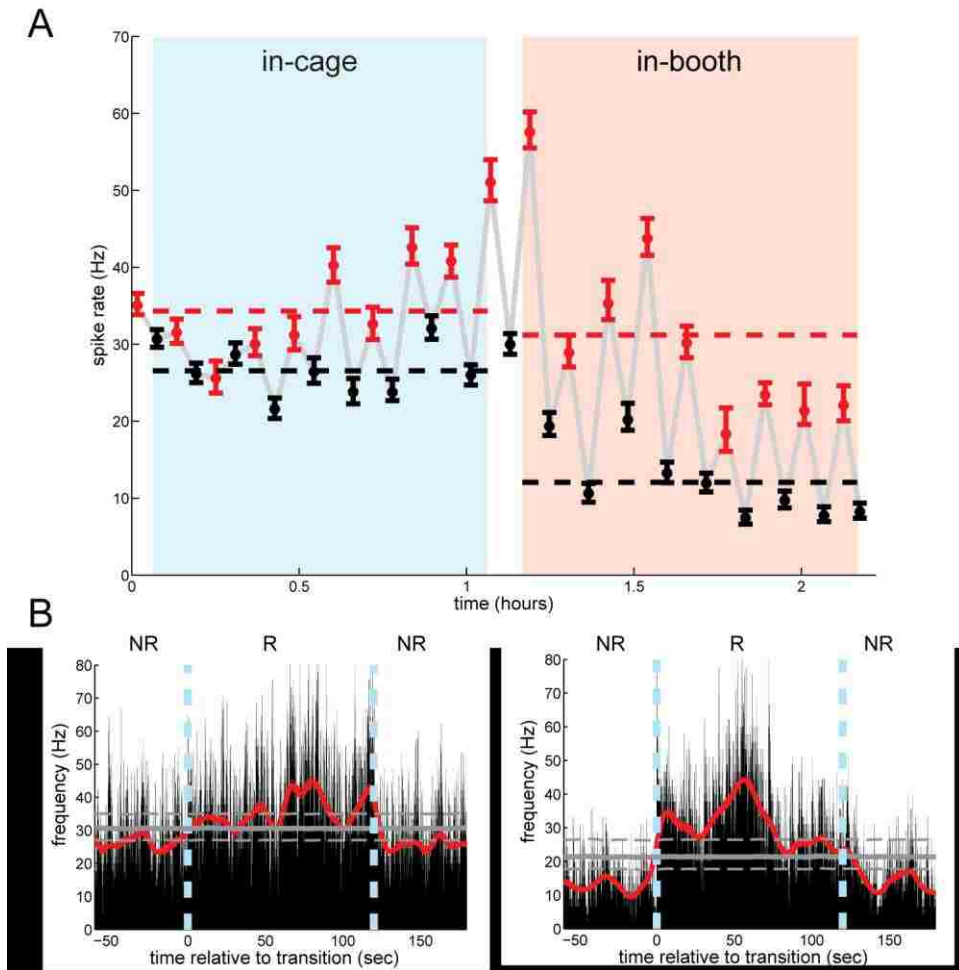
Figure 2.8. Rates of motor cortex neuron conditioned in-cage and in-booth with identical parameters. A. Average firing rates during R (red) and NR (black) and surrounding 95% confidence intervals. The monkey was in his home cage (light blue background) during the first hour and then quickly transferred to the training booth during second hour (tan). B. Peri-transition spike averages compiled during conditioning periods in-cage (left) and in-booth (right). Horizontal lines show overall mean rate for each environment. The Neurochip controlled conditioning in both environments; it ran continuously during the 6 minute transfer interval, and continued uninterrupted through the entire session.

The progression of alternating time-averages of reinforced and non-reinforced cortical cell firing rates show statistically significant increases during periods of BSR reinforcement compared to the intervening NR periods, both in the training booth and the end of in-cage conditioning. Comparisons between distributions of pooled R and NR time averages show statistically significant increases during reinforcement, in both the cage and the booth. The group median of NR period averages during in-cage conditioning (25 Hz) was substantially greater than the median of the NR group during in-booth conditioning (13 Hz), indicating higher baseline rates during free behavior. Peri-transition firing rates (Figure 2.8B) also show higher baseline activity during in-cage than in-booth NR periods and show that cortical cell firing peaked midway through the two-minute reinforcement interval.

2.3.I *Nucleus accumbens stimuli do not evoke cortical responses*

Recent anatomical investigations [29, 30] suggest a more direct pathway through which input from the NAc could reach primary motor cortex than the well-established striatal-pallidal-thalamo-cortical circuit [31] [32]. To address this possible confound of direct stimulus-evoked effects in cell firing, we delivered continuous 5 Hz test pulses to the BSR site while recording spike activity of the candidate cell prior to each conditioning session. None of the candidate cells exhibited statistically significant increases in firing probability at any latency between 0 and 200 ms following single-pulse stimuli delivered to NAc at the current intensity (1 mA) used for BSR. The four representative cases in Figure 2.9 show that the 95% confidence intervals surrounding kernel-smoothed traces of the observed spike event sequences (red) did not exceed chance levels (gray), indicating that the modest transient fluctuations in spike probability in these histograms did not achieve statistical significance. Thus, striatal-cortico linkage did not contribute directly to increases in cortical cell spike activity during unit conditioning with BSR.

31
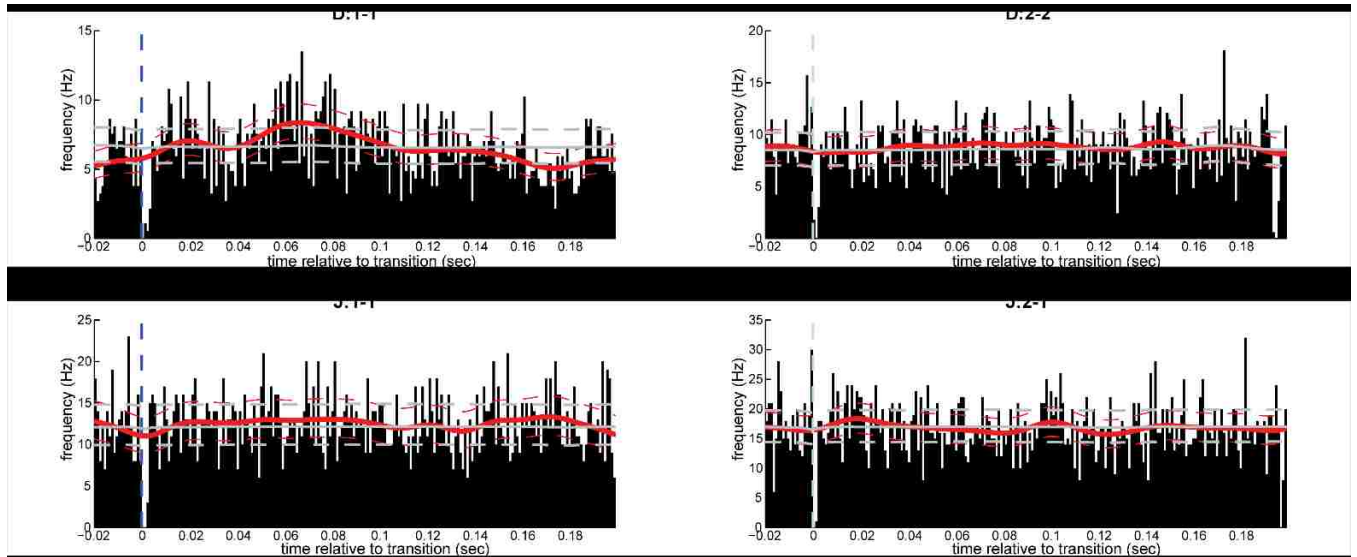
Figure 2.9. Peri-stimulus spike histograms compiled from spike trains surrounding single-pulse stimulation of NAc reinforcement sites. Histograms were compiled just prior to example experiments depicted in Figure 2.5A and Figure 2.5B using the same stimulation intensity. Vertical dashed lines depict stimulus delivery. Histogram bin width: 2ms. Point density average (thick red line) and limits of the 95% confidence band (red dashed lines) depict kernel-smoothed approximates of peri-stimulus mean spike rates.

2.4   DISCUSSION

This study shows that firing rates of motor cortical neurons and muscle activity can be operantly reinforced through delivery of rate-contingent stimulation of ventral striatum in non-human primates. We identified BSR sites in NAc whose stimulation reinforced performance of a target-tracking task with reward efficacy comparable to fruit sauce. Systematic testing of stimulus parameters (width, amplitude and frequency) with the FTT task demonstrated response rates consistent with the Law of Effect [28]. Our stimulation of NAc probably activated fibers that evoked dopamine release, including fibers from the medial forebrain bundle, which connects the ventral tegmental area to NAc and whose stimulation supports operant responding [33].

32

Axon terminals of the medial forebrain bundle release dopamine within the NAc on receipt of unconditioned rewards [34, 35]. Moreover, the reinforcing effects of stimuli that are normally rewarding, such as food, water, drugs of abuse and stimulation of the medial forebrain bundle, are blocked in animals given dopamine antagonists [36]. A significant proportion of macaque NAc neurons modulated their activity during task-contingent delivery of juice rewards [37]. Thus, the reinforcing effects of our stimuli were likely mediated by activating fibers that released dopamine.

2.4.A *Functional relationships between motor cortex and striatum*

The functional relations between the ventral striatum and motor cortex have been elucidated by anatomical, electrophysiological and behavioral studies. Polysynaptic projections from NAc to motor cortex have been revealed by retrograde transsynaptic transport of rabies virus [29]. Conversely, the motor cortex is one of the cortical areas from which the ventral striatum receives input [38], [39]. Simultaneous recordings of cortical surface ECoG and local field potentials in NAc showed evidence for electrophysiological interactions in a study demonstrating that NAc plays a significant role in recovery of motor function after corticospinal lesions [40]. Temporally precise coherence between output-relevant neuronal populations in motor cortex and dorsal striatum developed during learning to control cortical cell activity [41]. Despite this evidence for close relations, we found no evidence that our NAc stimuli produced any post-stimulus modulation of motor cortex neurons, indicating that the effect of stimulation on firing rates was mediated by behavioral reinforcement.

2.4.B *Activity correlated with conditioned neurons*

While BSR was delivered contingent on increases in firing of a single motor cortex cell, larger neuronal populations would obviously have to be co-activated; in particular, other neurons that provide direct and indirect input to the conditioned neuron would also be recruited to drive its rate increases. Such co-activation of large populations was evidenced by associated muscle contractions and neighboring cell activity. During in-booth sessions the monkey's conditioned changes in neural activity were often correlated with isometric torques produced around the wrist. This is not surprising since the neurons chosen for conditioning were modulated during the wrist task. A previous study found that chaired animals allowed to move limbs freely generated a variety of movements associated with operant bursts of the same cell [42]. Given this variability we did not attempt to document the monkeys' movements during the in-cage neural conditioning sessions. A more systematic analysis of movements related to operant bursts during free behavior could be pursued using synchronized neural and video recordings.

In some sessions the activity of a neighboring cell was recorded simultaneously with the reinforced neuron. As illustrated by Eaton (2014)[25], neurons whose cross-correlograms had central peaks indicative of common synaptic drive from upstream sources to both cells could be coactivated or modulated reciprocally in the R/NR periods. These results are consistent with previous studies of synaptic linkages between motor cortex neurons showing that common inputs are seen for both coactivated and reciprocally activated pairs [21].

2.4.C *Comparison of neural conditioning in-booth and in-cage*

Learning to control neural activity progressed more slowly during in-cage than in-booth conditioning sessions. In addition, rate increases were smaller and harder to discern for in-cage

R-period versus NR-periods. Several differences between the two conditioning environments could have contributed to this disparity. First, during in-booth sessions the monkeys were restrained, with their head and contralateral arms secured. We believe such restraint effectively reduced activity of the movement-related cells during NR periods, providing a lower "baseline" against which increases were measured. Second, most in-booth sessions involved stronger discriminative stimuli (e.g., auditory clicks and a rate-controlled computer cursor) than the barely-audible clicks produced by the Neurochip during in-cage sessions. More intense discriminative stimuli are more likely to be effective secondary reinforcers during the conditioning task. Third, the lack of restraint during in-cage conditioning permitted monkeys to explore a much broader range of motor activities. The greater behavioral repertoire provided more distractions when forming response-reward associations, thus requiring longer time to demonstrate acquisition. In contrast, in the training booth, where monkeys had spent many hours performing the FTT task for both food reward and BSR, monkeys likely drew from a much smaller pool of potential reward-eliciting responses when forming neural-response-reward associations. Fourth, the low-pass filtering of neural activity used for most in-booth experiments may have been more effective than the sliding-window method used for in-cage Neurochip sessions (Figure 2.2C). This possibility was disproven in a control session in which the sliding-window method was used for both environments: the monkey's performance was still more robust in the booth, where baseline firing rate was lower (Figure 2.8).

Finally, consistent with the r_e parameter of the Law of Effect model, the in-cage environment introduced additional reinforcers – for example, food, toys, presence of neighboring monkeys and grooming activities – that served to increase competing behaviors to the spike-rate operant. As the collective contribution from all non-task reinforcers, r_e, increases, the influence

of the task-associated reinforcer, r (BSR in our case), on operant responding is effectively

reduced, as shown by the mathematical expression of the Law of Effect for response rate (Eq.

2.1), in which the sum of the two terms $r + r_e$ comprise the denominator. Since fewer non-

task-reinforced response alternatives are available to monkeys in the training booth, the Law of

Effect predicts that the rewards paired to the operant response should be more effective than in

the cage, where there are many distractions.

Most of the above reasons that efficacy of conditioning during free behavior would be

reduced should also have applied for EMG conditioning; however increased EMG responses

proved quite robust for almost 20 hours (Figure 2. 4A). This difference raises the possibility that

conditioning of neural activity might be more difficult than muscle activity; however that

conclusion would be contradicted by many successful unit conditioning studies using

conventional rewards [42-44]. It may be possible that task acquisition itself was faster for EMG

conditioning specifically in the context of free behavior. Thus, while the target muscles were

normally active in the monkey's natural movement repertoire, the relevant neural activity may

not have been as readily discoverable in the cage. Since bursts of motor cortex neurons is

typically related to many different movements [42, 43], these diverse relations could have

undermined the acquisition of any particular effective movement. This hypothesis clearly

deserves further investigation.

2.4.D *Investigating neural coding*

Reinforcement of neural activity with BSR during free behavior has the potential of

investigating mechanisms of neural coding. In contrast to the conventional coding of

information in neural firing rates, the hypothesis that information could be coded in the precise

timing of spike activity remains to be proven. The operation of such temporal coding would significantly expand the bandwidth for neural computation [45]. While we have demonstrated the ability of BSR to reward increases in firing rates, BSR could also be used to test the volitional control of precise spatiotemporal patterns. If the brain uses such patterns during normal behavior, many of them should be volitionally controllable. The use of BSR to instantly reward the appearance of specific patterns under free conditions would provide ample time for the monkey to discover and repeat the relevant behavioral or cognitive state. This would represent a significant test of the existence of temporal coding in the brain.

2.5 BIBLIOGRAPHY

1.   Fetz, E.E., *Volitional control of neural activity: implications for brain-computer interfaces.* J Physiol, 2007. **579**(Pt 3): p. 571-9.

2.   Caminiti, R., et al., *Shift of preferred directions of premotor cortical cells with arm movements performed across the workspace.* Experimental Brain Research, 1990. **83**(1): p. 228-32.

3.   Aflalo, T.N. and M.S.A. Graziano, *Partial tuning of motor cortex neurons to final posture in a free-moving paradigm.* Proceedings of the National Academy of Sciences of the United States of America, 2006. **103**(8): p. 2909-14.

4.   Jackson, A., J. Mavoori, and E.E. Fetz, *Correlations between the same motor cortex cells and arm muscles during a trained task, free behavior, and natural sleep in the macaque monkey.* Journal of Neurophysiology, 2007. **97**(1): p. 360-74.

5.   Carmena, J.M., et al., *Learning to control a brain-machine interface for reaching and grasping by primates.* PLoS Biol, 2003. **1**(2): p. E42.

6.   Jackson, A., J. Mavoori, and E.E. Fetz, *Long-term motor cortex plasticity induced by an electronic neural implant.* Nature, 2006. **444**(7115): p. 56-60.

7.   Taylor, D.M., S.I. Tillery, and A.B. Schwartz, *Direct cortical control of 3D neuroprosthetic devices.* Science, 2002. **296**(5574): p. 1829-32.

8.   Olds, J. and P. Milner, *Positive reinforcement produced by electrical stimulation of septal area and other regions of rat brain.* J Comp Physiol Psychol, 1954. **47**(6): p. 419-27.

9.   Olds, J., *Satiation effects in self-stimulation of the brain.* J Comp Physiol Psychol, 1958. **51**(6): p. 675-8.

10.   Hiatt, D.E., *Investigations of operant conditioning of single unit activity in the rat brain*. 1972, California Institute of Technology.

11.   Widge, A.S. and C.T. Moritz, *Pre-frontal control of closed-loop limbic neurostimulation by rodents using a brain-computer interface*. J Neural Eng, 2014. **11**(2): p. 024001.

12.   Koralek, A.C., et al., *Corticostriatal plasticity is necessary for learning intentional neuroprosthetic skills*. Nature, 2012. **483**(7389): p. 331-5.

13.   Delgado, J.M., et al., *Operant conditioning of amygdala spindling in the free chimpanzee*. Brain Res, 1970. **22**(3): p. 347-62.

14.   Bichot, N.P., M.T. Heard, and R. Desimone, *Stimulation of the nucleus accumbens as behavioral reward in awake behaving monkeys*. Journal of Neuroscience Methods, 2011. **199**(2): p. 265-72.

15.   Bowden DM, M.E., McKeown CW, Baldwin DV, *Map of Positive and Negative*

Reinforcement Sites in the Brain of the Rhesus Macaque (Macaca mulatta), N.P.R. Center, Editor. 2015: Seattle, WA.

16.   Briese, E. and J. Olds, *Reinforcing brain stimulation and memory in monkeys*. Experimental Neurology, 1964. **10**(6): p. 493-508.

17.   Rolls, E.T., M.J. Burton, and F. Mora, *Neurophysiological analysis of brain-stimulation reward in the monkey*. Brain Res, 1980. **194**(2): p. 339-57.

18.   Routtenberg, A., E.L. Gardner, and Y.H. Huang, *Self-stimulation pathways in the monkey, Macaca mulatta*. Exp Neurol, 1971. **33**(1): p. 213-24.

19.   Jackson, A. and E.E. Fetz, *Compact movable microwire array for long-term chronic unit recording in cerebral cortex of primates*. Journal of Neurophysiology, 2007. **98**(5): p. 3109-18.

20.   Rathelot, J.A. and P.L. Strick, *Subdivisions of primary motor cortex based on cortico-motoneuronal cells*. Proceedings of the National Academy of Sciences of the United States of America, 2009. **106**(3): p. 918-23.

21.   Smith, W.S. and E.E. Fetz, *Synaptic Linkages Between Corticomotoneuronal Cells Affecting Forelimb Muscles in Behaving Primates*. Journal of Neurophysiology, 2009. **102**(2): p. 1040-8.

22.   Zanos, S., et al., *The Neurochip-2: an autonomous head-fixed computer for recording and stimulating in freely behaving monkeys*. IEEE Trans Neural Syst Rehabil Eng, 2011. **19**(4): p. 427-35.

23.   National Primate Research Center, U.o.W., *BrainInfo*. 1991-present: http://www.braininfo.org.

24.   Mavoori, J., et al., *An autonomous implantable computer for neural recording and stimulation in unrestrained primates*. J Neurosci Methods, 2005. **148**(1): p. 71-7.

25.   Eaton, R.W., *Operant conditioning of cortical cell and muscle response patters*, in *Physiology and Biophysics*. 2014, University Of Washington: Seattle, WA.

26.   Dayan, P. and L.F. Abbott, *Theoretical neuroscience : computational and mathematical modeling of neural systems*. Computational neuroscience. 2001, Cambridge, Mass.: Massachusetts Institute of Technology Press. xv, p. 460

27.   Davison, A.C. and D.V. Hinkley, *Bootstrap methods and their application*. 1997, Cambridge, UK ; New York, NY, USA: Cambridge University Press. x, p. 582

28.   Herrnstein, R.J., *On the law of effect.* J Exp Anal Behav, 1970. **13**(2): p. 243-66.

29.   Miyachi, S., et al., *Somatotopically arranged inputs from putamen and subthalamic nucleus to primary motor cortex.* Neurosci Res, 2006. **56**(3): p. 300-8.

30.   Miyachi, S., et al., *Organization of multisynaptic inputs from prefrontal cortex to primary motor cortex as revealed by retrograde transneuronal transport of rabies virus.* Journal of Neuroscience, 2005. **25**(10): p. 2547-56.

31.   Alexander, G.E., M.D. Crutcher, and M.R. DeLong, *Basal ganglia-thalamocortical circuits: parallel substrates for motor, oculomotor, "prefrontal" and "limbic" functions.* Prog Brain Res, 1990. **85**: p. 119-46.

32.   Parent, A. and L.N. Hazrati, *Functional anatomy of the basal ganglia. I. The cortico-basal ganglia-thalamo-cortical loop.* Brain Res Brain Res Rev, 1995. **20**(1): p. 91-127.

33.   German, D.C. and E.E. Fetz, *Responses of primate locus coeruleus and subcoeruleus neurons to stimulation at reinforcing brain sites and to natural reinforcers.* Brain Res, 1976. **109**(3): p. 497-514.

34.   Wise, R.A., *Catecholamine theories of reward: a critical review.* Brain Res, 1978. **152**(2): p. 215-47.

35.   Hernandez, L. and B.G. Hoebel, *Food reward and cocaine increase extracellular dopamine in the nucleus accumbens as measured by microdialysis.* Life Sci, 1988. **42**(18): p. 1705-12.

36.   Wasserman, E.M., Y. Gomita, and C.R. Gallistel, *Pimozide blocks reinforcement but not priming from MFB stimulation in the rat.* Pharmacol Biochem Behav, 1982. **17**(4): p. 783-7.

37.   Apicella, P., et al., *Responses to reward in monkey dorsal and ventral striatum.* Exp Brain Res, 1991. **85**(3): p. 491-500.

38.   Takada, M., et al., *Corticostriatal input zones from the supplementary motor area overlap those from the contra- rather than ipsilateral primary motor cortex.* Brain Research, 1998. **791**(1-2): p. 335-40.

39.   Tokuno, H., et al., *Corticostriatal projections from distal and proximal forelimb representations of the monkey primary motor cortex.* Neuroscience Letters, 1999. **269**(1): p. 33-6.

40.   Sawada, M., et al., *Function of the nucleus accumbens in motor control during recovery after spinal cord injury.* Science, 2015. **350**(6256): p. 98-101.

41.   Koralek, A.C., R.M. Costa, and J.M. Carmena, *Temporally precise cell-specific coherence develops in corticostriatal networks during learning.* Neuron, 2013. **79**(5): p. 865-72.

42.   Fetz, E.E. and M.A. Baker, *Operantly conditioned patterns on precentral unit activity and correlated responses in adjacent cells and contralateral muscles.* J Neurophysiol, 1973. **36**(2): p. 179-204.

43.   Fetz, E.E. and D.V. Finocchio, *Correlations between activity of motor cortex cells and arm muscles during operantly conditioned response patterns.* Exp Brain Res, 1975. **23**(3): p. 217-40.

44.   Moritz, C.T. and E.E. Fetz, *Volitional control of single cortical neurons in a brain-machine interface.* J Neural Eng, 2011. **8**(2): p. 025017.

45.   Fetz, E.E., *Temporal coding in neural populations?* Science, 1997. **278**(5345): p. 1901-2.

Chapter 3. OPEN-SOURCE, LOW COST, FREE-BEHAVIOR MONITORING AND REWARD SYSTEM FOR NEUROSCIENCE RESEARCH IN NON-HUMAN PRIMATES

3.1 INTRODUCTION

Behavioral neuroscience research has often used non-human primates as the ideal animal model due to their behavioral and cognitive similarities to humans[1]. Relationships between neural activity and behavior can be studied in controlled environments to elucidate, for example, how motor cortex neuron firing is related to muscle activity [2] or how populations of neurons relate to complex reach and grasp movements[3]. In these experiments the monkey typically sits in a specially designed chair or box used for restraint and transport. The monkey is normally "head-fixed" to protect the recording equipment and reduce noise in the neural recordings. For studying behavior during neural recordings, experiments have traditionally used mechanical systems such as joy-sticks[4] or torque tracking devices[5], implanted muscle activity recordings[2, 6] ,or video monitoring systems positioned carefully over the target behavior[3, 7]. These systems offer the benefit of being heavily constrained, allowing for precise studies of controlled movements and tasks. However, these systems also limit the animal's natural movement, restricting the types of movements that can be studied and therefore the real-world relevance of these studies.

Studies have shown that the correlations between behavior and neural activity may differ when the monkey is naturally behaving in its home environment [8-10]. A recent experiment [11] found significant differences in the cortical response to food reward between a constrained head-fixed task and a freely behaving task. In this experiment food reward was manually

41

delivered during the free behavior portion of the task and all behavioral data were collected with traditional video cameras. Human error and inconsistencies create difficulties in reproducing the same experimental conditions between trials, such as providing a reward in the same way, with the same latency, every time. This leads to confounds in both the neural and behavioral data that are difficult to account for when establishing a freely behaving analog to a traditional constrained task.

One recent application of the relationship between neural activity and natural behavior is the development of brain-machine interfaces (BMIs). Traditional BMI studies examine areas of the brain that have direct implications in voluntary actions, such the motor cortex which directly relates to physical movement [12]. Many BMI studies have shown success in controlling external devices, such as computer cursors[13] and robotic arms[14] using constrained monkeys. Human studies, however, have been much more limited. Chronic invasive BMI systems have been implemented, but they have been limited to patients suffering from tetraplegia [15, 16]. While BMI systems have tremendous potential in helping patients with tetraplegia, these studies do not directly translate to applications for all neurological injuries such as strokes and spinal cord injuries. Alternatively, acute studies lasting a few hours have shown learned BMI control in patients undergoing surgeries for epilepsy prevention [17], but the lack of training time associated with these studies severely limits their applications. Dissociation between motor behaviors and correlated neural activity is possible with training [18, 19]. However this does not mean that the previous mapping is not still present as multiple patterns of activity can exist simultaneously [20]. In terms of BMIs this could translate to unintentional activations of the machine during natural behavior. Thus, it is highly beneficial to begin studying the relationship

between neural activity and natural behavior in a model that is more amenable to a wide range of BMI applications.

Traditionally, neural recording equipment requires large rack-mounted recording systems and long wires coming from the animal's implant. These systems further require the animal to be head-fixed to prevent excess noise from animal movement and to protect the equipment from damage. This limitation has confined most behavioral experimentation to constraint in primate chairs. Recently, wireless hardware systems have been designed to challenge this paradigm by recording neural signals while the monkey is freely behaving [11, 21-23]. These untethered systems can record multiple channels of high quality neural data with limited impact on the monkey's natural movement. For example, the Neurochip systems [22, 23], can record multiple channels of neural data, perform online computations, and provide electrical stimulation while a monkey is freely behaving in its cage. This battery powered system operates autonomously and stores data to a memory card. This system has allowed neural activity to be studied in freely behaving monkeys, while simultaneously providing electrical stimulation [24] or local audio feedback[25]. Untethered recording systems have been used to study the long-term effects of stimulation on neural plasticity [24], correlations between motor neurons and muscle activity during sleep[26], and the relationship between neural activity and untethered treadmill walking [27, 28], among many others. Expanding the use of wireless recording systems to in-cage environments, however, presents new challenges in research. Behavioral monitoring and reward delivery are two important components for closed-loop behavior studies, which are not often included in literature discussing novel wireless recording techniques

Studying behavior in the primate model utilizes a variety of automated and manual methods. Primate chairs are often paired with joysticks or torque measuring systems to record overt motor movements in the controlled environment. These systems are used to directly trigger reward while monitoring neural activity[25] or as a monitoring system during a neural control task to quantify related behaviors [29]. Adapting these systems to a free behavior environment, such as the monkey's cage, is a challenge. Some free behavior studies have triggered behaviors by manually presenting food rewards using offline syncing methods to align the data [11, 26]. This method may be good for short term studies, but becomes unfeasible during long experiment sessions. Automated tracking of motor behavior in a freely behaving monkey has been accomplished with invasive muscle activity sensing [25, 26] or video tracking [27, 28, 30]. Muscle activity sensing only provides information on the instrumented muscles and manually processing video frames becomes unfeasible during long experiment sessions. Placing markers on the monkey [3] can allow for real-time automated movement monitoring. However, markers may begin to deteriorate over long, unrestrained sessions, due to the monkey's natural behaviors. Tracking free behavior quickly becomes a balance between the quantity of information and the ability to process the data in real time. Further, traditional behavior tracking cameras are expensive and require large spaces to operate. New advances in single camera depth tracking [31] can provide easier, lower cost solutions to tracking behaviors in humans. The adaptation of these new systems to primate behavior requires optimization for both the recording environment and the typical animal behaviors.

Here, we describe the creation and implementation of a system to enhance existing free-behavior experiment architecture. We present an approach that allows for short-latency (<500ms) reward delivery and behavior monitoring using off the shelf, low-cost components. Further, we

have created a method that takes the first steps toward fully describing a monkey's natural behavior in its home environment, through an automated motion capture system. We have modified the Neurochip system and created a wirelessly operated feeder that interfaces with the existing experiment architecture to provide short-latency rewards contingent on the monkey's neural activity. Additionally, we have developed novel algorithms for monitoring the animal's behavior using the Microsoft Kinect, a motion tracking camera system. Finally we show examples of the systems working in tandem in a novel operant conditioning paradigm.

3.2   METHODS

Here we describe the individual components (Figure 3.1, 3.2) that allow for untethered reinforcement of neural signals and the monitoring of associated behaviors. Further, we describe a simple operant conditioning task that makes use of these systems.

3.2.A   *Wireless communication from Neurochip device to Control Hub*

Like many untethered recording systems, the Neurochip system has multiple auxiliary ports for connecting additional low-bit rate signals, such as an LED or small speaker. We connected a small, low-cost, 433MHz RF transmitter (433 MHz RF Transmitter Module + Receiver Module Link Kit for Arduino / ARM /MCU WL) to this port allowing for wireless transmission of simple signals to an RF receiver outside of the cage. The Neurochip system can pulse the auxiliary port at fixed intervals or in response to a detected event in the neural signal. Each transmission event consists of 10 sets of 3 biphasic pulses of 1ms pulse width. This paradigm allows for redundancy within the system, increasing the odds that an event will be detected. With this system, we were able to achieve over 99% efficiency, calculated by

45

comparing recorded Neurochip events to transmitted events received by the control hub, only

missing 1 or 2 events during sessions lasting up to 10 hours with hundreds of events. While

other wireless signals, such as Bluetooth, may have higher fidelity and range, this solution is

easy to implement and works well at ranges less than 10 feet with minimal programming burden.

In our application, the signal was used to trigger a feeder system attached to the side of the cage,

so our needed range was less than three feet at all times.
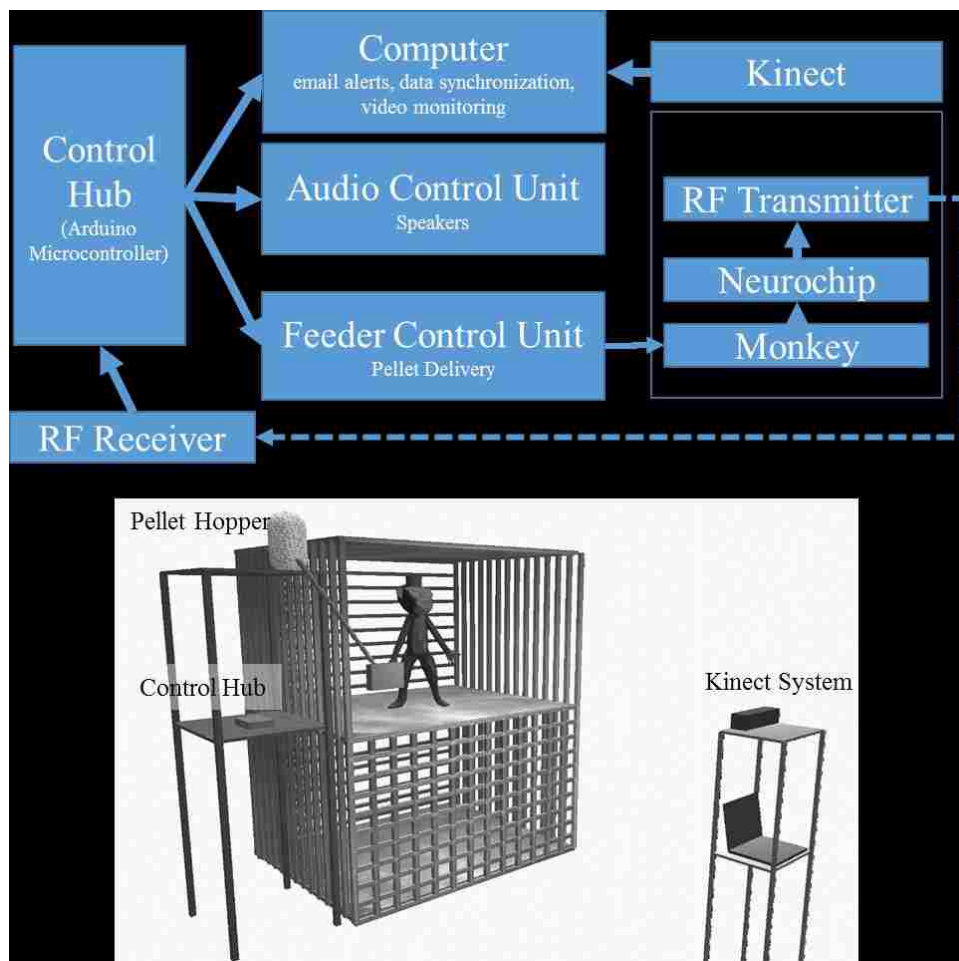
Figure 3.1. Free-behavior hardware setup. The overall system consists of multiple off the shelf

components and a few custom wired control units. **Top.** Block Diagram of primary hardware

components. Wired connections are indicated by solid lines, while wireless connections are indicated by

dashed lines. The control hub to computer connection was wired for simplicity, but wireless

communications, via Bluetooth or Wi-Fi could easily be implemented. **Bottom**. In-cage system setup.

The control hub, audio, and feeder units were mounted on a stand next to the monkey's cage while the

Kinect and computer were mounted on a separate stand across from the Monkey's cage.
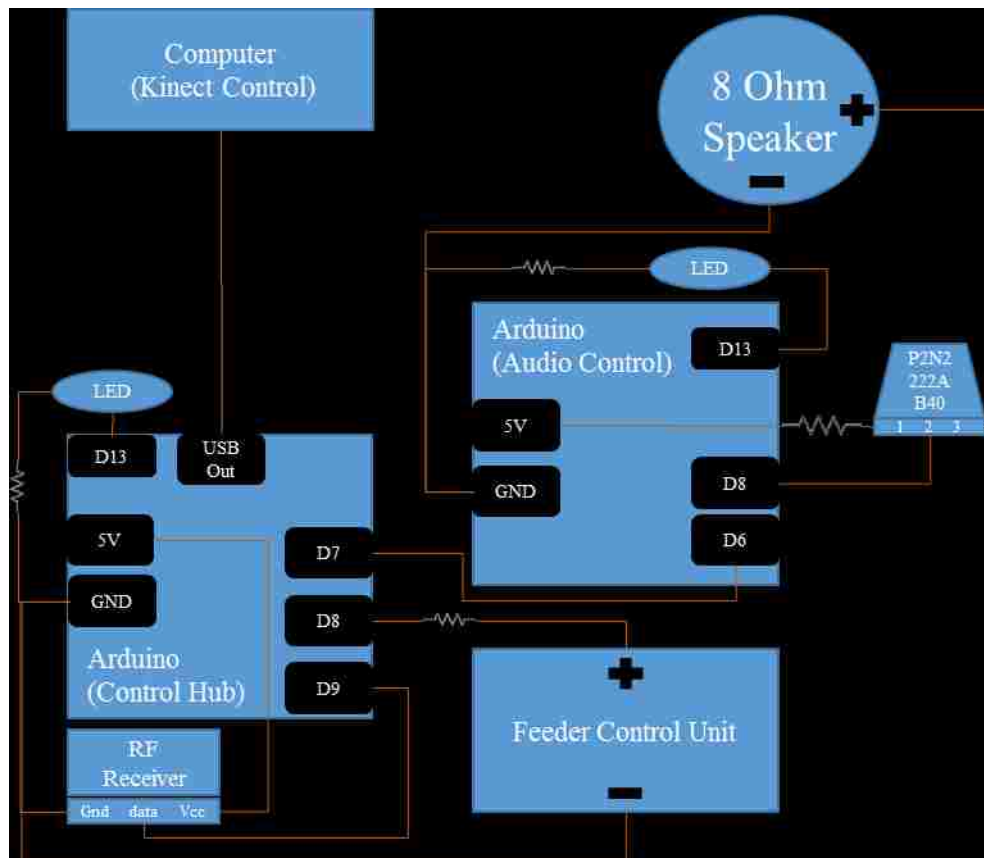
Figure 3.2. **Detailed circuit layout for control hub and surrounding components**. Two Arduinos are

used to control the acquisition of the reward event and the subsequent audio feedback signal. Brown lines

represent wiring between components. The black squares within the Arduinos represent individual

connections to the various Arduino ports. Digital output ports were selected for physical convenience and

could be interchanged depending on experiment needs. Both Arduinos contain an LED link to extend the

embedded Arduino LED feedback system from PIN 13. This LED was used for testing and verification of system function and was hidden from the monkey's view during experimentation.

3.2.B *Reinforcement through food reward and audio feedback*

The RF receiver is connected to an Arduino microcontroller hub (Control Hub), which sends control signals to three separate systems when an RF pulse is received:

1) A cage-mounted feeder that dispenses food pellets to the monkey. The feeder system consists of a stepper motor, a pellet hopper, and a dispensing tube. The feeder is positioned on the side of the monkey's cage with the dispensing tube leading to a small trough attached to the front of the cage.

2) An audio control unit. This unit consists of an additional microcontroller to handle precision audio feedback for the entire experiment. For each detected event, the audio control unit produces 3 short beeps to cue the animal that food reward is available and to provide audio conditioning. Constant tones can be used to distinguish between periods when food is available and when it is not. In control experiments the audio control unit can be configured to trigger additional long-term audio feedback to distinguish between control epochs. For example, during rest periods of an experiment, a white noise signal can be produced to cue the monkey to change its behavior. The pulse width of the feedback beeps and all audio frequencies can be easily configured with a minimum on time of 50ms and a range of 10-4000Hz.

3) A connected computer system to sync multiple streams of data. This computer system handles the behavioral monitoring data streams and provides experiment updates via a Wi-Fi based notification system.

3.2.C *Behavioral Monitoring*

We have developed a system that combines event-triggered video and depth sensing to create an automated monitor of the animal's relevant behavior during an experiment. This system uses a Microsoft Kinect camera to measure the animal's physical location through a combination of traditional video and infrared depth sensing. The camera is controlled through a computer and custom software. The software contains controllers for the video and depth data streams, movement calculations, data saving, and input from the Arduino control hub (Figure 3.3 Top). Both the video and depth streams are visible in the User Interface (UI) during setup. The camera and computer are placed approximately 3 feet away from the monkey's cage on a steady stand. Precision placement of the system is not required, as custom software can change the camera's field of view (Figure 3.3 Bottom). The top, bottom, right, and left margins of the analysis area can be modified such that the movement calculations only occur within the set margins. This is primarily used to outline the cage, removing other cages and animals from the viewport. The front and back depth field cutoffs can also be modified directly, making values outside of these cutoffs fall to zero to increase the depth specificity of the system. This is designed to limit excess reflections from the back side of a cage and minimize the effects of large objects passing between the cage and camera such as technicians and janitorial staff. The system can also be configured to send automated experiment updates via an SMTP email client for remote experiment monitoring.
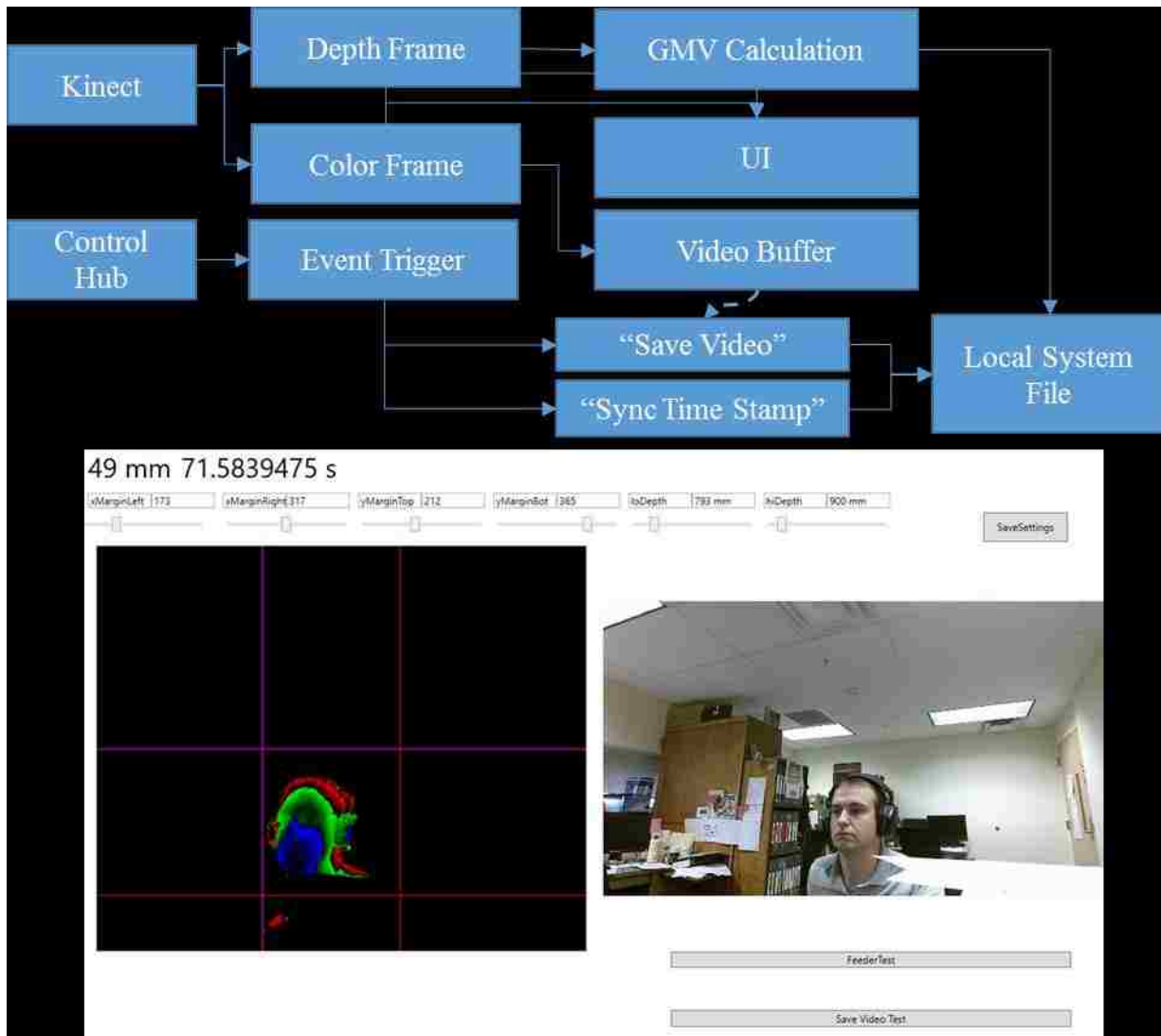
Figure 3.3 Kinect software setup. Top. Information flow block diagram. Behavior data are collected using a combination of the Kinect data streams and the control hub event triggers. The Kinect produces simultaneous color (bottom right) and depth value (bottom left) frames of the scene. Both frames are sent to the UI for visualization of the scene to set margin and depth cutoff values. The depth frame is also passed to the calculation module of the main program. The color frame is passed to a video buffering module which maintains a constant 8-second video. When the control hub sends an event trigger to the program, the video buffer is allowed to run 4 additional seconds before saving the video. The event trigger also triggers a time stamp value to sync the movement calculation. The calculations and the videos

50

are saved to file in real time to prevent excess memory requirements. **Bottom.** The primary UI of the system displays key elements for setting up the behavior monitoring system. The current GMV value, elapsed time and event count are displayed in the top left corner. The depth (left) and color (right) streams are displayed in real time. Each margin and depth cutoff setting has its own slider for on-the-fly adjustments of the field of view for the GMV calculation. Clicking "SaveSettings" applies the settings at the start of the next session. The feeder test button sends a test pulse to the control hub to trigger the feeder and the "Save Video Test" button triggers the video saving functioning of the program.

3.2.D *Gross Movement Measurements*

We developed an algorithm that calculates an average movement value in user-specified regions of the camera's field of view. This value, called the gross movement value (GMV) is calculated based on the movement of voxels attributed to the animal, where a high value indicates a large amount of movement. The GMV provides behavioral data throughout the entirety of a free behavior experiment without any post processing. This value can primarily indicate the amount of animal movement, but does not provide information on whether the animal was moving an arm versus a leg. To calculate the value, the software finds the subject's topmost, bottommost, rightmost, leftmost, closest, and furthest voxels from the depth frame. It then compares these values to the previous frame to calculate a movement estimate (Figure 3.4).

$$\textbf{GMV} = \textbf{||Top[1]-Top[0]||+||Bottom [1]-Bottom[0]||+||Right[1]- Right [0]||+||Left[1]-Left[0]||+||Front[1]-Front[0]||+||Back[1]-Back[0]||}$$

This value can be further refined by separating the camera's field of view into smaller quadrants. This total value can often reduce noise created by reflective elements in the camera's field of view (Figure 3.4C). The current GMV is displayed in the software UI in the top left corner.

$$\text{GMV\_total} = \text{GMV}_{1,1} + \text{GMV}_{1,2} + \dots\dots \text{GMV}_{n,n}$$

There is an inherent baseline in the GMV that depends primarily on lighting and reflections of the scene. This is caused by the creation of the depth value frame which uses infrared light to illuminate the scene. If there are objects that are highly reflective (such as cage bars) the depth frame will flicker around those objects. This causes a baseline GMV that will be non-zero for most cases. In many applications, it is thus appropriate to normalize this value within a session.

3.2.E *Event-Triggered Videos*

Using the camera's color video feed, short video clips can also be recorded automatically through communication with the Arduino control hub. During an experiment the software continuously updates a buffer with recent video frames, configurable based on hardware. For our experiments we processed and saved every fourth frame at 32 fps. When the software receives a control signal from the hub, it triggers the software to save the previous frames as well as a set number of future frames, often totaling four to eight seconds of video. This short video is then tagged with the event number and saved. This method creates short, easy to review videos during relevant time periods, circumventing the manual review required by traditional video monitoring.
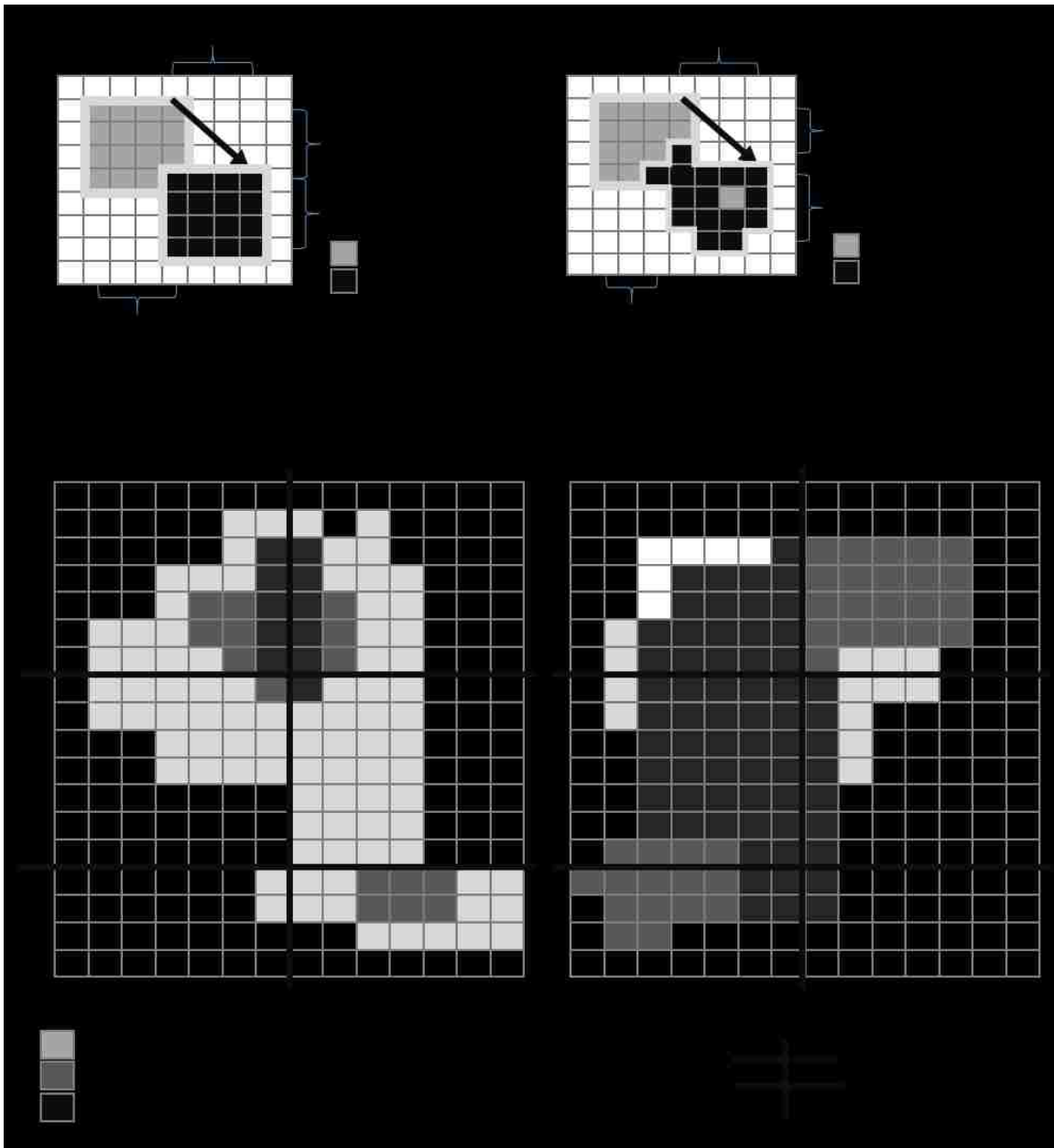
Figure 3.4. GMV calculations. The gross movement value (GMV) is calculated by detecting changes in pixels between frames. In examples A and B, the arrow indicates the transition from frame 1 to 2, whereas example C is split into two panels. The absolute differences between the rightmost, leftmost, top-most, bottom-most, front-most, and back-most pixels are added together to calculate the final GMV. Example B is more complex than example A: the GMV for the top left is higher as more of the mass of the object has moved. Example C most closely mimics the calculation done while the monkey is in the

cage environment. Each frame is broken down into segments (here represented as 6 different quadrants). By calculating the GMV per quadrant, we can detect regions of higher movement activity. This becomes useful in creating event-triggered averages of the GMV over the course of a full experimental session.

3.2.F   *Operant conditioning of multiunit activity in free behavior*

All experimental protocols were approved by the University of Washington Institutional Animal Care and Use Committees. One Macaca nemestrina (monkey J) was previously implanted [25] with tungsten microwires [26] over the wrist area of motor cortex under sterile conditions (see Chapter 2). Monkey J had been pre-trained on separate motor control tasks in both a traditional constrained environment and an in-cage environment [25]. Monkey J was used during the development of the system, in addition to the main experiments described below. During this development stage, alternative cues and control signals were tested, further exposing the animal to the experiment conditions.

Experimental methods closely follow those described in Chapter 2, with instantaneous firing rate calculated using a sliding window. Baseline neuron firing rates were recorded using an autonomous head-mounted recording device, the Neurochip, for 20 minutes while the monkey was freely behaving. This baseline rate was used to determine a threshold for conditioning experiments. Increases in neuron firing rate above a set threshold triggered a wirelessly controlled feeder, followed by a 5-10 second time-out period to prevent triggering of repetitive rewards caused by the retrieval of the food reward from the feeder. Food reward was available in 2-5 minute epochs (Reinforced(R)), followed by 2-5 minute extinction epochs (Non-Reinforced (NR)). R periods were indicated by an audible feedback tone located on the feeder (700Hz) with each reward event triggered by 3 beeps (1200Hz, 250ms on, 250ms off). This system allowed for

a direct comparison between periods of volitional control and periods of rest, within the same experimental session.

Due to the age of the implant and some major equipment damage, we lost the ability to isolate neurons on the microwires. Beta oscillations were still present in the recordings and were used in subsequent testing. This loss also prevented extensive controls from being performed on these experiment conditions.

3.2.G *Operant conditioning of motor cortex field potentials in free behavior*

Local field potentials were recorded from previously implanted tungsten microwire electrodes after no further single neurons could be recorded for the purposes of previous experiments. Monkey J was trained to volitionally modulate local field potentials in the motor cortex while freely behaving in its cage. The Neurochip-2 system recorded these signals using a bandpass filter (10-30 Hz) and calculated signal power from a running average of the rectified signal over a sliding 500ms window. A threshold was determined using a 60-minute baseline session during which no reward was given. A target threshold was set such that if the session had been a rewarded session, the monkey would have received a set number of rewards in the baseline period. This set number varied over time to encourage better task performance but averaged around 1 reward per minute (1.02±0.37, n = 13).

In the primary experiment, increases in beta power above the target threshold caused the Neurochip to wirelessly trigger the control hub to deliver a food reward through the attached feeder. This was followed by a 5-10 second lockout period to prevent multiple rewards. Alongside each reward event the control hub triggered the motion capture system to save the

55

event video and the audio feedback system to provide a reward tone. Food reward was available

in 2-5 minute epochs (Reinforced(R)), followed by 2-5 minute extinction epochs (Non-

Reinforced (NR)). R periods were indicated by an audible feedback tone located on the feeder

(700Hz) with each reward event triggered by 3 beeps (1200Hz, 250ms on, 250ms off). Reward

events were counted using the Neurochip during NR epochs for analysis purposes even though

no food reward was delivered. The lockout protocols were present during the NR epochs for

consistency. Animal movement was characterized by the gross movement value, with high GMV

values corresponding to high amounts of movement. Since no NR events were transmitted to the

feeder, triggers for analyzing GMV were not present during NR periods. This system allowed for

a direct comparison between periods of volitional control and periods of rest, within the same

experimental session. Control experiments, in which the feeder and tone were unavailable for

long periods of time, and food was delivered at random intervals were used to test superstitious

effects from the system. Through these experiments, we tested the efficiency of an overall totally

wireless system in training a monkey to volitionally control brain activity.

3.3   RESULTS

3.3.A   *Neuronal firing rate conditioning*

Only 11 sessions were performed before losing the ability to isolate neurons. Of those 11, 5

sessions resulted in hardware errors and software bugs inherent in the early development of the

system. Of the remaining 6, 4 showed significant increases in firing rate during Reinforced

epochs. In one experiment, the average rate of firing during R epochs was 17% higher on

average than firing rates during NR epochs (Figure 3.5). In the same experiment, firing rate

tended to increase immediately following a reward event (Figure 3.6). This may be the result of

the monkey retrieving the pellet from the hopper, as the 5-10 second lockout period was not

implemented during this early experiment.
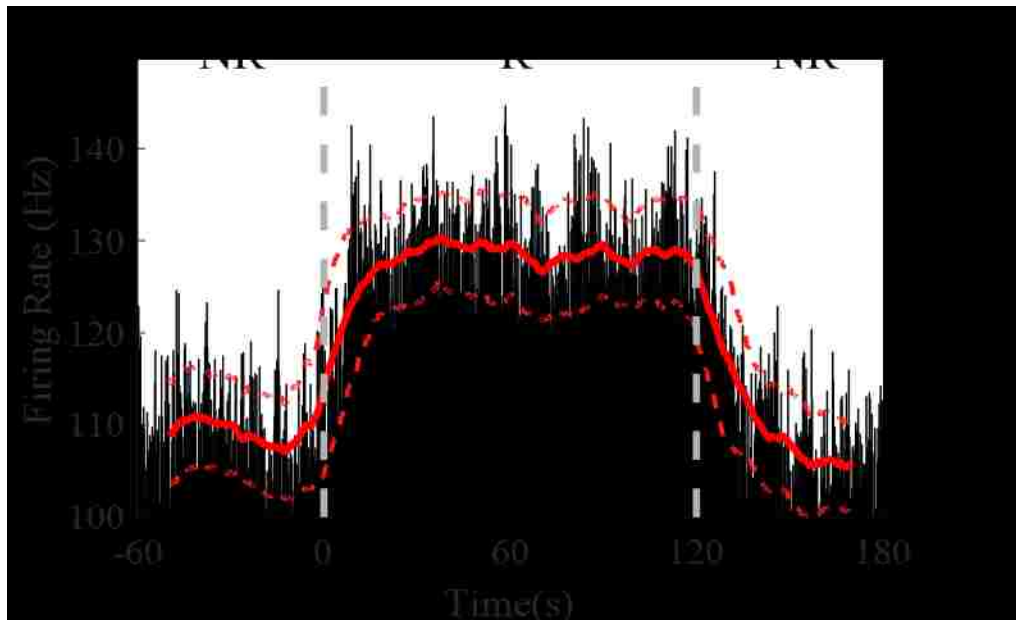
Figure 3.5. Multiunit conditioning of motor cortex neurons in free behavior. The R period indicates

when food was available. Mean+-standard deviation (solid+-dashed red lines) overlay an event triggered

histogram of neuron spikes over 24 transition events.
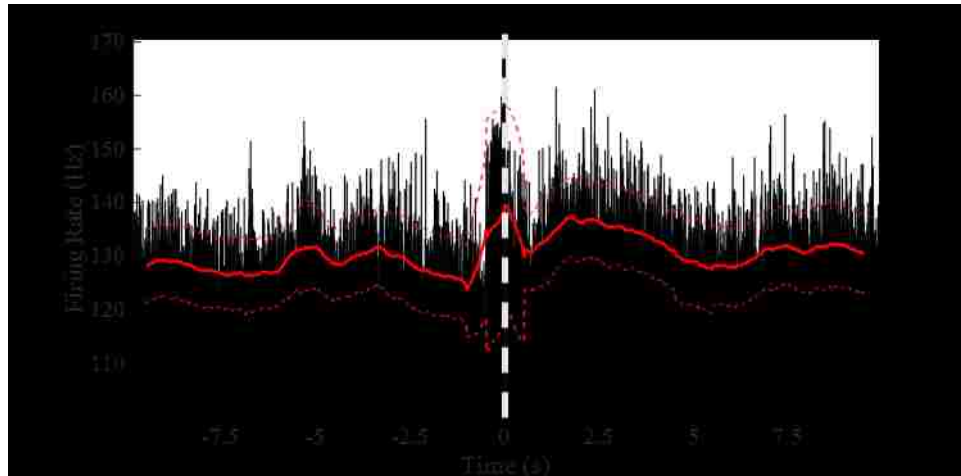
Figure 3.6. Effects of reward on firing rate in free-behavior conditioning. The average firing rate triggered over 473 reward events over a 3 hour conditioning session, shown 10 seconds before and 10 seconds after the event. Mean (solid red) and standard deviation (dashed red) are shown overlaying the spike histogram. Increase in firing rate occurs immediately before the event (-1<t<0) to receive the reward. A short decrease in firing rate follows the reward trigger (0<t<.5), likely due to the presence of the RF transmission artifact which will mask spike events and lower the overall firing rate. An increase in firing rate occurs after the reward event (1<t<4) as the monkey is retrieving the pellet from the hopper.

3.3.B   *Effect of system on local field potential signals*

The system has multiple effects on the monkey's cortical activity as recorded by the Neurochip. The first, and most apparent, is the signal artifact produced by the transmission of the RF signal (Figure 3.7 (left)). This artifact lasts 100-150 milliseconds, but has no other lasting effects on the underlying signal. Another important effect of the feeder system on the neural signal relates to the animal's behavior. When the animal is retrieving the pellet from the hopper after delivery, there is a decrease in beta activity (Figure 3.7 (right)). The signal returns to

58

baseline after 3-10 seconds. In response to these effects, a lock-out period was introduced in which no events could be triggered within 10 seconds of a previous event.
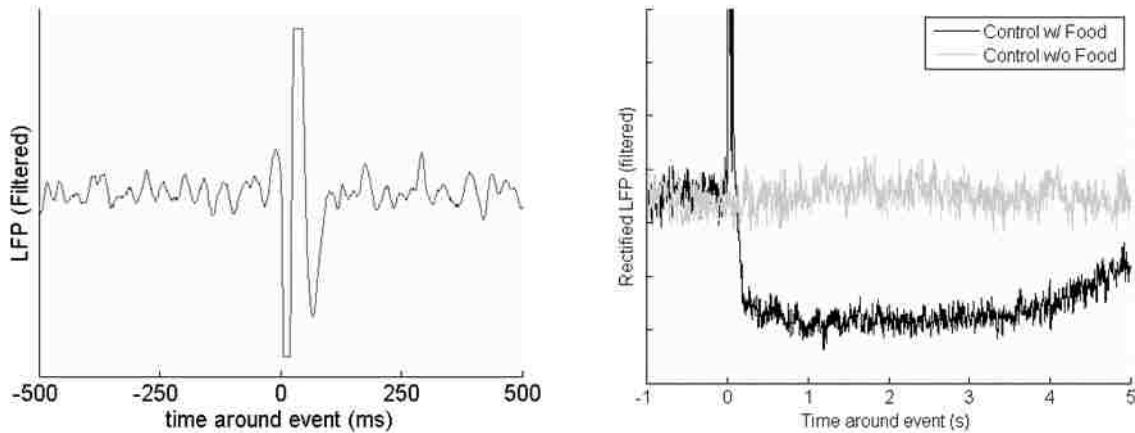
Figure 3.7. **Beta signal response during control experiments. Left.** The Radio Frequency transmitter causes a short (~100ms) artifact in the field potential recording. This figure shows a single artifact during a control experiment where no food reward or feedback was delivered. The artifact saturates the Neurochip amplifiers, but otherwise has no direct effects on the underlying cortical signal. **Right.** The receipt of a food pellet causes a drop in overall beta power compared to a non-food event. This effect is in line with expected decreases in beta power in motor cortex during overt motor movements as the monkey is reaching to retrieve the pellet from the hopper and then returning to a natural behavior.

3.3.C   *GMV correlates negatively with Signal power*

The Kinect behavior monitoring system recorded event-triggered videos and movement values as expected. Video represented only relevant behavioral data, amounting to a much smaller data impact than continuous video. The GMV modulated greatly over the course of an experiment with a typical dynamic range of 100. Baseline values varied slightly across experiments (+-10 GMV) so we normalized the values to the within-session GMV range. The

59

GMV correlated negatively with signal power (Figure 3.8). The video footage and GMV both

show decreases in gross movement immediately prior to an event. After the event, the GMV

spikes as the monkey reaches to retrieve the pellet. This inverse correlation between movement

and signal power in the 10-30 Hz range agrees with previous findings [32].
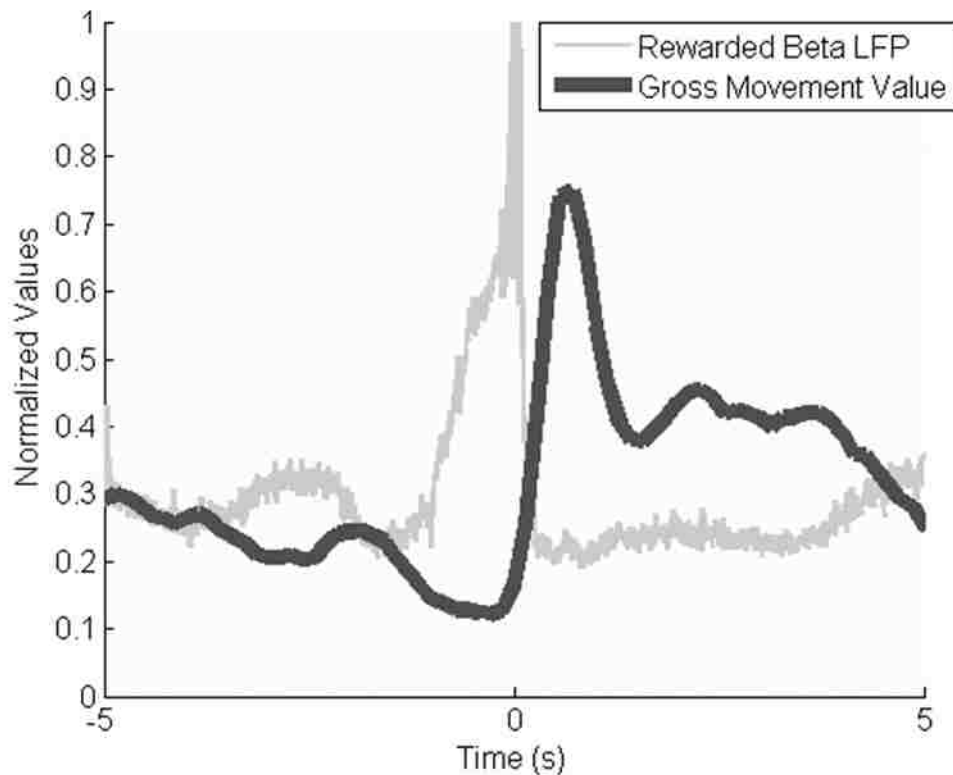
Figure 3.8. GMV vs LFP Beta Power. The inverse relationship between signal power and the GMV is

readily apparent during reward events. In the above figure, filtered (13-33Hz) and rectified neural signals

and the GMV were averaged over 287 reward events while the animal was freely behaving in its cage.

Immediately before the event (t<0) there was a brief drop in GMV as the signal power increases.

Immediately following the event (t>0) there was a large spike in GMV corresponding to the animal

retrieving the reward from the pellet hopper.

3.3.D *Volitional control of cortical signals is trainable in free-behavior*

Monkey J underwent 14 training sessions lasting 4.37 ± 0.64 hours. Comparing the reward rate between R and NR epochs shows that, over time, the monkey was able to distinguish when reward was available and increase task performance accordingly (Figure 3.9). It is important to note, however, that the thresholds were not constant throughout these sessions, varying with daily baseline rates and experimental design. In most experiments, threshold was set to emulate 1 event per minute, while still maintaining a whole-number value to compare against the integer-based power calculation. In session 4, however, the threshold was set incorrectly, potentially leading to worse behavior than would have normally occurred.
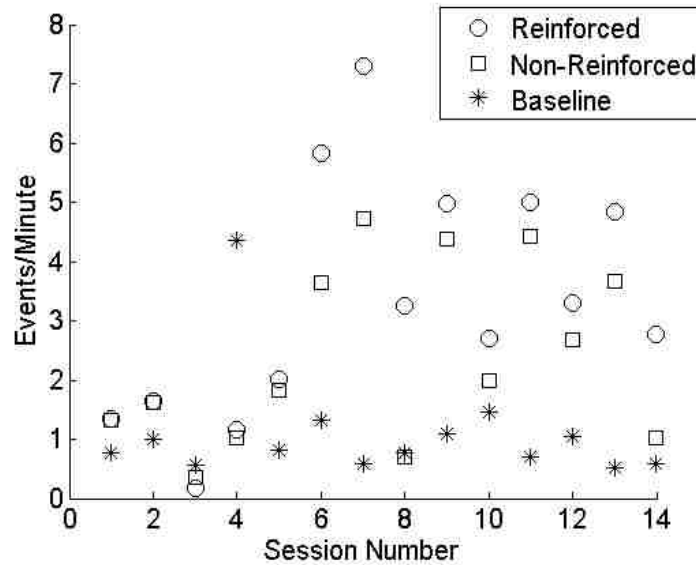
Figure 3.9. LFP conditioning results in free-behavior. Monkey J participated in 14 conditioning experiments over a 3 month period. During early sessions R and NR responding remained close to baseline responding. Responding during R and NR increased above baseline by session 6 and fluctuated over the subsequent sessions. Training was not performed daily, as emphasis was given to system design and refinement. Further studies would employ a more rigorous training regimen, to better optimize the

learning experience. This would likely result in more consistent increases over time and a larger

difference between R and NR periods.

Another important consideration is that the length of R and NR epochs were modified

throughout the course of these sessions (Figure 3.10). Sessions 1 and 2 started with 2 minute R: 2

minute NR ratio, which we hypothesized was too short an epoch to easily acquire the task.

Sessions 3-8 were therefore increased to a 5 minute R: 10 minute NR ratio to allow for a longer

learning and longer rest period. With the success of session 8, we returned the ratio to the

starting 2:2 ratio to provide a more even comparison.
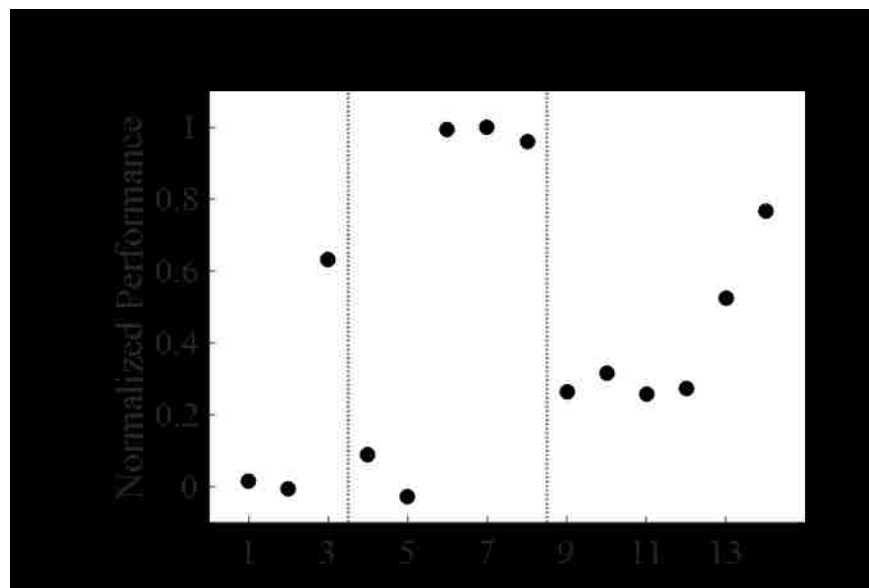
Figure 3.10. Normalized performance across varying reinforced epoch ratios during LFP

conditioning. The ratio between the lengths of each R and NR epoch was varied over time in an attempt

to enhance the monkey's ability to learn the task. The ideal comparison between R and NR epochs is how

many more rewards the monkey received in R than the same amount of time in NR. Increasing the ratio to

5 minutes R and 10 minutes NR allowed the monkey to spend more time becoming familiar with the task

and the associated audio feedback cues. Performance, as measured by R events per minute – NR events

per minute, normalized to the maximum value, was directly impacted at each transition, but returned to higher levels over time.

The distribution of average reward events between R and NR epochs further demonstrate the monkey's ability to distinguish between when reward was available, but more importantly indicates the lack of false positives that occurred during the NR periods. This distinction may be indicative of volitional actions towards retrieving the food pellets. Further insight into the difference between the R and NR periods can be viewed through comparing the average beta power responses around the triggered events (Figure 3.11). A clear artifact can be seen at time zero in the R epoch averages, but other differences are apparent both before and after the reward. Since there is no pellet to retrieve, post reward activity returns to zero. This method highlights an oscillatory wave at about 1Hz prior to the event triggering increase in power.
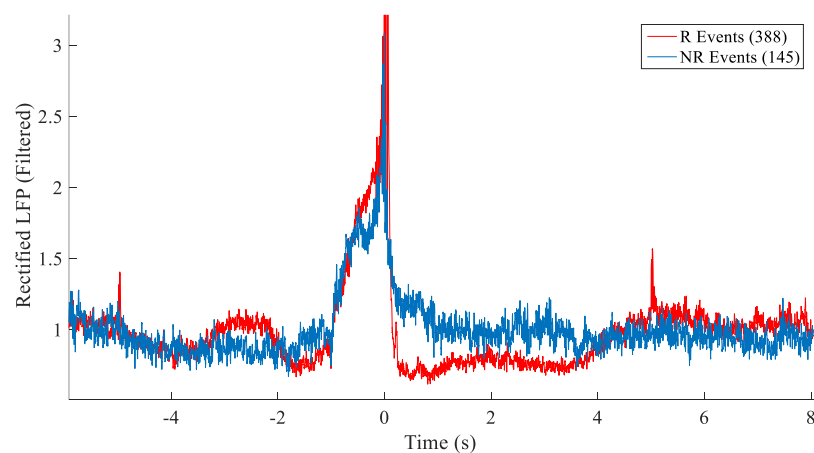
Figure 3.11: LFP power differences between R and NR reward events in free behavior conditioning. Band-passed (10-30Hz) and rectified LFP signals were averaged across events occurring during R (red trace) and NR (blue trace) epochs. The R trace value at t=0 is much higher than the NR

trace due to the RF artifact. Additional differences in the signals can be seen immediately following the reward event where in the R epochs the monkey moves to retrieve the pellet.

3.3.E   *Engagement in task increases likelihood of event*

At the beginning of each experiment, the threshold was set based on a 30-minute baseline session during which no rewards were available. In all but one session, the rate of responding for both R and NR epochs was higher than during the baseline sessions (Figure 3.9). Though audio feedback was available, the monkey may have continued to produce high power signals in case rewards were still present. A few notable sessions, mainly session 14, exhibited behavior closer to what is expected in a traditional operant paradigm. It may be that additional experimental sessions were required to fully wash out this superstitious behavior.

3.3.F   *Event-Triggered Videos Show Relevant Behaviors*

The event-triggered video system worked reliably in all of the 14 experimental sessions, producing four-second videos showing the monkey's behavior in a clear and concise manner. There was not a common pattern of monkey J's location or posture prior to a successful event. Some videos showed pacing behaviors while others showed the monkey sitting primarily on one side of the cage. Almost all videos, however, show a slight pause in movement immediately before each event. This is consistent with the average GMV traces (Figure 3.8) which show decreased values prior to the event trigger. For the purpose of these proof-of-concept experiments, the event-triggered videos served as a basic validation of the GMV. In other experimental setups, they could be used to remove trials in which personnel walked in front of the camera or to study specific nuances in behavior.

3.4 DISCUSSION

3.4.A *Behavioral data collection during ongoing system development*

The previously described system was developed over the course of 8 months with improvements and modifications occurring frequently. Monkey J was used for both testing and the behavioral experiments, which provides an inherent complication for a learning study. The initial experiments involving control of single neurons did not provide sufficient data to show significant task acquisition; however each of these sessions provided familiarity with the system paradigm that influenced the LFP control task. It is relevant to note that monkey J had demonstrated operant control of unit activity in previous sessions involving intracranial reward. Moreover, changes to the auditory feedback system, feeder position, threshold parameters, and Neurochip-feeder communication protocols directly changed the task during learning. As such, the above results only include sessions that occurred with the static system. Most sessions prior to this cutoff did not show significant results due to a variety of equipment and experiment design changes. Future experiments proving the value of this system would likely require exposing naïve monkeys to the system and comparing learning of this task to a more traditional BMI task. However, it is the author's hope that the inherent value of this type of system will be apparent and that similar systems can be adapted for BMI and neuroscience experiments beyond the bounds of traditional methods.

3.4.B *System changes and development: Auditory feedback*

Multiple adjustments were made to the auditory feedback mechanisms over the course of development. Initial faults with the auditory feedback mechanism became apparent during single

65

neuron experiments. Auditory feedback should occur immediately after the response has been successfully performed. In prior work with BSR (Chapter 2) auditory feedback was delivered locally using the Neurochip each time the firing of a neuron was discriminated. This provided immediate audible feedback that scaled with task performance. In the feeder experiments, the delay on transmitting signals from the Neurochip to the Arduino control unit was too large to allow for this type of feedback with latencies ~500ms. This resulted in a lag between the performance and feedback that would likely cause confusion during task acquisition. As such, we transitioned into a simpler auditory feedback control mechanism that only occurred at the completion of the task. Later, we added in a tone that differentiated between R and NR epochs and changed the audio transducer from a simple buzzer to 8-Ohm speakers to increase the volume. There was a noticeable improvement in performance in the R/NR schedule with these last changes.

3.4.C *System changes and development: Reward Lockout*

One of the most important changes was the addition and modifications of the reward lockout to the Neurochip programming. Early tests suffered from an overload of behavioral events causing syncing issues between the Neurochip recording and the behavioral system recording. In some cases, the act of retrieving the pellet would increase firing rates or beta power above the threshold level, causing a recurring event. These cycles were not often broken and would lead to reward satiation. We modified the Neurochip programming to allow for a lock-out to occur after each event that would prevent any subsequent rewards from triggering the "success" functions on the chip. Further testing showed that it could take up to 10 seconds for the

beta power to return to baseline levels after an event. This value is likely to change with different control signals and would need to be modified for each experiment and animal.

3.4.D *Implications for BMIs*

In this work, we presented a system for studying operant conditioning paradigms and BMI systems in an untethered environment. Further, we show an example of the system in use with an untethered monkey controlling specific brain signals to acquire a reward over multiple unsupervised hours. This, to the author's knowledge, is the first representation of a truly freely behaving BMI system with a purposeful end effector and behavioral monitoring. Previous studies have examined individual aspects of this system, such as the wireless data collection, or the behavioral monitoring, but none have fully combined these systems in an operant model. We would also argue that this implementation is significantly different than treadmill models that ask for specific behaviors, since our system allows the animal to perform any behavior to accomplish the task.

The advantages of this type of system are multifold. Operant conditioning studies are possible over longer sessions while the animal is freely behaving. Applications towards novel BMI control paradigms can be more easily explored in an automated way as these systems require limited personnel to operate and can be adapted to run in parallel. New on-chip signal processing algorithms and alternative cortical signal types and areas can be tested quickly, enabling the best solutions to be prioritized for further evaluation in human studies. Areas such as volitional control over prefrontal cortex neural signals[33] and the use of new signal processing techniques [34]could be prime targets for this type of exploration. Further, the system

removes traditional task and booth training from the experiment timeline. These advantages will make studying new BMI systems easier for the field.

3.4.E *Comparison to other systems*

There are several alternatives for behavioral monitoring and reward systems. Investigational systems, such as the treadmill model[27], and commercial systems such as Clever Sys Inc.'s PrimateScan utilize multiple cameras spaced around the behavioral area. These systems offer much greater resolution for behavioral studies than our system. However, they require additional resources to operate that may increase the barrier to entry. Our system uses off the shelf components that (at the time of this writing) any lab can readily acquire. Further, many of these systems require clear lines of sight to the monkey, limiting their use in cages with bars on the front. The removal of cage bars clearly allows for better study of the monkey's behavior, but may not be feasible in some study environments. Behavioral monitoring with our system may not provide a comprehensive model, but it is highly useful for applications where precise monitoring of individual joint angles may not be relevant. Further, this system is ideal for preliminary studies where a gross movement measurement could be used to justify further study with more expensive equipment.

In this work, we presented an integrated system consisting of an augmentation to a wireless recording system, a cage-mounted feeder system with audio feedback, and the behavior monitoring system. Each of these components can easily be interchanged with other available systems. Wireless recording systems, such as the Hermes line [21] and others, could be outfitted with an RF transmitter, and the low resource requirements of the communication protocol should not hinder existing chip operations. The feeder system could likewise be modified to provide

alternate feedback modalities, such as lights or video, thanks to the Arduino system's open architecture. Finally, the behavior monitoring system could be swapped for any of the previously discussed behavior systems. In this way, this work can be viewed as a three separate systems that are adaptable to many other experiments, adding additional control or data streams with a low barrier of entry.

3.4.F *Effect on behavior*

The presence of the camera system, feeder attachment, and control hub could have directly impact the animal's behavior. These systems were moved in and out of the animal housing facility for each experiment to extract the data and tweak the system during development. No abnormal behaviors were noticed during these transitions, indicating that the presence did not increase stress or discomfort of the animal. To further avoid potential confounds arising from these events, it would be ideal to leave the larger components of the system in the animal housing facility between experiments.

The direct effects on behavior caused by the task warrant more exploration in designing improvements to this system. In our sample experiment, we used signals recorded from the wrist area of the motor cortex and thus, any wrist movement could influence our control signal, including retrieving pellets from the feeder hopper. A ten-second lock-out period is too long for many behavior or neural conditioning studies. It may be that this system has better applications in situations where the control signal is not active when the monkey is retrieving the reward.

3.4.G *Reward Latency*

The time period from reward event to pellet delivery was difficult to measure accurately in real time but was less than 500 milliseconds. This was estimated by watching an LED on the Neurochip turn on after a successful reward event and hearing the pellet drop into the pellet retrieval trough. A more robust quantification would be possible with additional sensors, such as an infrared interrupter switch attached to the Arduino control hub, but was not implemented for these tests. This latency is primarily driven by the length of time required for the pellet to travel down the dispensing tube. In future iterations, the dispensing tube could be made shorter or pellets could be cued up at the bottom of the tube with the gating mechanism located closer to the pellet retrieval trough. The larger reward latency comes from the monkey retrieving the pellet from the pellet trough on the outside of the cage. The trough was placed there to prevent damage and allow easier setup; however future iterations could place this trough on the inside of the cage so the monkey would not need to reach through the cage bars. Observational data indicated that the time required to retrieve the pellet decreased over time, but this value was not directly measured.

3.4.H *External Distractions*

One of the main drawbacks to this system, and free-behavior systems in general, is the lack of control over the exterior environment. In our setup, the animal was returned to its cage in a room with other monkey cages. Activity of other monkeys or personnel may cause a distraction that affects the "natural" behavior of the animal. In our system, this effect can be increased, as other animals may be affected by the audio feedback or by the presence of the additional equipment. Additionally, personnel walking in front of the motion capture camera will cause a

70

temporary blanking of the GMV and video data. Overall, these effects were minor in our testing, but are important to take into account when planning an experiment.

3.4.I   *Comparison to BSR Conditioning Methods*

A direct efficacy comparison between this free-behavior feeder system and the brain stimulation reward system discussed in Chapter 2 is difficult for multiple reasons. First, the reward delivery for both systems are drastically different. BSR delivers rate contingent stimulation immediately dependent on individual spike events, while the feeder system accumulated local field potential signals for 1000ms before delivering reward at up to a 500ms latency. With ideal conditions, it is clear that both the BSR and feeder methods can lead to significant conditioning results, though each system is suited for different types of experiments. For example, BSR conditioning works well for conditioning single units, but would require a different signal-to-stimulation mapping to condition gross motor behaviors, muscle activity, or slower brain signals such as local field potentials. BSR requires an additional surgery and reward efficacy verification -- steps which may be costly and take longer to implement. BSR conditioning, however, does not need additional setup at the cage environment and may not suffer the same satiation effects as food reward [35].

3.4.J   *Improvements*

Further improvements to the transmission and receiving protocols could help increase the range and efficiency of the system. RF transmission is cheaper than Bluetooth and smaller than most Wi-Fi modules. However, it lacks the range and efficiency of these other systems. Our transmission and receiving protocols produce very few dropped events; however in some

circumstances, such as when the animal is hunched over away from the receiver, the number of

dropped events can increase. This could be remedied by placing multiple receivers around the

cage, at the cost of increased setup complexity. Additionally, the current protocols prevent

multiple systems to be operating in the same room. Increasing the complexity of the RF

transmission signal by adding a unique ID code could provide one solution, but this could

decrease the chance of detecting an event. Due to these issues, we are actively exploring new,

low cost options in Bluetooth and Wi-Fi to include in future iterations.

Improvements in motion tracking could aid in creating a more precise movement value

that can categorize different types of movements. The second generation of Kinect can provide

more accurate readings of the depth values at higher resolutions. This should allow more

accurate documentation of kinematics in real time, including joint angles and monkey

orientation.

3.5   Conclusions

We have developed the first comprehensive system for rewarding monkeys during free

behavior through an untethered, cage-mounted feeder system. This system was controlled

wirelessly through a small adapter to the Neurochip 2 system, enabling wireless reinforcement of

neural modulations. Further, the system can be paired with a novel motion capture system that

provides relevant behavioral data and short time scale videos. The system exhibits reward

delivery latencies (<500ms) comparable to booth food-delivery systems. Initial experiments

conducted with the feeder have shown promising results in using modulations of local field

potentials to control the feeder. Future refinements to the radio frequency communications

protocol and microcontroller programming will enable greater transmission distances with fewer

dropped signals. These improvements will allow multiple feeders to be operating in the same animal enclosure.

3.6 BIBLIOGRAPHY

1.   Anderson, D.M., *The nonhuman primate as a model for biomedical research*, in *Sourcebook of Models for Biomedical Reasearch*, P.M. Conn, Editor. 2008, Humana: Totowa, NJ. p. 251-8.

2.   Fetz, E.E. and D.V. Finocchio, *Correlations between activity of motor cortex cells and arm muscles during operantly conditioned response patterns.* Exp Brain Res, 1975. **23**(3): p. 217-40.

3.   Vargas-Irwin, C.E., et al., *Decoding complete reach and grasp actions from local primary motor cortex populations.* J Neurosci, 2010. **30**(29): p. 9659-69.

4.   Ifft, P.J., M.A. Lebedev, and M.A. Nicolelis, *Reprogramming movements: extraction of motor intentions from cortical ensemble activity when movement goals change.* Front Neuroeng, 2012. **5**: p. 16.

5.   Moritz, C.T. and E.E. Fetz, *Volitional control of single cortical neurons in a brain-machine interface.* J Neural Eng, 2011. **8**(2): p. 025017.

6.   Griffin, D.M., et al., *Do corticomotoneuronal cells predict target muscle EMG activity?* J Neurophysiol, 2008. **99**(3): p. 1169-986.

7.   Chen, J., et al., *Neural representation of hand kinematics during prehension in posterior parietal cortex of the macaque monkey.* J Neurophysiol, 2009. **102**(6): p. 3310-28.

8.   Caminiti, R., et al., *Shift of preferred directions of premotor cortical cells with arm movements performed across the workspace.* Experimental Brain Research, 1990. **83**(1): p. 228-32.

9.   Aflalo, T.N. and M.S.A. Graziano, *Partial tuning of motor cortex neurons to final posture in a free-moving paradigm.* Proceedings of the National Academy of Sciences of the United States of America, 2006. **103**(8): p. 2909-14.

10.   Jackson, A., J. Mavoori, and E.E. Fetz, *Correlations between the same motor cortex cells and arm muscles during a trained task, free behavior, and natural sleep in the macaque monkey.* Journal of Neurophysiology, 2007. **97**(1): p. 360-74.

11.   Fernandez-Leon, J.A., et al., *A wireless transmission neural interface system for unconstrained non-human primates.* J Neural Eng, 2015. **12**(5): p. 056005.

12.   Green, A.M. and J.F. Kalaska, *Learning to move machines with the mind.* Trends Neurosci, 2011. **34**(2): p. 61-75.

13.   Santhanam, G., et al., *A high-performance brain-computer interface.* Nature, 2006. **442**(7099): p. 195-8.

14.   Taylor, D.M., S.I. Tillery, and A.B. Schwartz, *Direct cortical control of 3D neuroprosthetic devices.* Science, 2002. **296**(5574): p. 1829-32.

15.   Hochberg, L.R., et al., *Reach and grasp by people with tetraplegia using a neurally controlled robotic arm.* Nature, 2012. **485**(7398): p. 372-5.

16.   Wang, W., et al., *An electrocorticographic brain interface in an individual with tetraplegia.* PLoS One, 2013. **8**(2): p. e55344.

17.   Wander, J.D., et al., *Distributed cortical adaptation during learning of a brain-computer interface task.* Proc Natl Acad Sci U S A, 2013. **110**(26): p. 10818-23.

18.   Fetz, E.E. and D.V. Finocchio, *Operant conditioning of specific patterns of neural and muscular activity.* Science, 1971. **174**(4007): p. 431-5.

19.   Schieber, M.H., *Dissociating motor cortex from the motor.* J Physiol, 2011. **589**(Pt 23): p. 5613-24.

20.   Ganguly, K. and J.M. Carmena, *Emergence of a stable cortical map for neuroprosthetic control.* PLoS Biol, 2009. **7**(7): p. e1000153.

21.   Miranda, H., et al., *HermesD: A High-Rate Long-Range Wireless Transmission System for Simultaneous Multichannel Neural Recording Applications.* IEEE Trans Biomed Circuits Syst, 2010. **4**(3): p. 181-91.

22.   Zanos, S., et al., *The Neurochip-2: an autonomous head-fixed computer for recording and stimulating in freely behaving monkeys.* IEEE Trans Neural Syst Rehabil Eng, 2011. **19**(4): p. 427-35.

23.   Jackson, A., et al., *The Neurochip BCI: towards a neural prosthesis for upper limb function.* IEEE Trans Neural Syst Rehabil Eng, 2006. **14**(2): p. 187-90.

24.   Jackson, A., J. Mavoori, and E.E. Fetz, *Long-term motor cortex plasticity induced by an electronic neural implant.* Nature, 2006. **444**(7115): p. 56-60.

25. Eaton, R.W., *Operant conditioning of cortical cell and muscle response patterns*, Ph.D. dissertation in *Physiology and Biophysics*. 2014, University Of Washington: Seattle, WA.

26. Jackson, A. and E.E. Fetz, *Compact movable microwire array for long-term chronic unit recording in cerebral cortex of primates.* Journal of Neurophysiology, 2007. **98**(5): p. 3109-18.

27. Foster, J.D., et al., *A freely-moving monkey treadmill model.* J Neural Eng, 2014. **11**(4): p. 046020.

28. Fitzsimmons, N.A., et al., *Extracting kinematic parameters for monkey bipedal walking from cortical neuronal ensemble activity.* Front Integr Neurosci, 2009. **3**: p. 3.

29. Fetz, E.E. and M.A. Baker, *Operantly conditioned patterns on precentral unit activity and correlated responses in adjacent cells and contralateral muscles.* J Neurophysiol, 1973. **36**(2): p. 179-204.

30. Schwarz, D.A., et al., *Chronic, wireless recordings of large-scale brain activity in freely moving rhesus monkeys.* Nat Methods, 2014. **11**(6): p. 670-6.

31. Shotton, J., et al., *Efficient human pose estimation from single depth images.* IEEE Trans Pattern Anal Mach Intell, 2013. **35**(12): p. 2821-40.

32. Sanes, J.N. and J.P. Donoghue, *Oscillations in local field potentials of the primate motor cortex during voluntary movement.* Proc Natl Acad Sci U S A, 1993. **90**(10): p. 4470-4.

33. Kobayashi, S., W. Schultz, and M. Sakagami, *Operant conditioning of primate prefrontal neurons.* J Neurophysiol, 2010. **103**(4): p. 1843-55.

34. Bryan, M.J., et al., *Probabilistic co-adaptive brain-computer interfacing.* J Neural Eng, 2013. **10**(6): p. 066008.

35. Bichot, N.P., M.T. Heard, and R. Desimone, *Stimulation of the nucleus accumbens as behavioral reward in awake behaving monkeys.* Journal of Neuroscience Methods, 2011. **199**(2): p. 265-72.

Chapter 4. VHAB: A GAMIFIED THERAPY AND ASSESSMENT PLATFORM FOR RECOVERY AFTER NEUROMUSCULAR TRAUMA

4.1 BACKGROUND

4.1.A *Introduction*

Injuries affecting the upper extremity, from the shoulder down through the hand have a disproportionate impact on overall quality of life and long-term independence. These injuries encompass direct trauma to the limb, cases of bone fracture, dislocation, and soft tissue damage as well as higher-order injuries to neural control of limb movement for a variety of conditions including stroke, spinal cord injury and neuro-degenerative conditions. The clinical practice of rehabilitation is one of the most important aspects of long-term recovery from these injuries. This is especially true for neurological trauma like stroke and spinal cord injury which require the re-learning of skills after loss of function. Repetitive practice, broadly through the forms of physical and occupational therapy, can lead to neural plasticity allowing patients to regain function over time. For a complete review see Duncan and Lai 1997 [1], Richards and Pohl 1999 [2], and van Der Lee 2001 [3]. Compared to lower limb dysfunction, regaining upper limb function may be more difficult to achieve [4] but brings significant improvements in quality of life. For patients suffering from paralysis after stroke, recovery is highly dependent on the severity of injury. Stroke has significant heterogeneity and there is much ongoing research on what levels of treatment bring the optimal outcomes for varying levels of injury. Motor loss localized to one side of the body, or hemiplegia, and general limb weakness affects approximately 80% of patients following stroke, creating a huge annual population of patients with clinical need for rehabilitation.

Rehabilitation comes in a variety of specific techniques, most of them involving repetition of specific activities with different types of feedback and sensory support. Traditional upper extremity interventions include neurodevelopmental techniques, bilateral arm training, strength training, task-specific training, trunk restraint, sensorimotor stimulation and training, mental practice, splinting, constraint-induced movement therapy, and mirror therapy. The major categories are detailed below. Recent technology developments and a health system-wide focus on evidence-based care have also fueled innovation in the robotic and digital health delivery of therapy as detailed below. Full context is important for understanding the design choices, parameter space, and long-term goals of developing new technology-based interventions for functional recovery post-injury. These techniques have been clinically tested against each other in a variety of registered clinical trials; however a great deal of diversity exists in clinical practice. This diversity mirrors the heterogeneity of stroke and the lack of understanding of motor learning and functional output reorganization after an individual's unique disease presentation.

4.1.B *Traditional Rehabilitation*

4.1.B.1 Neurodevelopmental Techniques:

Neurodevelopmental techniques focus on inhibiting abnormal muscle patterns and highlighting correct activation patterns in patients. Three main varieties have been adopted in clinical practice over the years. Brunnstom's Movement Therapy [5] promotes synergies of flexors and extensors during recovery, hoping that natural processes will allow specialization later in recovery. Proprioceptive Neuromuscular Facilitation combines manual movement with a focus on the patient's strongest existing movement and verbal coaching to drive functional gains

[6]. The <u>Bobath approach</u> puts patients in postures opposed to their developing synergies and relies on autonomic and reflexive responses to drive correct behavior [7]. These treatments have been found to not be significantly different from each other in patient outcomes [8]. These techniques are the basis for many exercises currently performed in clinic.

4.1.B.2 Task-specific Training:

Specific task practice and performance are required for motor learning to occur and have been established in the rehabilitation space [9]. The intensity of this training does not directly relate to the improvement of function [10]. High intensity, short time programs (3 weeks, 45 min/day) of upper limb training periods have shown improvements in limb function and dexterity [11]. Low intensity, longer span interventions have reported similar improvements in function [12, 13]. The key to this therapy is the task-specific nature of the activities in driving functional outcomes.

4.1.B.3 Constraint-Induced Movement Therapy (CIMT)

CIMT is a slight departure from traditional therapy techniques. To force the patient to use the affected limb in more daily scenarios and increase the focus on repetitive use of the injured limb, therapists bind the unaffected limb, constraining it, and preventing the patient from using their good limb for compensatory strategies [14]. Meta-analyses of the technique have shown an improvement across a variety of clinical functional assessments, including Wolf Motor Function Test, Action Research Arm Test, and the Fugl Meyer Assessment [15]. In most cases, patients were required to self-administer restraints targeting hours per day and six of seven trials included in the review tracked self-reported adherence and compliance; however the one study that actually tracked adherence found that patients were averaging less than 50% adherence [16].

78

This underscores an overarching need for compliance monitoring and motivation in rehabilitation to maximize patient outcomes as well as building techniques for allowing the patient to focus use on the affected limb.

4.1.B.4 Mirror Therapy

Mirror therapy relies on visual feedback to improve outcomes. The technique was originally adopted from the field of phantom limb pain treatment. Patients hide their affected limb behind a large mirror and perform rehabilitation tasks with both limbs while doubling visual feedback only from the unaffected limb. A review of mirror therapy applications has shown improvements in daily function, pain, and neglect [17]. This underscores the importance of dynamic positive visual feedback during stroke recovery for neuroplastic and motivational purposes.

4.1.C Emerging Techniques

4.1.C.1 Robotic Therapies

One factor in retraining function for weak or hemiparetic arm function is adding support and guiding movement through the use of robotic systems, both active and passive, that can support or move the arm through therapeutic movements. These systems benefit from not requiring any implicit ability in the disabled hand and have shown significant recovery potential in the clinical setting. The first and most notable of these systems is the MIT Manus Robot. The Manus and most others allow patients with very limited function to begin motor rehabilitation at the earliest stages of recovery. Several studies documenting the use of the Manus over the course of a decade have shown definite and sustained patient improvement both in the earliest [18] and

later stages of stroke progression [19]. There are a variety of robotic devices currently on the market. However cost remains a significant barrier to clinical adoption at scale, with robots costing between $80k and $1.5M, depending on the model.

4.1.C.2 Digital Therapies

Advances in computer, sensor, and gaming technology have created a new opportunity for delivering and quantifying rehabilitation. Gaming technology has driven a majority of the growth of digital rehabilitation. The most relevant platforms thus far have been based on the Nintendo Wii [20] and Microsoft Kinect [21]. Preliminary studies listed previously indicate that this type of gaming technology provides similar outcomes to traditional therapy techniques, but has the added benefit of automatic quantification of therapies and decreased in-person costs. While the manufacturer's platform games have been used in a therapy setting alongside traditional techniques, a number of companies are emerging to create novel content using the respective sensors. These platforms have the ability to emulate most of the types of therapy listed above. These platforms leverage motion capture through specialized infrared cameras and skeleton tracking algorithms as well as handheld controllers that transmit accelerometry data to a computer to control on-screen actions. The scope and scale of these products allow for capture of low-resolution (~10cm) motion capture of normal postures including large arm and shoulder movements as well as balance and posture tracking.

Table 4.1.1 Kinect-Based Therapy Platforms Currently Available

Company	Product
Jintronix	JRS-Jintronix Rehabilitation System [22]
Reflexion Health	VERA – Virtual Rehabilitation Assistant [23]
Rehabtics	Rehabtics [24]
Mira	MIRA [25]
RespondWell	RespondWell [26]
5Plus	MFAsT [27]
SPECS Laboratory	RGS – Rehabilitation Gaming System [28]
GES Therapy	GesBalance, GesArcade, GesAircraft [29]

While these systems may be able to address large motor deficits in patients post-stroke, the technical limitations of the systems and sensors are not applicable directly to hand function recovery. None of the existing Kinect-based systems provide fine resolution for hand motion capture or any additional physiological data streams.

4.1.C.3 Hand-Specific Digital Tools

There are several digitally-equipped glove sensors that use cloth capacitance sensors to detect finger tapping with the Flint Rehabilitation MusicGlove [30] and the digital exoskeleton from Neofect, the RAPAEL [31]. Additional exoskeleton-like passive systems for assistance with flexion and extension of the fingers can be purchased from the Saebo Company. Each of these glove-based systems brings advantages to the rehabilitation process, from quantifying single degrees of freedom in the fingers to providing mechanical support for flexion and extension during movement. Gloveless motion capture for hand function has recently been possible at low latency and low cost following the release of the commercial Leap Motion sensor. There are a few companies with commercially available beta-stage Leap-Motion-based

rehabilitation devices including Virtualware based in Europe and VirtualTouchTherapy based in the US.

4.1.D *Tools for Clinical Assessment of Motor Function*

One of the key factors driving clinical excitement about the digital revolution of rehabilitation is the ability to bring rapid, automated quantification into the clinical workflow. Traditional measures include a host of standard questionnaires and rating scales as well as a few basic tasks with primitive outcome measures. A full review of relevant hand and arm qualitative questionnaires for documenting hand and arm function are summarized by Baker et al. [32]. Few of these tests incorporate physiological data into determining a patient's status or progress. Quantified assessments are more vague, including the Box and Blocks task, a numerical count of the number of small blocks a patient can move across a partition in a minute [33] and the 9-Hole Peg Task, a timed dexterity task requiring patients to remove and replace 9 pegs from holes in a board [34]. While they both provide a quantifiable number outputs, neither of these standardized tests is based on physiological function or data. Other techniques for assessing recovery include surface electromyography and grip dynamometry. Surface EMG is rarely used clinically because of the time and effort required by clinical staff to prep skin, apply electrodes, and analyze data. Grip dynamometry is also used occasionally to track progress during recovery [35]. Advances in digital sensing technology can be leveraged to improve the speed and consistency with which complex measurements of the hand and arm are captured and reported.

4.1.E *Factors in Clinical Adoption*

There are many confounding factors in the adoption of new methods in rehabilitation that impact the quality, length, and intensity of rehabilitation. The American healthcare system places

an enormous burden on the speed and efficiency of care delivered through most channels. Proper documentation and billing practices in hospitals, inpatient rehabilitation facilities, skilled nursing facilities, and outpatient centers push therapists to deliver care with very little down time. As a result of these factors innovation and novel techniques are slow to reach the rehabilitation space because there is typically a long delay between innovation and payer acceptance of methods and changes in the federal and private payer structures. Consequently, cost and workflow integration are two of the biggest challenges when creating solutions for long-term rehabilitation.

4.1.F *The Advantages of Digital Health Solutions for UE Rehabilitation*

Digital solutions for quantifying and delivering engaging, focused rehabilitation for the upper extremity have the opportunity for enhancing the current standard of care and extending traditional therapy outside the standard clinical channels. The evolution of low-cost consumer sensors and ubiquitous computing with laptops, desktop computers, mobile phones and tablets offers a set of platforms from which rehabilitation can be delivered and quantified in the clinic, in post-acute settings, and long-term in the home. The continuity of care across the long-term recovery allows clinicians and patients the ability to understand small changes in function and manage long-term goals to improve recovery potential. Using remote technologies for care management lowers costs of healthcare delivery and improves adherence to care when barriers such as travel and time, are reduced. The digital world also offers the ability to create engaging simulations and repetitive exercises that keep patient's attention and provide tools and content for educating them during recovery while providing task-specific training and repetitions to drive functional outcomes. By providing digital solutions that emulate the key drivers of current therapy standards we can provide care more consistently and deliver treatment more cost

effectively to a larger number of patients while tracking detailed data on biometric progress

everywhere therapy is delivered.

4.2   SYSTEM SUMMARY AND DRIVING PRINCIPLES

vHAB is a gamified therapy and assessment platform designed to assist patients in upper extremity rehabilitation after an injury such as stroke, spinal cord injury, or trauma. vHAB uses custom software and commodity sensors to deliver fun and engaging games that emulate real therapy tasks (Figure 4.2.1). Therapists can use vHAB with their patients to increase engagement in traditional therapy while simultaneously recording fine details of biometric improvement. This is all accomplished using a motion capture camera to track a patient's hand to control custom games tailored to specific movements. Patient place their hands approximately six inches above the sensor and performs simple movements, such as wrist flexion and extension, which controls various in-game objects. We pair the motion capture camera with a muscle activity armband to provide further insight into the patient's arm function. Combining muscle activity sensing with the kinematic recordings allows us to provide an unprecedented view of a patient's hand and arm during therapy. Data recorded with the system are automatically analyzed to provide detailed measurements to the patient and therapist.

vHAB combines this ability to view detailed progress with the engaging games to enhance adherence to therapy and drive care decisions. vHAB is designed to be used in clinics, such as skilled nursing facilities, outpatient centers, and inpatient facilities, as well as patients' homes after they have left the clinic. This continuity of care allows patients to go home fully understanding the therapy tasks they should perform to enhance their recovery. The software and hardware that comprise the vHAB system are described in detail in sections 4.3, 4.4 and 4.5. In sections 4.6, 4.7, and 4.8 we will discuss the use of vHAB in various pilots and studies, both at home and in the clinic. In the remainder of this section we will discuss the path and design choices made to reach this final system.
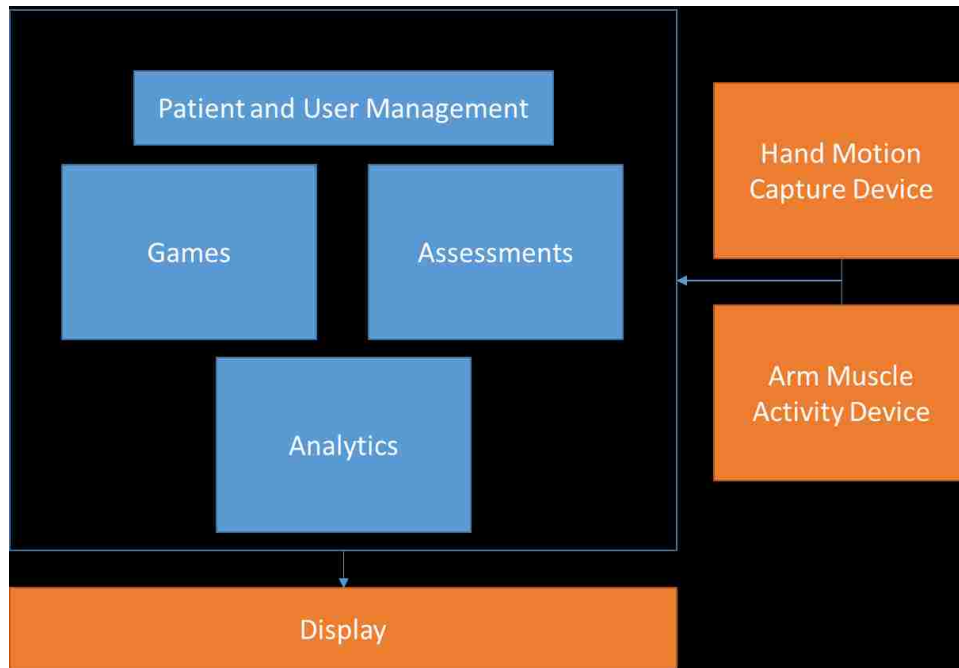
Figure 4.2.1: vHAB components block diagram. The vHAB system uses commodity hardware (orange) and custom software modules (blue) to deliver engaging and cost effective therapy to patients recovering from neuromuscular injuries.

4.2.A   *vHAB Design Principles*

vHAB did not start out as the system it is today. The first iteration of vHAB was created with little thought to the end user. We were designing a game platform partially inspired by the Neurogame system created by Moritz et al. [36]. The Neurogame system utilized muscle activity of the forearm to control a few actions in a commercially available game. Early conversations with the system's creators revealed that patients were able to get much better at the muscle activity control aspect of the game itself, but that the improved control did not directly translate into functional improvement outside the game environment (C. Moritz, personal communication, March, 2014). We knew then that we had to develop more effective games, and we established a few key principles of the vHAB system:

1) **The games are simple to understand, only containing 1 task and 1 goal per game module.**

2) **The games are based on existing therapy tasks.**

Our main hypothesis relied on the idea that therapy tasks performed in the clinic are the current best standard for recovery. We believe that having patients perform those exact movements with our system would at the very least be equivalent to the care they were previously receiving, and that then we would not fall victim to the same challenges of the Neurogame system. With those key principles established, we could then attempt to enhance the therapy experience. These advancements were as follows:

3) **Provide feedback to the patient to drive the correct actions.**

4) **Make therapy engaging to drive adherence in both the home and clinic.**

5) **Present biometric insight to drive quantified, evidence-based care.**

These three principles are clearly evident in existing gamified therapy systems, but have yet to be incorporated into a system built from the ground up containing the first two principles. Finally, we fully understood that the system needs to be cost effective and easy to use to reach patients we set out to help. This first prototype of the vHAB system included a $20,000 clinical EMG recording system and a virtual reality headset. This prototype was too expensive, and difficult to set up in practice. This led us to our final development principle:

6) **Use commodity hardware that is accessible to a general population.**

The word accessible refers not only to the cost, but to the usability. We will expand this concept further in the sections below.

Over the course of this project, we have iterated through 4 main versions of the vHAB system to attain these six principles. Below we detail some of the major design choices that went into building the system. Changes were informed through interviews and focus groups with end-users and field experts, as well as general user testing of the system with patients and therapists.

4.2.B   *Game Design*

As discussed previously, we were strongly inspired by the works of Moritz, among other gamified therapy systems previously described. Many of these systems use commercial consumer games (AAA games) and pass-through software that translates external sensor data into mouse movement or key presses. These systems suffer from three major flaws: 1) learned behavior in the game does not always transfer to physiological recovery or performance in daily tasks, 2) changes in the AAA game, for either security reasons or game improvement may lead to incompatibility with the therapy system, and 3) the game's inherent design does not reflect the therapeutic exercise and may require additional abstraction and training to be effective. We determined early in the design process that to address these flaws as well as meet principles 1 and 2, we needed to create our own therapy games from the ground up. However his brought to light a new problem: we were not therapy experts or game developers.

4.2.B.1 Task Design through Shadowing and Interviews

Early in the development process we established ongoing relationships with occupational therapists, rehabilitation medicine professors, and other field experts. Common themes were present across all of the shadowing and interviews, such as the importance of feedback and functional relevance in a task. During a meeting with Dr. Jared Olson at the University of Washington we came up with most of our early game concepts and some of our design

principles. From this meeting we began development on the first iteration of a reach and grab game and a pinch movement game.

Through a set of meetings with Dr. Janet Powell and one of her Ph.D. students we discussed the importance of accounting for compensation during therapy tasks. Patients will often shrug their shoulders to move their hand higher or move their trunk to reach further forward. In a normal clinical setting this can be accounted for with an attentive therapist's feedback, but at home the patients have no such restrictions or feedback. In our development, we strove to strike a balance between ease of use and preventing unwanted compensation. We also discussed the importance of time sensitivity for clinicians and therapists. They need simple systems that do not interrupt their workflow. Alternatively the system has to be a significant time saver in the long run to be of clinical value. Our system attempts to address this issue through the automatic documentation and assessments, while also providing increased accountability of the therapists.

We had the opportunity to shadow therapists in practice at Harborview Medical Center in Seattle, Washington. In addition to seeing firsthand the antiquated tools they used, we saw the importance of individual finger dexterity and range of motion. One of the most striking examples, however, was a therapist playing a "patty-cake" style game with their patient. The patient was hemiparetic and suffered from spatial neglect. She struggled to keep her right arm in sync with the therapist, but was mostly unaware of her poor performance. The therapist had to continuously remind the patient to watch her right hand, and each time she was given that feedback she was able to perform slightly better. This showed the importance of using both hands concurrently after a hemiparetic injury, and also reinforced the need for audio and visual feedback during task performance. After this experience, we began the development of a two-

handed game in which the patient was rewarded for moving both hands simultaneously while we provide immediate visual feedback that addresses some of the difficulties associated with spatial neglect. It is also important to note that this task was one of the most complex performed in that facility. Most of the other tasks consisted of much simpler movements and instructions, further reinforcing our first design principle.

One of the most important concepts that we are unable to currently address is the use of physical objects during tasks. A significant amount of in-clinic therapy uses putties, paper clips, playing cards, and pencils that the patient is tasked with manipulating in a variety of ways. We knew at the start of our development that we would not be able to view the hand while it is holding an unknown object. These objects are not used as often in the home, usually due to the complexity of the setup. If a patient struggles to pick up a paper clip from the table, they will also likely struggle with getting the paper clip from the packaging or drawer. Additionally, if the paperclip falls to the floor they may not be able to retrieve it. This component of the use experience was particularly striking. We may not be able to use objects, but we can still design an interface and game set that are playable and navigable by the patients.

Many additional games were developed over the course of two years with similar shadowing and interview experiences. We have had the great opportunity to continuously test our developing games with these groups, gaining invaluable feedback and ideas for new games and assessments. Game development during this time was also supplemented with outside knowledge.

4.2.B.2 Game Mechanics and Graphics

We knew we were never going to be able to design a AAA game so we began deriving inspiration from smaller game development studios that use Unity. Unity is a game development platform that uses C# as its main scripting engine and a 3D viewpoint within its IDE to view the game in real time as it is developed. Developing games within this environment was a clear learning process that shows through our game development history (Figure 4.2.2). Increased familiarity with the software led to clear improvements with the game functionality, leading to fewer bugs and better usability. However, we were still limited in our game design skills, and the games were often clunky and visually unappealing. We were aiming to develop a system that not only utilized principles 1 and 2, but were engaging and fun (principle 4) so that people wanted to do their therapy. To help us meet these goals we reached out to a few experts in the gaming field.

Figure 4.2.2. Comparison of the Reach and Grab game over time. **Left**. Initial working prototype from April 2014. The patient's hand was a brown cube that changed height and width to represent an open vs closed hand. The task of the game was to move the white cubes into the trashcan at the bottom left. UI elements were placed randomly throughout the environment. **Right**. Modern version from May 2016. The virtual hand (red) opens and closes with the patient's hand. The task is functional, in that the patient moves fruit into a bowl. UI elements are offset from the environment.

4.2.B.3 Difficulty

We knew the importance in designing games that are playable by patients with differing degrees of impairment, but we were unsure how to accomplish this. In our case, our adjustable parameters were primarily related to the mapping of real world movement to in game movement. By increasing and decreasing the "Sensitivity" values we can make a game easier or harder respectively. Increasing difficulty of a standard game can be done automatically as long as it is seamless, but in our implementation we are also dealing with issues of fatigue and pain. We met with an expert in game difficulty design with experience in designing adaptive educational games (Yun-En Liu, personal communication, November, 2014). We discussed these main issues and began prototyping multiple systems for handling difficulty control. While we created automatically adjusting settings algorithms based on performance or time, we concluded that in the early iterations of the system, simple is better. In the current instance of the system, the therapist has full control of the settings while the patient is in the clinic. The home version, however, has preset easy, medium, and hard settings. This was a difficult engineering choice, where the game design principles were overwritten by the ease of use principles presented from the therapists. While changing settings may take more time, they provide direct control to the therapist and do not require increased learning to understand what the system is doing. Difficulties are set to the easiest settings for each patient at the start of their therapy. Increasing the difficulty is optional at this point, which ensures that the therapists and patients do not become frustrated with the automatic-style systems. These automatically adjusting systems will be further explored later as we continue to learn more about our system's use in practice.

4.2.B.4 Engagement

Throughout the design process we discussed game design principles with Dave Roberts from PopCap Games and Peter Anderson from both Bigfish Games and DoubleDown Entertainment. These three companies represent a combination of commercial casual, AAA, and gambling games. Showing our system to these individuals brought about two main concepts that we incorporated into the development of vHAB and the fulfillment of principle 3. Firstly, our graphic design and game modeling abilities are limited from both a skill and cost perspective. If this is the case in other fledgling game companies, traditional practice stresses that it is more important to focus on game mechanics and usability rather than trying to improve the visuals of a game. Further, steps should be taken to future proof the games so that improvements can easily be made. Following this advice, we began modifying existing games to have a plug and play visual style. This has allowed graphical transitions as shown in Figure 4.2.2 to happen swiftly without slowing down usability of the system.

Secondly, while addictive is a word we tend to avoid in the medical space, we surely want our games to keep people coming back to their therapy on a daily basis. In the casual gaming space, this is accomplished primarily with engagement "wrappers" that tie the game's individual mini-games together. A popular example is PopCap's Peggle which uses Unicorns and other animals as mascots throughout the experience, despite the game having nothing to do with these fantasy creatures. This concept resonated well with vHAB since each of our games has a different theme (or no theme at all). As in the models discussed above, however we did not have the expertise to design a theme around our system. To accomplish this we sought an additional collaboration with the Digital Future Lab (DFL) at UW Bothell to create our garden wrapper. Over the course of 12 weeks, we partnered with the DFL to design the wrapper and

modify our overall user interface (Section 4.2B). This included the creation of art assets and diagrams suggesting the proper use of these assets. For the wrapper, DFL did some initial demographic focus group testing of the persons that would fit the skilled nursing facility market. For this testing, they proposed both a travel metaphor and a garden metaphor. While the testing was not substantive due to time constraints, evidence suggested that the garden metaphor would have the most traction with this market. Again, as in the 3D modeling, we have built the wrapper in a way that could be easily extended into a travel, or other themed, metaphor. The garden wrapper sits as a separate set of UI elements in the Game Select Screen (Section 4.4.A.7-8) and within the games themselves (Section 4.4.B). Within those sections we fully describe the individual aspects of the garden wrapper. Overall, we believe the use of the garden wrapper will aid in the patient's desire to return to the system over the course of their recovery.

4.2.B.5 Instructions and Tutorials

As previously discussed, we set out to design simple games, with only one task and one movement. While we succeeded fairly well in this game design, we struggled with the patient experience with the system. Users were verbally instructed to start with their hand about 6" above the sensor, but for nearly all first-time users (patients and otherwise), patients would place their hand directly on the sensor, sometimes even picking it up and waving it around. Solving this design problem was two-fold: we slightly modified the hardware, which we discuss in Section 4.2.D.2 below, and we began creating tutorials within the games.

Early tutorials consisted of in-game video overlays (Figure 4.2.3 (Top Left)), which were often obtrusive and barely helpful, or a series of looping images in the top left corner of the game (Figure 4.2.3 (Top Right)) which were time-intensive to create and had limited success in

helping patients understand the game. These systems were steps in the right direction but did not quite reach the level of instruction we needed. With the help of DFL, we designed non-blocking tutorials that led a new user through each game (Figure 4.2.3 (Bottom)). A non-blocking tutorial provides information to the user while allowing them to perform the same actions they could normally. For example, in *Reach and Grab*, a user is instructed to reach out and grab virtual fruit and place them in a bowl. The tutorial starts with a text popup saying "Place your hand 6" above the sensor." As soon as we detect their hand over the sensor, we show them their virtual hand and the text changes to say "Move your hand towards the fruit." As their virtual hand moves towards the fruit we then present them with the next prompt. These tutorials are designed to be ignored by experienced users, but helpful for first time users. They allow for general exploration of the virtual space and are only presented during the patient's first play through of the game. Any tutorial can be retriggered through a button in the user interface.

Through our work with Digital Futures Lab we also determined that our potential offering of 17 games would be too many for any therapist to start with given our current UI system. We noticed this as well during one of our first pilot studies (Section 4.6), where we tried to explain how just 10 games worked over their short 1-hour lunch period. Our current system has more than 10 games built into the system, but only 7-10 are presented at any given time. Further, we have developed training documentation and one-sheet summaries that we leave behind with the therapists for further reference.

Figure 4.2.3. Tutorials for the Ball Roll game throughout vHAB development. Top Left. A quickly

abandoned tutorial method. Clicking the small "?" button in the top left corner would popup a video

showing the system in use which would block the user's view of the game. **Top Right**. A less blocking,

but inefficient tutorial method. The image on the top left would cycle through 6-10 frames showing how

to play the game. These were effectively carefully controlled videos with additional information overlaid,

but were not viewed as very helpful in our pilot studies. **Bottom**. The current tutorial method. Our current

method has less information that either of the previous versions, but has been viewed as much more

helpful. We still include video tutorials, but they are in the game select screens before the game is loaded

(Figure 4.2.5 (Bottom)).

4.2.B.6 Game Audio and Visual Feedback

Early versions of the vHAB system did not include any audio feedback. Additionally, the

visual feedback only related to either a "score" event or the movement of the patient's hand. This

made the games relatively mundane over longer playthroughs. Adding sound effects to contact

events made a huge difference in the games, while also making many of the games easier to

understand. Many people who played *Reach and Grab* would move their hands into the fruit and try to make a fist, which led to a very unnatural experience (Figure 4.2.4 (Left)). Further, the fruit would be held in odd positions with the virtual hand or they would not know whether they had successfully grabbed the fruit before moving away. With the assistance of the Digital Futures Lab, we designed and implemented a "magnetization" system (Figure 4.2.4 (Right)). The magnetization system works by quickly moving the target virtual object to the correct position under the patient's virtual hand as soon as the hand is nearby. During this movement we provide a "zip" sound that informs the patient that they have successfully moved towards the object. While the object is in position, the patient can make a fist to "grab" the object which is signified by both the change in hand shape and a "ding" sound. The patient can move the magnetized object around while in a small area without grabbing but if they move out of that range the object will move back to its starting position. This change to the visual feedback, with the addition of task related sounds, made the games much easier to play and understand.
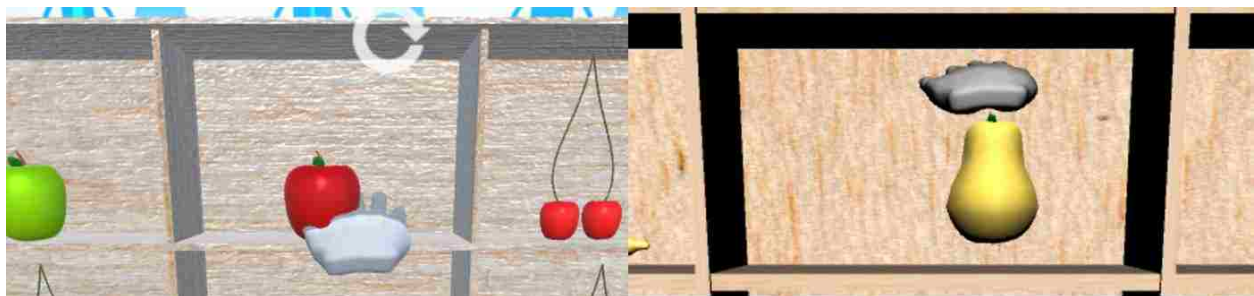
Figure 4.2.4. Reach and Grab grabbing control techniques. Left. Initial control system. In this version the virtual hand could intercept the fruit before a grabbing motion was made, causing an unnatural experience for the user. **Right**. Current magnetization control system. This version may feature floating fruit, but it allows for the user to know they are in the correct position to then make a fist to grab the fruit.

4.2.B.7 Assessments vs Games

All of our games make measurements related to that game's task and associated functional movements. These measurements include individual joint range of motions, reaction times, tremor, and many others. Over time we have refined these measurements, making them faster to compute and more robust to missing data or short sessions. Each game, however only computes a few task related measurements. We set out to design a system that allowed for a full workup of arm function that could be performed periodically throughout a patient's recovery. This would allow a therapist to identify weaknesses in the patient's progress and adjust care accordingly, while simultaneously providing a reliable and consistent measure. Our first attempt was the creation of vAssess, a preloaded set of tasks that led a patient through making a fist and squeezing, reaching range of motion, and wrist range of motion measurements. vAssess was well received with outpatient clinics that did not necessarily perform therapy in the clinic, but wanted a consistent way to measure progress. However, most of these clinics are very time-restricted with their patients and vAssess took anywhere from 3-5 minutes to complete. Further, the clinicians only cared about 2 or 3 of the 10 measurements we were presenting. We incorporated this feedback into the creation of a modular assessment system that allows for custom sets of tasks to be created for specific uses (Section 4.4.C). Now a clinician can create an assessment that only performs the wrist angle range of motion measurements, saving time and making the presented results easier to understand.

4.2.C *UI Design*

The user interface for the vHAB system has undergone nearly as many changes as all of the games combined. First we needed a way to manage navigating between all of the games, then

we needed a way to change the settings for each game. Next we needed a way for patients to have their own settings, so that when a therapist started using a system with a patient they did not need to reset all of the settings. Finally we needed a way for a therapist to create new patients and identify them easily, which required a login and therapist management system. This base functionality then allowed us to add items such as patient notes, data viewers, and patient dashboards. All of these systems are described in detail in Section 4.4.B. In the following text we describe a few of the design changes that led us to that final system.

4.2.C.1 Aesthetics

One of the important changes throughout all of our user interface screens was color and UI element consistency. This was not one of the strongest pieces of early feedback from healthy users, patients or therapists. These groups certainly cared more about system functionality than aesthetics, but as became apparent in early user testing, the system would see easier adoption with a cleaner and more consistent UI. Users would often miss some functionality because it was hidden behind some off-color button that they thought was just a label. As an example, for many months during an early pilot, no one knew they could take notes on their patient because the "User Profile" tab (Figure 4.2.5 (Top Right)) was the same color as the currently visible level navigation panel. We have made substantial efforts to make UI elements consistent in our development since this feedback.

4.2.C.2 Level Navigation

We tested multiple level navigation and level organization methods throughout the course of development. We started with just numbering each scene and using keyboard shortcuts to navigate. This clearly would not last in a clinic, as even the developers has difficulty

99

remembering which game was assigned to each number. We then moved to a UI button system where each game had a static button on the screen that loaded the game (Figure 4.2.5 (Top Left)). This quickly became a problem as we started swapping games in and out. We then moved to a dynamic button list that was generated based on all the levels in the system (Figure 4.2.5 (Top Right)). This worked much better and was great for use by the therapist; however many patients had difficulty pressing the smaller buttons. As we added the garden wrapper and moved to a unified patient experience color theme, we settled on the current game select screen (Figure 4.2.5 (Bottom)). The therapists can then limit the games displayed on this screen through a separate Game Settings screen (Section 4.4.A.4). Our next challenge with the level navigation comes alongside the addition of more games. One of our long-term goals is to categorize the games based on functional movements or associated activities of daily living. This will require yet another level navigation redesign to ensure the system works well with both patients and therapists.

Figure 4.2.5. Level navigation screens throughout vHAB development. Top Left. First button focused level navigation menu from version 0.1.4. This menu was easy to use, but was limited in the information provided about each game. **Top Right**. A more organized level navigation menu from version 0.3.9. Here we provided more information for the therapist, and were closer to the correct solution. The "User Profile" tab was eventually moved into the patient dashboard screen (Section 4.4.A.3). **Bottom**. Final level navigation screen. With the addition of the garden wrapper, the layout in 0.3.9 was not sufficient for the amount of information provided. Further, we wanted to scale back the number of games presented at any time to enhance the user experience.

4.2.C.3 Settings Controls

Modifying settings within the vHAB system is crucially important for not only ensuring that a patient can play a game at the beginning of their therapy, but that the game can become more difficult as the patient recovers. The very first iteration of settings controls required a complex pressing of buttons while in a game that resulted in unclear functionality while

simultaneously blocking the gameplay (Figure 4.2.6 (Top Left)). This was quickly discarded in favor of an intermediate screen that sat between the level navigation and the start of the game (Figure 4.2.6 (Top right)). This still provided a confusing experience for the therapists, especially when they did not want to modify any settings. The current iteration allows settings to be changed in two ways, both of which are easy to do, but fully optional. The first is with a special Game Settings screen (Figure 4.2.6 (Bottom Right), Section 4.4.A.4) and the second is with an in-game popup accessible in the in-game UI (Figure 4.2.6 (Bottom Left), Section 4.4.B). These two complementary methods allow for session planning before a patient is present and for real time changes in response to the patient's performance. Additionally, we have made many changes to the names and values associated with the settings. In the game functionality a value may range from 0.3 to 5.3 and be named "fingerAngleGain", but we remap this to a 1-10 slider named "Sensitivity". Finally, all settings are paired with detailed descriptions and usage suggestions that can assist a naïve therapist.

Figure 4.2.6. Game settings throughout vHAB development. **Top Left**. Early settings management system in the dial turn game (v 0.1.4). Here we used the Dial Turn game settings as the example since it had the most settings at the time. This system allowed for changes within the game, but blocked gameplay and was difficult to understand. A score Difficulty of -1.016148 was an irrelevant number for both us as developers and the therapist. **Top Right**. The settings management system from version 0.3.8. Here we started getting better at describing the settings, but having a separate screen in between game select and the game did not work with therapist workflow**. Bottom Lef**t. The Game settings screen of the current version. Here a therapist can control the settings for each game before the game select screen is loaded. **Bottom Right**. In-Game Settings in the current version. Alternatively, the therapist can change the settings in game without hindering the patient's ability to play the task.

4.2.C.4 User Interface Flow

The user interface flow underwent extensive changes throughout the development process as we iterated on making the experience seamless for therapists and patients. In early

builds, the level navigation system, like one shown in Figure 4.2.5 (Top Left), was the first

screen the therapist saw. The patient management and therapist management were added at the

point when we needed a way to load patient specific settings. After this point we had 4 distinct

UI screen types: therapist login, patient select/management, game settings, and game select.

Integrating with therapist workflow is one of the greatest challenges in ensuring the system's

ability to help patients. We attempted to minimize the number of button presses a therapist would

need to perform to start using the system. This is balanced, however, with system security and

clarity of use. We eventually made the game settings screen an optional screen supplemented by

in-game settings, but getting to this settings screen was not straight-forward and the system was

overly circular. Therapists would find that they had accidentally logged out a patient in the

process of changing settings and were then unsure if the changes had stayed the same. For the

most recent version we added a patient dashboard screen. This screen comes directly after a

patient is logged in and acts as a patient "home" screen. Four simple options are presented in the

dashboard that categorize the actions a therapist can take: Games, Assessments, Settings, and

Data Viewer. Each of these screens has home buttons that take the patient straight back to the

dashboard.

4.2.C.5 Data Presentation

A core feature of the vHAB system is the automatic measurements that occur while a

patient is playing a game or doing an assessment. While calculating these measurements is

relatively straightforward (Section 4.5), presenting them was an entirely different matter. To

accomplish principle 5 we needed to be able to inform the therapists how their patients were

performing within any given game session. We attempted to do this in real time with the first few

iterations of the system (Figure 4.2.7 (Left)), but this was obtrusive and was not very useful for

tracking progress. We then moved to a small graph present in the Game Select screen (Figure 4.2.7 (Right Top)). This graph created points for each instance of the game that occurred and plotted them each on the graph. This method was flawed for two main reasons: 1) the data were presented linearly, but not collected that way. A session four days ago had the same spacing as two sessions within the same day, which made the data difficult to interpret; 2) the graph was quite small and we had many different variables to present. A toggle system allowed therapists to turn on and off the displayed variables, but it was not quite enough to present the data clearly. We ended up creating a completely separate screen for viewing patient data and modified the x-axis to be calendar based (Figure 4.2.7 (Right Bottom)). Therapists can view data based on day, week, or month, allowing for easy comparison of values over time. The toggle system still exists, but as a larger UI Element. In future iterations we will modify the toggle system further by presenting the most useful variables at the top of the list for easy access.
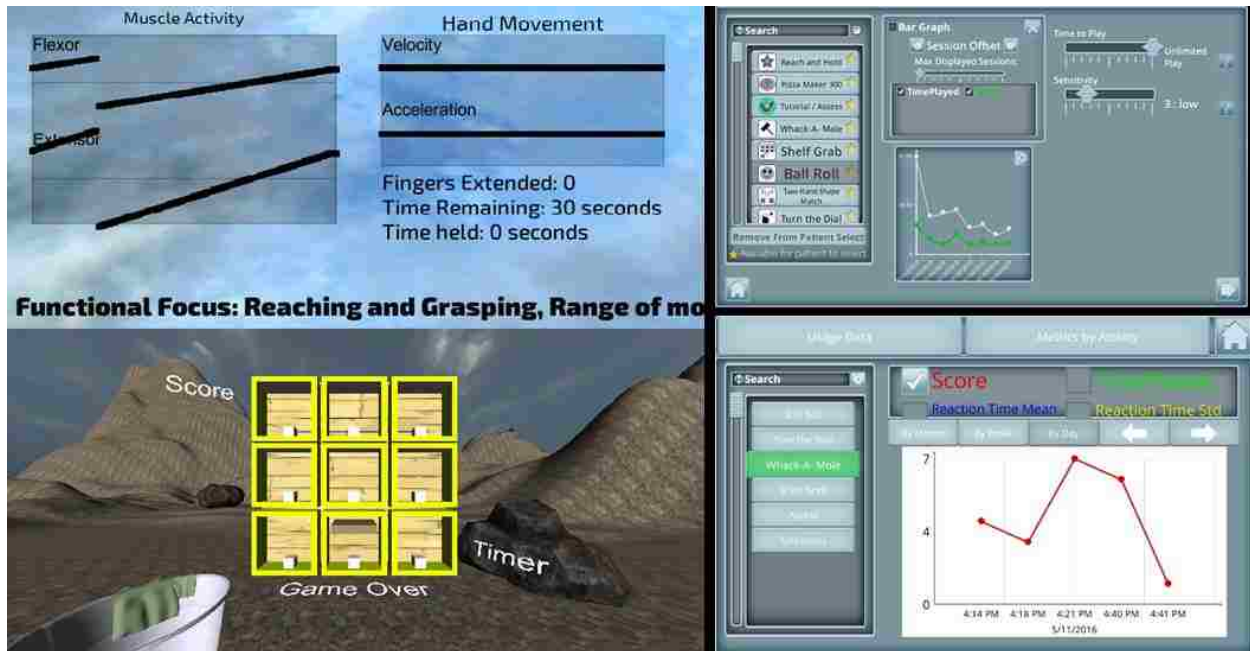
Figure 4.2.7. Data presentation methods throughout vHAB development. Left. Full data presentation framework for *Reach and Grab* in version 0.1.2. This full panel was viewable on the system screen showing muscle activity and key metrics updated in real time as the patient played the game. Aside from taking up half of the screen, the data were not useful for the average therapist. **Top Right**. Data presentation framework for version 0.4.7. Data were presented alongside game settings in a small panel in the center-middle. Options to control the graph were available as a popup immediately above the graph. Presenting data here was succinct and allowed for a one-screen therapist planning portal, but ended up being too small to be useful. **Top Left.** Current data presentation framework. This full screen viewer contains much of the same information as that in 0.4.7 but is larger and presents the data by time, not session.

4.2.D   *Hardware design*

While vHAB is primarily a software platform, the therapy we deliver would not be possible without the sensors and systems we use. By bundling the hardware with a pre-loaded computer we ensure that the system is easy to use out of the box. This is a key component of

106

what we have developed. Early feedback and research into other therapy devices showed clearly that having a software download that required external peripherals would never make it to the patients. Jintronix, for example, attempted to sell a Microsoft Kinect based therapy platform into facilities but required the therapists or patients (for home use) to purchase and set up the Kinect on their own (P. Goodrich, Madrona Principal, Personal Communication May 2014). This delivery model was not successful and now Jintronix bundles all of their hardware together with pre-installed software. In following this model, in accordance with principle 6, we set out to create a bundle of hardware that is still easy to use and robust to user error. A full description of the final included hardware along with its specifications, capabilities, and connections is provided in Section 4.3.

4.2.D.1 Display and Computation

The first version of the vHAB system was designed to use a virtual reality (VR) headset (Figure 4.2.8). Immersive VR offers many benefits to the vHAB system that are difficult to describe to a naïve VR user. Using VR with the vHAB system allows users to feel as if they are directly controlling the game object as opposed to moving their hand which in turn controls a virtual object. Further, VR removes a patient's affected limb from their view. This allows the vHAB system to display function that may be greater than their actual ability to drive progress. These benefits, however, are not fully explored as the use of VR currently violates our sixth design principle. VR headsets are expensive, even two years after we first prototyped a VR system. This makes delivering the full bundled system to a care facility very difficult and a home user nearly impossible. Further, VR headsets are difficult to put on by a healthy user, let alone a patient suffering from a neuromuscular disability. Even if patients have help to put on the system, they may reject it for cosmetic reasons. One early piece of feedback we received when

107

testing our VR prototype was "I'm not putting that thing on my head" from a user who did not

want to mess up her hair. VR is a hardware display technology we are actively following for

future improvements to the vHAB system, but for the current version does not represent a good

design decision. Using a tablet system or desktop screen provides a great experience for users

that is more cost conscious and accessible.

Figure 4.2.8. VR Headset used in prototype version of the vHAB system. In this example, the user is

playing an early version of a pinching game using the VR headset.

The choice of computers for the vHAB system is primarily driven by the availability of

USB ports for our peripherals. Significant testing was performed on multiple tablets to determine

the most cost effective option for the end-users. Many tablets do not have USB ports and others

use processors that are not fast enough to provide enough data for our analysis or a smooth game

experience. This led to a tradeoff between cost of the hardware and user experience. The most

expensive tablets would provide the best experience, but a lower cost tablet may lead to lags in

the game controls or sparse data. In the end we decided to utilize the middle end Surface Pro

tablets from Microsoft, but are continuously evaluating new technology as it becomes available.

4.2.D.2 Peripherals

The peripherals are the leading drivers in vHAB's hardware functionality. Traditional computer peripherals (ex. Mouse, keyboard, game controller) are efficient and ergonomic input modalities, but they do not capture the actual movement of the hand that is driving the control. Input modalities that more closely mimic physiological function have become more prevalent alongside the adoption of commercial virtual reality and gaming systems. For example, the Microsoft Kinect has seen widespread use in therapy [25] due to its ability to accurately track body posture and limb positions.

For our system, we needed an accurate representation of the hand's movement, which large camera-based systems such as the Kinect do not provide. From the beginning of the system's development we have used the Leap Motion controller to track the hand kinematics. Hardware changes relating to the Leap primarily revolved around placement and use of the sensor as opposed to its cost or capabilities. This hand movement tracking provides the "what" of the movement, but does not provide the how. The relationship between hand movement and muscle activity becomes more difficult to predict with neuromuscular traumas. One common complication after a stroke, for example, leads to decreased muscle control complexity, as patients work to compensate for spasticity with over-exerted movements [37]. Measuring the muscle activity of a patient throughout their recovery provides insight into "how" a patient is moving. By combining the muscle activity with kinematics, we can immediately verify the movements the patient is making without supervision. This allows us to make comparisons of muscle activity across activities based on the precise movements as opposed to second hand therapist records. Our muscle activity recording systems, however, have changed dramatically along the way.

109

Hand Motion Sensor – Leap Motion

The main challenge we have had in the use of the Leap Motion sensor came from its inherent novelty to our target end-user. Very few people have used optical motion trackers, such as the Kinect, outside of gaming enthusiasts and those few who have undergone new therapy sessions using these systems. Further, people are used to interacting with a computer with their hands, but always through touch. The Leap requires people to hold their hand about six inches above two stereoscopic cameras embedded within the sensor. In many cases, however, people would either place their hand directly on the sensor, like they would for a computer mouse, or they would pick it up and wave it around. To address this, we designed a plastic holder for the Leap to make it seem more permanently grounded. This holder extends the Leap's physical footprint on the table, while making it more cumbersome to hold. Our current 3D printed prototypes seem to have helped account for these user errors, but significant testing is still underway.

An additional problem we saw in development was that therapists often placed the sensor too close to the display screen. This would prevent patients from being able to reach forward far enough to reach targets in any reaching style game. To assist therapists in the setup of the system, we designed a custom mousepad with outlines of the ideal peripheral and tablet placements.

Muscle Activity Sensor

The first vHAB prototype used the Biometrics Muscle Activity Sensing platform, a $20,000 clinical-grade system that could record precise muscle activity at a high data rate (1-2 kHz). This system required sticking electrodes onto the patient's forearm where the electrodes

110

were attached to a bulky wireless battery pack that would sit awkwardly on the table. Aside from the cost and obtrusive physical qualities of the system, the system required that the electrodes be placed in the same orientation and position for each use in order to compare data across sessions. While this may have been possible to achieve in a development setting with careful planning, it was not going to possible at wide scale use by untrained therapists. This system clearly violated principle 6, so we set out to find an alternative.

At first, we set out to create our own muscle activity sensing sleeve. Fair progress was made towards this goal, but part of the way through the development a commercial alternative named the Myo Armband, by Thalmic Labs, was released at a $200 price point. The Myo records eight channels of bipolar muscle activity using equally spaced electrodes set within a plastic armband. The armband transmits data wirelessly through Bluetooth Smart and contains a rechargeable battery. However, the data recorded by the Myo have only eight-byte resolution and updates between 50 and 100 Hz, depending on the required transmission distance. This is a significant downgrade from the clinical system used in our prototype, but represents a good tradeoff in cost and ease of use. The cost, however, still may be prohibitive for some of our end-users and is not normally included in the home version of vHAB. We decided, based on customer feedback and internal analysis, that having muscle activity data for the average home user was not as important as the standard range of motion data gathered by the Leap alone, and therefore did not justify the increased cost.

4.3 SYSTEM DESCRIPTION: HARDWARE COMPONENTS

The hardware systems used to support the vHAB software consist of several parts that encompass both the computation hardware and physical sensors as well as the peripheral hardware accessories that provide enhanced usability and clarity for system setup in the clinic and at home (Figure 4.3.1). Each of the components contribute to system data requirements or clinical usability.

Figure 4.3.1. Block diagram of vHAB hardware components and image of assembled Home system. Individual hardware pieces are connected through a combination of wired and wireless methods while accessories aid in system setup and use.

4.3.A *Tablet Computer Hardware*

The vHAB system runs on the Microsoft Surface Pro line of tablets. Tablet computers are an important design choice for use in the clinical rehabilitation setting. Therapy gyms are crowded and often therapy is provided in a variety of settings including standing in gym, sitting in gym, and bedside for patients who are unable to travel to the therapy gym. Tablets combine

112

the necessary portability and small size factor while not sacrificing computational power. The Surface tablets run full Windows Operating Systems, have touch screen input, support Intel i5 processors, and have a dedicated USB 3.0 port for connecting to peripheral sensors. For device implementation all standard software is removed from the tablet prior to use to create a clean slate for the vHAB installation.

4.3.B *Leap Motion Kinematic Sensor*

The Leap Motion is a commodity camera system designed to use hand gestures to control objects while engaged in a Virtual Reality Headset environment. The system uses stereoscopic visual and infrared time-of-flight cameras with proprietary low-resource overhead to output computational estimations of up to two hand positions over the hemispherical sensor capture area. The outputs of these hand positions are noted in figure 4.3.2. The positional accuracy of these outputs has been documented at 0.2mm [38]. The positional data are output at 120Hz over a USB 2.0 or greater connection. Because of the nature of the stereoscopic infrared illumination there are some limitations to device performance that could potentially affect system use in certain environments. In the presence of direct sunlight or bright overhead lighting, the data output rate may decrease or some hand poses may not be recognized.

Figure 4.3.2. Example kinematic output from Leap Sensor API [39]. The Leap sensor provides direct access to individual joint positions (green spheres) as well as direction vectors (grey bars) of up to two hands at a time.

4.3.C   *Leap Motion Holder*

The Leap Motion camera is a symmetrical device with a USB output cable. Early patient and clinical beta testing exposed the need for a carrier device to properly orient the Leap Motion for naïve clinical users and patients alike. We designed a case for the sensor that properly oriented the sensor in front of the user and provided additional information on hand positioning over the sensor (Figure 4.3.3). A 3D model of the holder was created in the Blender development environment (blender.org), exported to the CatalystEX 3D printing software, and printed to size using a uPrintSE laminar 3D printer (Stratsys, Eden Prairie, MN).
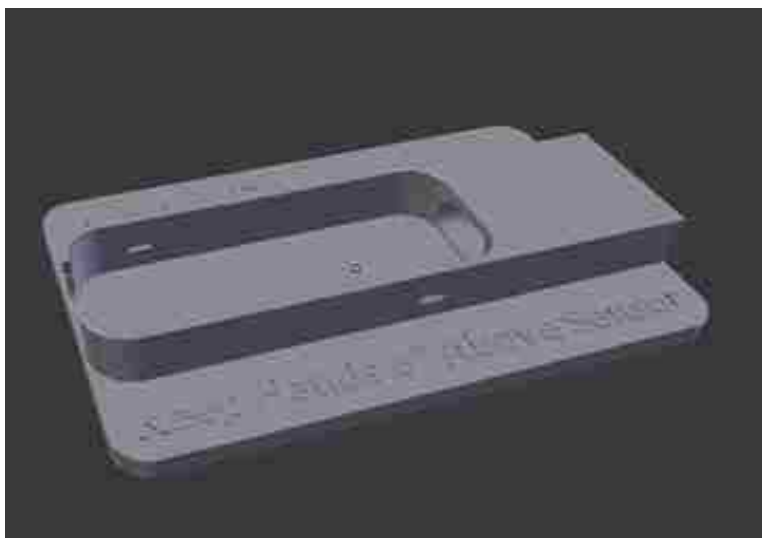
Figure 4.3.3. 3D Rendering of computer model used for holding Leap Motion Sensor in the correct orientation. The holder assists in device placement and prevents patients from picking up the sensor.

4.3.D *Myo Armband*

One of the fundamental barriers to technology adoption in the rehabilitation space is complexity and time spent in preparation. The Myo EMG Armband is a commodity 8 channel dry, bipolar surface EMG system equipped with simultaneous gyroscopic and accelerometry-based data streams (Figure 4.3.4). The primary function in commercial applications is the gesture-based control of assigned functions that plug into existing software modules to provide a new input modality to computing devices. The device communicates over the Bluetooth Low Energy (BLE) protocol. The data from the Myo consist of eight raw EMG channels sampled at 200Hz, accelerometer and gyroscopic data sampled at 50Hz, and pre-classified pose data. Our application centers around the EMG data acquisition so all other data streams are ignored during gameplay. The primary design consideration in choosing the Myo as a data acquisition device is the user friendly and adjustable armband which can be easily cleaned between patients while providing high fidelity EMG data. We have developed algorithms that provide a rotational

normalization of electrodes to account for inconsistent placement between sessions. This information will be detailed elsewhere.

Figure 4.3.4. Myo armband with exposed sEMG contacts. The design of the band allows for use on arms with circumference between 7.5 and 13.4 inches.

4.3.E   *Setup Placemat*

The vHAB environment facilitates full reaching and grasping motions. To facilitate the correct placement of the sensor to capture the full (~16") arm extension from a neutral position we created a setup placemat that specifies the correct orientation and spacing of the hand sensor. The design includes placement instructions for the screen or monitor as well as the hand sensor. It also includes written instructions for interacting with the hand sensor and directions to a web site for any technical issues. The design was created in Adobe Illustrator and realized on a custom-printed 11x17x1/16" foam mouse pad.

4.4   SYSTEM DESCRIPTION: SOFTWARE COMPONENTS

4.4.A   *User Interface and User Experience*

A key component of the vHAB system is a seamless user experience for both the patient and the therapist. Early in the development a large emphasis was put on creating games that were intuitive to start playing, but as we kept adding new features and additional settings it was clear that we needed a UI to wrap it all together. Further, as we shadowed therapists and gathered early product feedback we began to see the importance of patient-specific data, which led to the need for a therapist-specific patient management system. In the following sections we describe the functionality of the user interface and the design choices made to reach the final version.

The front end user interface for facility use consists of seven screens: A1: Therapist Management, A2: Patient Management, A3: Patient Dashboard, A4: Settings Management, A5: Data Viewer, A6: Assessment Select and A7: Game Select (Figure 4.4.1). The Game Select also contains an optional engagement Wrapper (A8). The interface is organized to account for multiple system use cases. For security and privacy reasons we hide a majority of the patient specific information behind both a therapist login in A1 and a patient select in A2. Further, if a therapist primarily uses Assessment modules, but rarely the Games, the dashboard will allow them to see the information most relevant to them. The home version of vHAB consists only of screen A7 and the Wrapper. This slimmed down version provides a much simpler interface for patients to navigate at home, even if their movement is impaired.
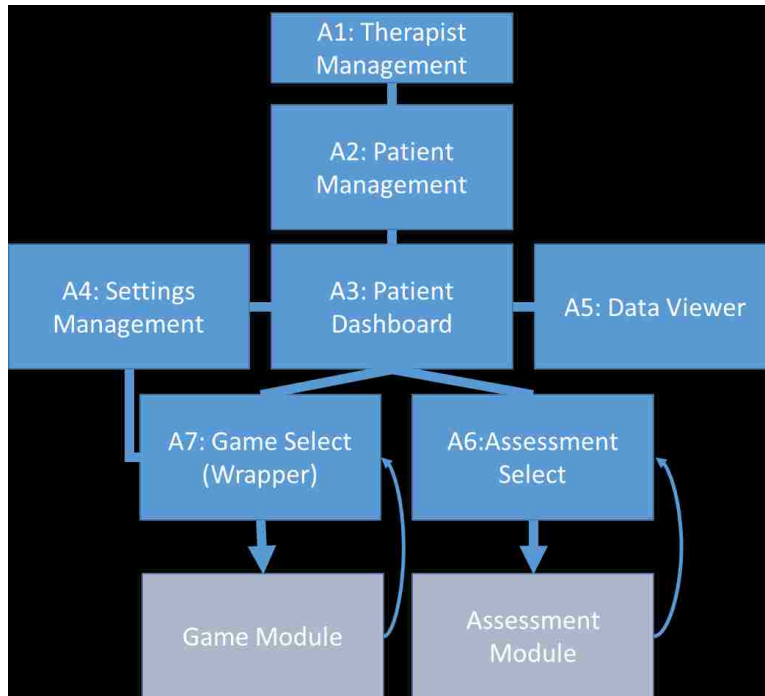
Figure 4.4.1. Screen flow for vHAB User interface. Blue boxes represent individual user interface screens used for managing user information and settings prior to using a game or assessment (Grey boxes). Lines between flows represent a bidirectional option to change screens, whereas the arrows represent a one way flow.

The 2D assets for the user interface were primarily designed through a collaboration with the Digital Futures Lab at University of Washington Bothell. During this design phase we took our initial user interface concepts and molded them into the final version seen below. This process created a few overall design guidelines such as color, element positioning, and margin styles. Asset creation was handled primarily in Adobe Photoshop and Balsamic. Photoshop was used in the image creation, while Balsamic was used for prototyping the organization of the individual components (Figure 4.4.2). Further image editing was performed in GIMP, an open source editing program, or Adobe Illustrator.
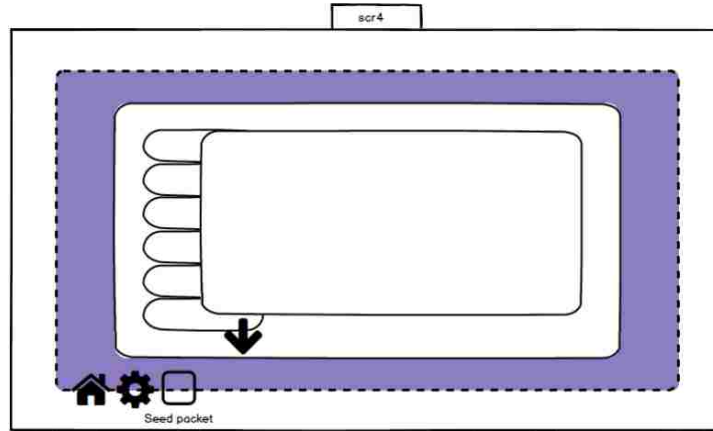
Figure 4.4.2. Early Balsamic markup of the Game Select Screen. Balsamic was useful in prototyping what a UI screen might look like early on in the design process.

The creation of each screen was accomplished by assembling the 2D assets alongside scripting components in the Unity game engine. The Unity engine contains prebuilt UI functions through a set of Canvas elements. Each element contained functions such as "OnClick" handles for "Buttons" and "On Value Changed" handles for input fields. These prebuild functions were useful for creating basic functionality, but custom scripts were required in each screen to build a working interface. Navigating between screens was accomplished through a set of custom "screen navigation" functions and Unity's built-in scene manager. These screen navigation functions also handled information between screens by either saving variables to local or system memory.

The vHAB UI is also built for multiple screen resolutions. While we currently are deploying primarily on the Surface Pro line of tablets, other use cases may demand support for more resolutions. Within the Unity Canvas framework, all UI elements can be set to scale their widths or heights up or down in relation to a base resolution. When each UI element is placed on the screen in Unity, we also set a set of scaling parameters that dictate how it will scale. This

119

solution is far from perfect. Some elements scale better than others, mainly based on how they were created, which results in empty space on the screen, or oddly pixelated items. It does however, ensure that all of the information is available to the user, regardless of their screen resolution.

Key Concepts

Patient User Name (pUN): The pUN is a display handle used in screen A2 to select the correct patient to work with. We recommend that the pUN is not the patient's real name to limit the PHI we are storing.

Patient Identification String (pID): The pID is an additional step in securing the patient's PHI. The pID is created when the patient is created using the "New Patient" functionality in screen A2 and is a unique random MD5 hash string value consisting of 25 character values. After a patient is selected, all data are stored in a folder tied to the pID. Further, all global usage data and summary statistics are stored with the pID as a tag. This security step prevents someone from accessing the root directory of the system and gathering a patient's data by name or other identifying factors.

Patient lists. Each therapist has a list of patients that they are currently working with. This list contains a mapping between the pUN and their pID. This list also contains any additional PHI for that patient such as demographic or injury information. This list is encrypted at rest and can only be decrypted using the specific therapist's login information.

The therapist management screen is the entry point for all use of the vHAB system. Early

on we decided that we wanted each therapist to have their own list of patients. If the system

housed a shared list across an entire therapy facility, the patient list would quickly become

overwhelming and there would be increased chances for duplicate patient User Names (pUN).

Therefore, the first thing we wanted a therapist to do was access their unique patient list and set

up the system to show only those patients. Further, as we begin looking forward towards the

secure storage of PHI, we knew this list of patient information needed to be secure. This led to

the need for both therapist User Names and associated login credentials. The therapist

management screen consists of 5 distinct features: i. Therapist Registration, ii. Therapist Login,

iii. Password Reset, iv. System Feedback, and v. Exit (Figure 4.4.3).

Figure 4.4.3. Therapist Management screen A1. This screen has five main buttons that serve the

functions described below. The "Suite" text on the top of the screen represents the version of the system

(to be distinguished from a "Home" build).

i. **Therapist Registration.** New therapists must first register an account so that we have a place to store their patient lists. Here we ask for their intended user name, password, and email. Upon pressing the Register button a popup will appear asking for these values and lead them through the registration process. This information is hashed and stored on the local system for future lookup. The system also checks that the user name does not already exist for that facility and that it does not contain any special characters.

ii. **Therapist Login.** To get the patient list for the therapist, the therapist must enter the user name and password they used during registration. These values are then rehashed and compared to the stored values on the local system. Upon successful login the therapist's patient list is loaded into memory and screen A2 is loaded.

iii. **Password Reset.** In order to recover a lost password the therapist must enter the email and user name they used during registration. Upon pressing the Password Reset button a popup will appear asking for these values and leading them through the reset process. These values are rehashed and compared to the stored values on the system. If the email and username are found then a temporary 8 character key is generated and sent to the email. This key is valid for 15 minutes during which the therapist must enter the key into the local system. If the keys match, the therapist has an opportunity to overwrite the stored password with a new password. If 15 minutes have expired, the key is removed from local memory and the process must be restarted. This security procedure aligns with industry standard protocols.

iv. **System Feedback**. Using this feature, a therapist can directly send feedback and comments to the developers of the vHAB system. Upon pressing the Feedback button a popup will appear leading them through the feedback process. We use an SMTP email client setup through a special email to automatically send an email containing the information entered into the text fields. Further, we send the device name, IP address, and facility name if available so that we can send a response if requested and identify potential issues with a group of systems.

v.   **Exit.** This is the main way to close the system. Exit functionality is crucial in ensuring all processes have properly exited and specific variables are reset. In some use cases, we disable the exit button since the computer will only be used to run the vHAB software.

4.4.A.2 Patient Management

When a therapist logs in correctly, their list of patients is loaded into memory. This list then populates the patient list panel with a button for each patient. If a therapist is logging in for the first time, or has not yet created a patient, this panel will be empty. Simple patient management functions are available in this screen to create and delete patients. In this screen we attempt to limit the amount of PHI displayed at any given time. Simple information, such as pUN is available in list format, but a patient's name must be clicked in order to view the basic demographic information. We made this design choice to accommodate multiple workflows. In an ideal world, a therapist would have set up a patient, with the appropriate game settings and activities prior to seeing the patient. However, in the case where a therapist session changes and they decide to use vHAB, there needs to be a strong emphasis on UI efficiency and PHI security. Thus, this screen has six main functions i. Create new patient, ii. Select a patient, iii. Delete selected patient, iv. Load selected patient, v. sort patients, and vi. Logout therapist (Figure 4.4.4). Additionally, the therapist name is displayed in the top right corner as a verification that the therapist login process was successful.
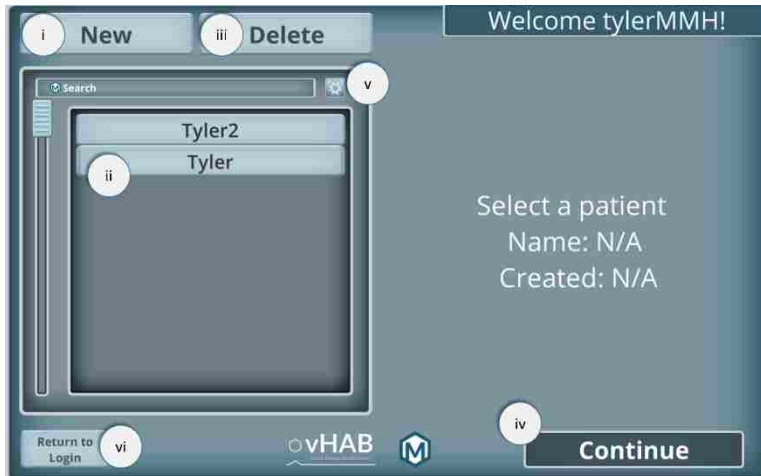
Figure 4.4.4. Patient Management screen A2. This screen has six main buttons that serve the functions described below. The "Select a patient" text on the right of the screen represents the currently selected patient information. If no patient is selected it will display as above.

i. **Create new patient**. Upon clicking the "New Patient" button, the therapist will be presented with a popup that will lead them through the creation of a new Patient. In this popup, the therapist will be prompted for the pUN, as well as optional demographic information such as age, gender, and handedness. Finally, the therapist will be prompted to enter any additional notes on the patient. These notes are available in the Patient Dashboard (A3) screen. When the prompts are completed, the system takes the information and appends it to the therapist's patient list and then creates a patient select button in the patient select panel. Specific checks are made on the pUN upon creation to ensure the pUN is unique for the therapist and that it contains no special characters.

ii. **Select patient.** A button is added to the patient select panel for each patient in the therapist's patient list. When clicked, the screen will be updated with that patient's information and the "Continue" button (iv) and "Delete" button (iii) will become interactable.

iii. **Delete selected patient.** At times, a therapist may create a test patient or they may simply have too many patients in their list. The delete functionality removes the patient's pUN and pID from the therapist's patient list; however it does not delete the de-identified data behind the pID folder. Saving

this data provides the possibility of undoing a delete operation and allows for the global usage data of a system to maintain accuracy. Upon clicking the "Delete" button, the therapist is prompted with "Are you sure you would like to delete (pUN)?" before the operation is complete.

iv. **Load selected patient.** This button becomes interactable when a patient is selected (ii). When clicked all of that patient's information will be loaded into memory from their pID folder. When the information is loaded, the therapist is taken to the Patient Dashboard (A3) screen.

v. **Sort patients.** The small cogwheel icon and the "search" input field allow for sorting of the patient list. Searching moves the best string matched pUN to the top of the patient select panel. Clicking the cogwheel icon allows for alphabetic sorting (A-Z or Z-A) and by the time created. Patients are initially sorted by time created, with newest patients being pushed to the top.

vi. **Logout therapist**. This button returns the therapist to the Therapist Management Screen (A1). Additionally, all patient information is cleared from program's memory. At the end of each session the therapist should logout to ensure system security.

4.4.A.3 Patient Dashboard

The patient dashboard is the landing page for the patient. Here, the therapist loads the next screen based on what they would like to do in their therapy session. In the development of the system, we decided to split the system between Assessments and Games. This dashboard allows a therapist to clearly know whether they are doing a game or assessment with their patient. The patient dashboard is also the home for a patient's notes and contains functionality for adding new notes to the ones already created in the "New patient" functionality. Loading the patient dashboard screen officially starts a "Session" which is a key component for the data saving structures discussed below in D. Data Management. The Patient dashboard screen consists of 6 main functions: i. Load Assessments, ii. Load Games, iii. Load Settings Management, iv. Load Data Viewer, v. Add new notes, and vi. End Session (Figure 4.4.5).

125

Figure 4.4.5. Patient Dashboard screen A3. This screen has six main buttons that serve the functions described below. The "Current Patient" text on the top of the screen represents the currently selected patient. This text serves as a verification that the proper patient was loaded into memory from the patient management screen.

i. **Load Assessments.** Loads Assessment Select screen (A6).

ii. **Load Games.** Loads Game Select screen (A7).

iii. **Load Settings Management.** Loads Settings Management screen (A4).

iv. **Load Data Viewer.** Loads Data Viewer screen (A5).

v. **Add new notes.** Clicking the '+' icon in the notes section creates a popup for the entry of a new note. Notes are stored as a text document in the pID folder and contain both a date-time string and the notes string. Notes are added to the notes panel with the most recent notes at the top. Notes are loaded into this panel when this screen is loaded. The notes panel is scrollable to view past notes.

vi. **End Session.** Pressing this button ends the current patient session and returns the therapist to the patient management screen (A2). This also removes all of the current patient's information from the program's memory.

4.4.A.4 Settings Management

Each game activity has multiple settings that affect how the game is played in terms of difficulty, length of play and required movements. These settings can all be modified in this screen before a game is loaded. This allows a therapist to pre-plan a therapy session with specific settings for that patient. It is important to note that settings for all games are patient specific and stored as a key-value pair text file in the pID folder. When a new patient is created, they are assigned a default set of settings from a *GlobalSettings.txt* file. Each time settings are changed the key-value pair in the patient's folder is modified. This screen also contains information on each game and the ability to add or remove a game from the game select screen (A7). This screen contains 3 distinct panels each with multiple functions as well as basic screen navigation functions (Figure 4.4.6).

Figure 4.4.6: Settings Management screen A4. This screen has three main panels, two buttons and a toggle that serve the functions described below.

i.   **Game Select Panel.** This panel contains all of the game activities available to the therapist. While we currently have more than 10 games, we may choose to present less games to a therapist based on their

functional focus. In this panel each game with a filled in star is made available in the Game Select screen (A7). Removing games from the patient's view was requested from early system feedback. Therapists were concerned that a patient may be distracted by too many games, or may want to play a finger movement related game based on its name, when they are supposed to work on wrist movement. Clicking any of these buttons updates the Game Information Panel (ii) and Game Settings Panel (iii) with that games information and settings.

ii. **Game Information Panel.** This panel displays the name of currently selected level along with a tutorial video and task specifications. The tutorial videos are aimed at the therapist and show how to use the system with a patient. The task specifications relate the functional tasks associated with the game, such as the movement types and body parts used in the game.

iii. **Game Settings Panel.** Each Game has many different settings, described in full in 4.4.B. Games below. In this screen all of the settings can be modified before a game is played. This is useful for the therapist if they would like to pre-plan an entire therapy session without having to change settings in game. Settings in this panel are presented as sliders, toggles, and Boolean buttons. Sliders represent integer or float value settings, whereas both the toggles and Boolean buttons represent 0 or 1 values. Boolean buttons are used where there may be only two choices (such as left or right hand) but a traditional toggle option does not make sense.

   Each setting is also populated alongside a '?' icon. Pressing these icons produces a popup panel describing the setting and its functionality. We also provide suggestions on how to use the setting appropriately with different patients.

iv. **Wrapper Toggle**. This toggle turns on and off the wrapper component of the game select menu. The wrapper is useful for patients using the system over multiple sessions, but may be a hindrance if the patient is not using it.

v. **Screen Navigation Buttons.** The Home icon returns the therapist to the Patient Dashboard screen (A4) while the arrow button takes the therapist directly to the Game Select screen (A7). If the Game

Select screen is loaded this way and a game was selected in the Game Select Panel (i) that game will

be preselected in the Game Select screen.

4.4.A.5 Data Viewer

One of the major advantages of vHAB over other rehabilitation systems is the inclusion

of detailed metrics and analytics that are calculated automatically during each game or

assessment. These metrics are detailed further in Section 4.4.D below, but here we overview the

presentation methods for these metrics. The Data Viewer screen has two main panels, a Usage

Data panel (Figure 4.4.7 (Top)) where activities are organized by time, and a Metrics by Activity

(Figure 4.4.7 (Bottom)) where metrics are organized by the game or assessment activity. Having

these two different panels allows the therapist to start with the information that is interesting to

them. For example, an outpatient therapist who only sees a patient once every few weeks may be

interested in seeing what activities were done last time the patient visited and would use the

Usage Data Panel. On the other hand, an inpatient facility may do the same activities every day

and is more interested in documenting the progress a patient is making in wrist range of motion

from *Ball Roll*. Presenting data at multiple levels is an ongoing design decision.

As discussed previously, the vHAB system has many different end users, both from their

actual role in therapy and their level of interest in the data coming from the system. Simply

documenting how often a patient performed an activity may be enough for many users, but

completely mundane information for others. The current version was the product of many focus

groups and general discussion with experts in the field. Addition of other sorting methods, such

as a By Function panel, will likely be added in the future to allow for users to see all information

related to metrics such as "Speed" or "Endurance". These "rolled-up" metrics would contain

metadata from all calculations of related data such as "Time to trial" or "reaction times over a session". Additional steps on the data analytics side will need to be taken on this data to create these metrics.

When the data viewer screen is loaded, a data parser goes through all of the data stored under the pID folder and converts each data point into a graphablePoint class. A data point can be simple usage data such as time played, or complicated metrics such as wrist range of motion. These graphablePoints are stored in local memory in lists for quick access. These lists can be sorted and applied to the User Interface when the appropriate buttons are clicked. This method is very quick, but may become problematic with large amounts of data. Other data management options are being explored, such as only loading the most recent 5 days' worth of data, or storing the most access data in different structures in the pID folder.

The Data viewer screen consists of the two main panels as discussed above. Each panel has various functionality as discussed below (Figure 4.4.7).
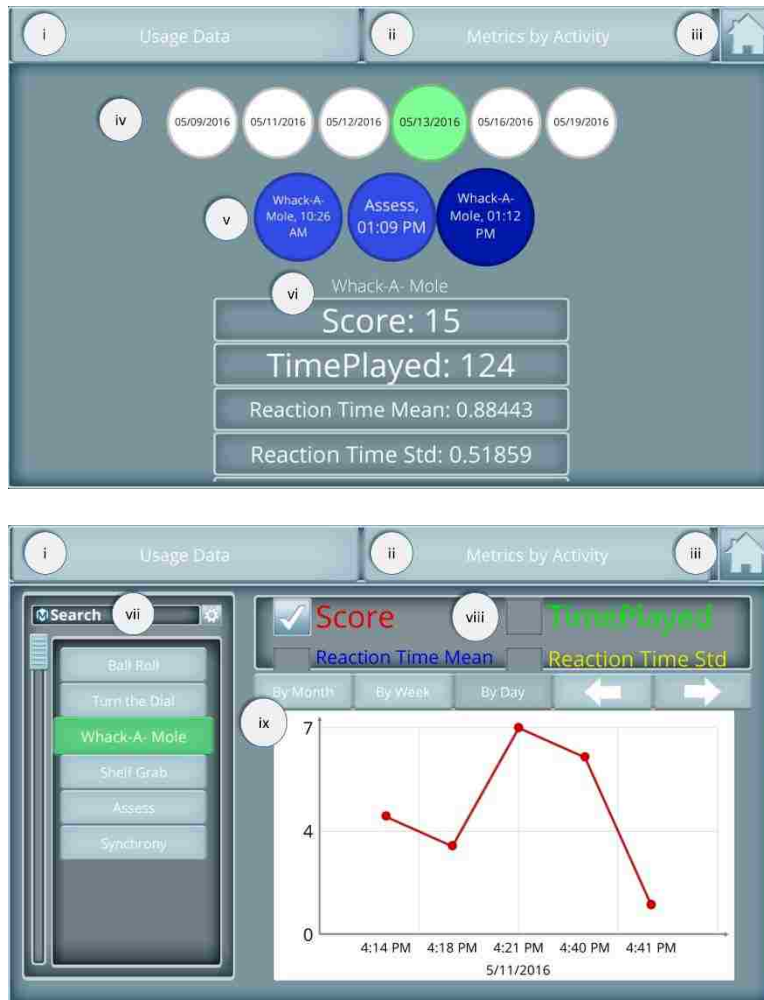
Figure 4.4.7: Data Viewer screen A5. **Top. Usage Data Panel.** This panel contains usage data and metrics sorted by time played. The system uses a node-like functionality, allowing additional information to be displayed when a given node is clicked. **Bottom. Metrics by Activity Panel.** This panel contains individual metrics sorted by game. In this example, the therapist has selected the "*Whack-a-Mole*" activity and is viewing the score over multiple sessions occurring on May 11, 2016.

i. **Usage Data Navigation Button.** Clicking this button will load the Usage Data panel. This will also reset the currently selected nodes, providing a blank slate for the therapist to navigate.

ii. **Metrics by Activity Navigation Button.** Clicking this button will load the Metrics by Activity panel. This will also reset the main graph and currently selected metrics.

iii. **Home Button.** Clicking this button will take the therapist back to the Patient Dashboard screen.

iv. **Date Nodes.** These nodes are populated based on a sorted and condensed "graphablePoint" list. Each unique date is given a node button that, when clicked, generates activity nodes (v) that occurred on that date.

v. **Activity Nodes.** These nodes are populated based on a sub list from the Date Nodes' list. Each activity on a given date is given its own node, even if that activity occurred multiple times in that date. Activities are sorted by time, with the most recent activities being the rightmost nodes. Clicking an activity node populates the Activity panel (vi) with all of that activity's information.

vi. **Activity Panel.** This panel is populated by all of the "graphablePoints" found for the currently selected activity. Each "graphablePoint" is given a small panel displaying the metric name and the value. Each of these panels is clickable and, when clicked, will preselect that activity type (vii) and metric toggle (viii) in the Metrics by Function panel (Figure 4.4.7 (Bottom)).

--- Metrics by Function Panel Only

vii. **Activity Select Panel.** This panel contains a list of all of the activities that the patient has done. If the activity is not visible on this graph it means that either the activity has not been played, or that no metrics were available for that activity. This functionality primarily comes into play if someone selects an activity, but for one reason or other cannot complete the activity. Clicking an activity in the panel will create a list of all of the "graphablePoints" that came from that activity and generate the metric toggles (viii) for all metric types found for that activity.

It is important to note that not all instances of that activity will be guaranteed to contain every metric type. Some metrics require minimum number of trials to compute, and therefore will not exist for all instances. This is handled by skipping the plotting of that point in the main graph panel.

Clicking a new activity will also clear all of the currently graphed points on the main graph, but

will not reset the time offsets or date sorting methods. Activities can be sorted by most recently played or can be searched through with the sort options at the top of the panel.

viii. **Metric Toggles.** These toggles are generated when an activity button is clicked in the Activity Select Panel (vii). Each toggle controls whether or not that metric is displayed on the main graph. Some activities contain more than 10 different metrics, which can get very crowded on a single plot. Each toggle is color coded to the line color of its corresponding metric on the main graph.

ix. **Main Graph.** The main graph displays the metric data for every checked Metric Toggle. The main graph has three main display modes: "By Month", "By Week", and "By Day". Each of these modes changes the x-axis values of the graph to the appropriate time span. The design of this view came after seeing the challenges most therapists had interpreting an early version of the data viewer that plotted data "as-available". The old method simply labeled each point with the date and time it came from, but this resulted in a non-linear data set.

The challenge with the current method is that it does not allow for multiple data points to exist for a given day in the "By Month" or "By Week" views. To fix this, we average all of these points and display a slightly darkened data point to convey that the point is averaged. Clicking these darkened points will load that day in the "By Day" view, where multiple points can be displayed.

Time navigation arrows (top right of the main graph) allow the therapist to go backward and forward in time. These arrows only allow the therapist to go back in time to the first date where data exist. However, if data do not exist for a given week, this empty week will still be displayed in the "By Week" view. This design choice was made for navigation consistency, but may be revisited in the future.

Graph assets were developed with GraphMaker [40], a third-party Unity asset. GraphMaker provides a framework for placing data points in the proper locations on a screen of varying sizes and allowed greater flexibility when designing the system.

4.4.A.6 Assessment Select

In early builds of the system we had a single game called vAssess. This game contained about five minutes of preset activities that were designed to assess hand function in one simple test. We learned, however, that many users (both therapists and physicians) were not interested in all of the activities included within vAssess. Instead of creating multiple games that addressed each of these user's desired activity sets, we created a modular assessment platform that allows the user to customize their vAssess experience by selecting only the measurements that are relevant to that specific patient and combining them into a seamless game session. To date, we have created ten different modules consisting of range of motion tests, strength/fatigue tests, and questionnaires. These modules are described in detail in Section 4.4.C below.

Managing this customization, however, required a special screen to create, load and save these custom lists (Figure 4.4.8). The Assessment Select screen contains multiple features and is currently undergoing major changes to the framework to make it easier to use. The current system prompts the user to create their custom list by dragging and dropping modules into a list. The user can also load preset or saved lists for speed. The current system, however, is not optimized for therapist workflow, especially as we continue to add modules. Future iterations of the system will include better sorting methods for the modules (such as by type, or by functional relevance), and will likely start the user at a load module set screen to make starting an assessment faster.

Figure 4.4.8. Assessment Select screen A6. This screen allows for the creation and loading of custom lists of assessment modules. Users can drag and drop modules from the left column listing all the modules to their custom list on the right. In this example, a user is dragging a module over to their custom list. The hand-cursor represents the user's interaction point (either a mouse click or a screen touch). The green box shows where the module would be placed if the user let go of the BubblePopper60 module.

i. **Module List Panel.** This panel lists all of the available assessment modules that can be added to a user's custom list. To add a module to the list, the user clicks (or touches) and drags the module to the Selected Module Panel (ii). When the module is halfway across the screen, the target module slot will turn green. To add the module to that slot, the user releases the mouse (or touch) and the module will snap into place. Releasing a module before a slot has turned green will not add it to the custom list. Dropping a new module on a slot that already contains a module will remove the old module from the list and replace it with the new one.

Each module button also has an inset help button ('?'). Clicking this button will show a popup panel for that specific game. Each popup panel contains a tutorial video for that module and a brief

description of the module (Figure 4.4.9 (Top)).

This module list is also sortable with either string matching through the search function or alphabetically by pressing the cogwheel button.

ii. **Selected Module Panel**. This panel lists the user's current module list. When the user starts the assessment, these modules will be presented to the patient in the order they are displayed in this list. New slots can be added by clicking the "Add Slot" button at the bottom of the list. Slots can be removed by clicking an "X" button within the slot. This "X" button is only viewable when the user is not dragging a module. An empty slot will be ignored when generating the module list.

The list can contain multiple instances of the same module, in which case each module is treated as a separate instance of the activity for data analysis purposes. This allows for the comparison of a module before and after some other module. For example, a user may be interested in fatigue before and after a range of motion exercise.

Each module in this panel can also be clicked and dragged to rearrange the order of the list. Here, if a module is dropped on a slot that already contains a module, the old module is placed in the selected module's previous slot. This previous slot is indicated by a yellow tint on the slot while the module is being dragged. To remove a module from the list the user must move it to the opposite side of the screen.

iii. **Load Functions**. There are two options for loading preset lists into the Selected Module Panel. When a module list is loaded with either method, it can still be modified and then resaved.

The "Load from List" button will generate a popup (Figure 4.4.9 (bottom right)) prompting the user to select from a list of module sets. This list of module sets contains both pre-generated lists that we have created (designated with a star in the button) and custom lists that the user has generated and saved (iv). Clicking one of these module sets will display that modules set's individual modules to the left of the panel. The user can then load that module set or return to the main Assessment Select screen.

The load recent method will load the module set that was most recently used. Currently, this is user independent and relies only on a program memory variable. However, if a facility often uses the same module set this may be a quicker way to load their list. This feature will likely be replaced with a "Favorites" list that is the first thing the user sees to further increase workflow efficiency.

iv. **Save Function.** After a user has created a module set in the Selected Module Panel (ii) they can save their list to a local file for future use. Clicking this "Save" button will generate a popup (Figure 4.4.9 (bottom left)) that prompts the user to name their custom list. This popup also displays the modules they have selected as a verification step.

v. **Clear Function.** This button will clear the list in the Selected Module Panel (ii) allowing the user to start from scratch when designing a custom module list.

vi. **Start Button.** This button will load the vAssess activity scene with the modules in the Selected Module Panel preloaded.

vii. **Home Button.** Clicking this button will return the user to the Patient Dashboard screen (A3).
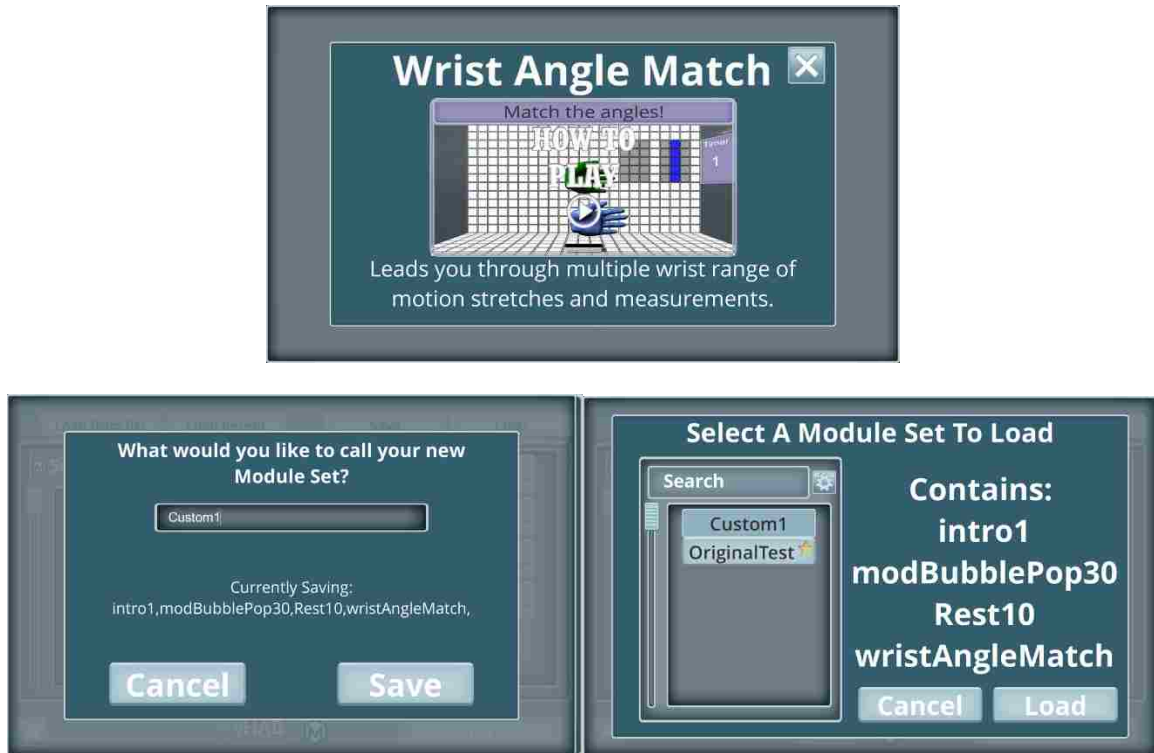
Figure 4.4.9. Additional Assessment Select screen A6 popup functions. Top: Help panel popup. This popup is generated when an Assessment Module button's inset "?" help button is pressed. It contains a module-specific tutorial video and description. **Bottom Left**: Save module list popup. This popup is generated when a user clicks the save button. This popup allows the user to save a custom list of modules for viewing in the load module list popup. **Bottom Right**: Load module list popup. This popup is generated when the user clicks the "load from list" button. This popup allows the user to load preset lists of modules as opposed to custom creating a list.

4.4.A.7 Game Select

The Game Select screen contains a list of the available games, a video tutorial, and a "Play" button (Figure 4.4.10). This screen is designed to be presentable to the patient as well as the therapist. An ideal therapy session would involve the therapist selecting the patient, modifying all settings, and loading this screen before they have started working with the patient.

138

In this way, the patient only sees basic information about the available therapy activities and can even interact with the UI in a useful way. This screen and the UI within each game have a different set of UI elements than previous screens, with different theme colors and styles. This is to convey the difference to the therapist, but also creates a more unified experience for the patient. This screen is also the home of the Garden Wrapper which is described further in section 4.4.A.8.

Figure 4.4.10. Game Select Screen A7. This screen allows the therapist and the patient to select the game activity they would like to do. After a game is played, the users are returned to this screen to select another game.

i. **Game Select List.** This list is populated with all of the games that are within the patient's list as defined in the Game Settings screen (Figure 4.4.6.i). Clicking each button will load the appropriate tutorial video in ii and allow that game to be loaded.

ii. **Tutorial Video.** Each game has a tutorial video used to instruct a naïve patient on how to play the game. While most of the introduction and training will be provided by the therapist, we found,

through various user testing experiences, that a short video can be a great starting point for both users. By watching a video before launching the game, the patient's focus is on learning the task as opposed to trying the task incorrectly. In viewing early patient – therapist interactions, many patients would ignore the therapist when the game started, likely because they assumed they could automatically figure out how to play. Since our paradigm is unlike what most patients would have seen before, these initial training steps are very important.

iii. **Play Button.** This button will load the currently selected game. Before loading the game, the backend software will load the appropriate variables for that game to ensure a smooth experience.

iv. **Navigation and Wrapper Buttons.** The home button returns the therapist to the Patient Dashboard screen (A3). The cogwheel button returns the therapist to the Game Settings screen (A4). The tree button shows the garden for the wrapper (Figure 4.4.11) while the seed button loads the seed select menu for the wrapper (Figure 4.4.12). Both of the wrapper functions are described further below.

v. **Current Seed Display.** This text element shows the patient what seed packet they are working towards within the wrapper. Showing this element here reminds the patient that they are working towards a goal, while also providing information to the patient if they would like to change their target goal.

4.4.A.8 Wrapper

The vHAB game engagement wrapper was a product of the collaboration with the Digital Futures Lab at University of Washington-Bothell (UWB). The wrapper is designed to tie all of the games together with a unified theme, despite the fact that all of the games are very different in both activity and art assets. The wrapper also provides a global scoring system which allows for motivation to be sustained across games and between therapy sessions. The wrapper is only available in the game activities and not the assessment activities. This was designed to drive use towards the games as exercises and assessments as one time check-ins of function and progress.

140

The wrapper can be turned off or back on at any time in the Game Settings screen (A4). Turning off the wrapper may be useful in short-term use cases where a patient may only be exposed to the system for a short time and the long-term engagement is not an important factor in using the system.

<u>Wrapper Mechanics</u>

Before starting a game for the first time, the user is prompted to select the goal they would like to work towards and all points scored within a game go towards this goal. In our wrapper, there are 18 distinct goals that the patient works toward. At the completion of each goal, the patient is rewarded by viewing progress in the wrapper's home screen in the form of animations and sound. Completing all 18 goals will unlock a button that allows the patient to view all of these animations in sequence.

To complete a goal, the patient must accumulate 100 progress points. In-game points are mapped to progress points based on an estimated eight week use of the system, though this timespan can be modified based on the system's use case. The mapping between progress points and in-game points is not constant, and is instead designed to change over time. Over time, more in-game points will be required to accumulate the same number of progress points. This is designed to engage patients early on while driving them to increased use and better performance over time. To create the current mapping model we made a few basic assumptions:

1) The amount of time played each day would increase over time
   a. Patient recovery and increased endurance will allow for slow, yet sustained increases over time.

141

      b. Working towards a goal may increase the amount of time they play if they are close to

         completing a goal.

  2) The number of points scored within each minute would increase over time

      a. Patient familiarity with the game's rules will provide early increases

      b. Patient recovery will allow for sustained increases assuming game difficulty settings are

         not changed

Further, we wanted progress to feel linear over time, so that putting in a day's work on day 36

felt as rewarding as day 1. To this effect, we build the model around finishing 25% of the total

progress every 2 weeks. We also assumed the patient was only using the system 4 days a week

and that they could start with a 1:1 mapping. The current model can be represented by:

$$\textbf{Eq. 4.4.1}. \qquad y = -0.0011x + 1$$

where y = points/progress point and x is cumulative minutes played. Currently, y is clamped to

not be less than 0.1 which works well in this model, but may need to be adjusted for longer time

frames. This model does not take into account the variability of points scored in different games.

It is also likely that this model does not optimize patient motivation over time. Further research

will need to be performed to reach this optimization point for all different use cases of the

system.

Wrapper Theme

      Based on user acceptance testing performed by UWB, we decided to use a Garden

metaphor for the wrapper. In this metaphor, the patient is growing a garden with their progress

towards recovery. The garden contains six distinct plants with three stages of growth. Each stage

of growth represents a goal and upon completion shows a brief animation of the plant growing. When a plant is completed, the patient is also rewarded with a new animal in their garden near the completed plant. A patient starts with an empty garden but over time can grow a full garden with six plants and six moving animals (Figure 4.4.11).

Figure 4.4.11. The Garden View of the Wrapper within the Game Select screen (A7). Left. An early stage garden with only stage 1 of the Hydrangea goal complete. **Right**. A completed garden with all six plants visible. Clicking the continue button here will take the user back to the main game select screen.

When selecting a plant (target goal) the user is presented with a set of seed packets to choose from (Figure 4.4.12). Selecting that seed will update the user's goal, but will not reset progress previously made towards a goal. Progress towards a goal is monitored within the Game's UI by showing a meter that fills with each progress point made. For each 20 progress points, the user is rewarded by a brief seed growth animation within the Game UI's progress panel. Upon completing a game, the patient is presented with a popup of summary stats (Figure 4.4.13). After a few seconds, this popup will move to the top right of the screen and show the patient their garden. If a goal was completed (i.e., 100 progress points were achieved), the animation and sound effects for the new plant will be displayed.

143

Figure 4.4.12. Seed select popup for the wrapper. This popup occurs within the Game Select screen
(A7) when the seed button is selected from the navigation panel. The navigation panel (i) changes based
on what popup is displayed. In the seed select popup the user can return home, go to settings, view their
garden, or load the game select popup. Seed packets are displayed (ii) in a fanned out format. They can be
navigated through by clicking on a given packet, which will bring that packet to the center, or by clicking
the arrows in the lower panel (iii). Clicking the center packet or the check box will select that packet for
the patient's goal.

Figure 4.4.13. The game summary popup for the wrapper. This popup panel will appear after each game if the wrapper is enabled. This popup displays the basic stats from the recently played game alongside the progress made towards the current packet.

Wrapper Use Prompts

At multiple points, a patient may not have a goal selected. This can happen with new patients and after a goal is completed. In both these cases, we prompt the users with a popup asking if they would like to select a seed packed. The popup can be dismissed and the game select view is shown, or they can accept and the seed packet select popup can be shown. Upon five successive dismissals with no goal selected the wrapper will be disabled. The wrapper can be re-enabled at any time through the Game Settings screen (A3).

4.4.B   *Games*

Games are the primary activity within the vHAB system. In our system a game is an activity in which the patient moves their hand or arm above the main kinematic sensor to control various digital objects on the screen. These objects can be direct representations of their hand with a three dimensional hand model, or more abstract representations such as a paddle or a dial.

145

When patients move their hands above the sensor, the mapped game object seems to move immediately due to the low latency of the sensor (<33ms) [39]. In all games we only represent a subset of the hand's full movement in the control of game objects. For example, we may only look at the rightmost hand's palm normal angle relative to a preset plane to control the angle of a game object. Lowering the dimensionality of this control space makes games much easier to learn for the patient and allows us to design carefully controlled experiences that are optimized for the sensor's abilities.

For the main vHAB system we have created over 15 games, with about 8 of them finished and the other games in various states of user testing and development. The process of creating games was a very iterative process, involving shadowing therapists, in-person interviews with field experts, user testing, and creative discussions with experts in user experience design. We have previously detailed the design process (Section 4.2), but it is important to reiterate that many games have changed drastically since the first version and many of the current games will continue to change. In the sections below, we describe the games in their current form.

Each game was created using the Unity game engine which provided a framework for placing 3D models, in game 2D UI elements, and assigning scripts to control the movement of these objects. 3D models were created primarily using Blender, a 3D modeling tool. Skinning the 3D models was accomplished using 2D image editors such as Adobe Illustrator and GIMP. 2D elements were created in the same way the main UI (Section 4.4.A) was created with the help of the Digital Futures Lab at University of Washington Bothell (DFL UWB). DFL also assisted in user experience design of some of the games. Audio elements for the games were acquired through a third-party Unity asset [41].

Each game possesses its own unique control, target generation, and trial structures. However, multiple threads are common throughout each game. Audio feedback is used both as a reward signal at the completion of a trial and as a cue element within a trial to show the patient they are making the correct movements. In addition to the main control elements, particle effect visual cues are used in varying circumstances to show in-trial feedback. These visual elements may draw a patient's attention to a given game object or signify the completion of an in-trial step. These audio and visual reward elements add a layer of fun and engagement that users expect from traditional games, while also providing crucial performance feedback.

Games consist of a set of trials over a set amount of time. A trial consists of presenting the patient with a target or objective and the patient performing the appropriate movement to match or move towards the target. Completing a trial gives the patient in game points which are then mapped to "progress points" as described above. This trial structure comes into play with the analytics structure described in section 4.5: Analytics Description.

Each game has multiple settings that are specific to that game's difficulty or control options. These settings allow patients of multiple ability levels to play the same games. For example, a patient with limited mobility in their wrist can still play *Ball Roll* at a high sensitivity setting. The sensitivity setting provides a multiplier for the movement of the game object. With a high sensitivity, a small movement of the wrist will move the paddle as much as a large wrist movement with a low sensitivity. We provide preset minimums and maximums for most settings to prevent the games from being unresponsive, but for the most part our settings ranges cover all levels of function. Some settings are more Boolean, simply turning on or off a given feature or game target requirement. Settings are also patient specific, stored within their pID folder, so that the therapist does not need discover the correct settings for a patient on each use. Future changes

to the settings structures will allow for automatic starting settings based on a first-use

assessment. Additionally, we have explored multiple options for automatically adjusting settings

as we see increased or decreased performance. At this time, however, most therapists requested

that they have direct control over the settings as they are learning how to use the system. Settings

for all games are summarized in Table 4.4.1.

Table 4.4.1. Settings for all games.

Setting	Function	Games used
Time to play	Controls how long the game is played. If set to the maximum value the game is in "Unlimited" play mode.	All Games
Sensitivity	Controls how much real world movement is needed to move an in game object. This value modifies preset remapping functions discussed below.	Ball Roll, Reach and Grab, Reach and Dwell, Pizza, Whack-A-Mole, Two Hand Shape Match, Pill Box
Min and Max Values	Dictates a range of values from which a game can generate targets	Turn the Dial
Toggles	Toggles have varying functionality such as specifying which fingers can be used, whether a patient is able to make a fist, or what types of targets are generated	Reach and Grab, Pizza, Whack-A-Mole, Two Hand Shape Match,
Thresholds	Dictates at what level of a value does a specific action occur, such as when in a grasp is the patient officially making a fist	Reach and Grab, Pizza, Two Hand Shape Match, Finger Position Match, Giant's Teeth

Each game has a common game UI (Figure 4.4.14) created in the same theme as the

Game Select screen (4.4.A.7). This UI serves multiple functions such as changing game

parameters and navigating between screens. The UI also displays relevant information about the

game state, such as score, elapsed time, and garden progress. This UI was designed to be

unobtrusive while still being easy to use on multiple hardware systems. One of the key

components of the in-game UI is the ability to modify settings while a patient is playing a game.

A therapist can make a game easier or harder based on a patient's ability without having to return to the Game Settings screen (4.4.A.4). Each setting for the currently loaded game is dynamically populated in a small popup at the bottom of the screen (Figure 4.4.14 right). A therapist experienced with the vHAB system can utilize this function to modify the game's difficulty while a patient is playing the game to subtly increase the difficulty and push for better performance. Each setting is also accompanied by a help "?" button that creates a small descriptive popup just above the settings panel.

Each game also contains built in pop-up tutorials (Figure 4.4.15). These popups show up the first time a patient is playing a game or anytime the '?' button is pressed in the in-game UI. These tutorials consist of a set of text prompts that are displayed within the game space. Specific requirements must be met to advance the tutorial. Some prompts are advanced with time, but others require specific in-game actions. For example, we want to continuously remind patients to keep their hand at least 6" above the sensor. The prompt reads "Place your hand 6" above the sensor" and an in-game tracker looks for a hand above the sensor before advancing the prompt. This interactive tutorial provides a good first experience for patients, while not limited experienced patients from exploring the space on their own.

Figure 4.4.14. In game user interface. Left. The standard starting UI using *Whack-a-Mole* as an example. **Right**. In-game UI with the settings panel enabled.

i.   **Game state display.** These panels show the current score and elapsed time of the game. The score value is updated at the completion of each trial. The time value counts up in the "Unlimited" play mode and counts down if a game time is set.

ii. **Game Control Panel.** These four buttons control multiple in-game functions. The circular arrow restarts the game. Restarting a game resets the game score and timer, but progress points are saved and all analytics are still calculated. The "?" button generates the in game tutorial popup (Figure 4.4.15). The cogwheel button generates the in-game settings popup (iv). The forward arrow finishes the game and loads the wrapper congratulations screen (4.4.A.8).

iii. **Goal progress.** This panel shows the accumulated progress points towards the currently selected goal. The meter on the right fills with each progress point earned. At each 20% fill mark, the seed image "grows" with an animation to signify a mark in progress.

iv. **Settings Popup.** This popup allows for the modification of settings within the game. Each setting can be modified by interacting with the slider or toggle element in the center of the popup. Clicking the inset "?" button generates a help panel describing the setting. Clicking the arrow button will show the next available setting.

150

Figure 4.4.15. In game tutorial popup. This popup shows up when a patient plays a game for the first time or when the "?" button is pressed. These tutorials are designed to help the patient learn to play the game in an interactive way.

Game Summaries

The sections below describe each game with its corresponding movements, control paradigms, and therapeutic relevance. We have selected 10 games that represent the wide range of tasks our system is capable of. Additional games are briefly referenced in B11. Other Games below, and are removed as they represent either an incomplete experience or utilize the same movements and control schemes but use a different visual. The games below are ordered in level of movement complexity. All games can be played with either hand automatically unless otherwise indicated.

Key Terms.

Sensor zero. A neutral position above the sensor (Figure 4.4.16) where x = z = 0, and y = a detectable distance (ideally >6").
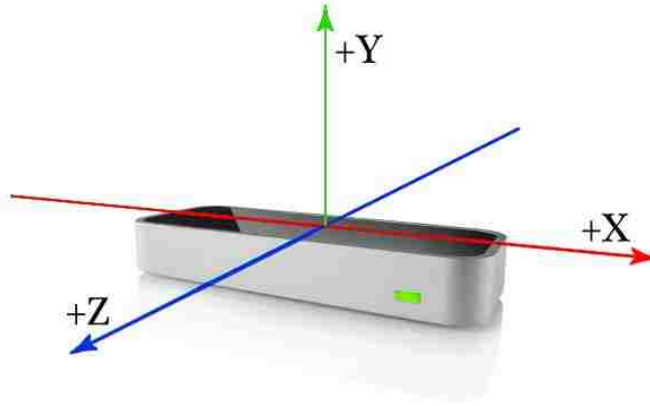
Figure 4.4.16. Leap coordinate system. The Leap motion reports measurements in three dimensions (x,y,z) as displayed above. Positive X axis is normally to the right of the user, positive Z towards the user, and positive Y, up. Figure reproduced under fair use from [39].

Remapping. We use a set of remapping functions to move between the patient's real position and game object positions on the screen. These set of remapping equations utilize preset values within the control scripts that translate a volume of real space to a volume of 3D game space. These remapping equations follow a similar pattern:

Eq. 4.4.2. $\quad \text{game}_{new} = \dfrac{(\text{leap}_{new} - \text{leap}_{min}) * (\text{game}_{max} - \text{game}_{min})}{(l_{max} - l_{min})} + \text{game}_{min}$

Leap minimums and maximums are set using the leap's sensor zero as well as estimated ranges, while the game minimums and maximums are set based on the game's design and required operating space. This function primarily applies to positioning, such as translating the x, y, and z position of the patient's real hand to their virtual hand in game. This remapping can be adjusted using a sensitivity 's' that decreases the measured range of the leap sensor without adjusting the game boundaries. By moving both leap$_{min}$ and leap$_{max}$ in Eq. 4.4.2 closer to sensor zero, a given leap$_{new}$ will produce a larger game$_{new}$ value.

Eq. 4.4.3. $$\text{leap}_{min} = \text{leap}_{min} * \frac{s}{10}$$

Lerp. A game development term used to smooth two values over time. The lerp functionality allows for incremental movement towards a target value to be automatically adjusted each frame. For example, if we want a game object to eventually end up at position (1,1,1) and it is currently at (0,0,0). We would not want to just change the position in a single frame since it would result in a visually choppy jump instead of a smooth movement. We ca not just move a set amount each frame since we are using a variable frame rate. With a properly setup Lerp function, we can move a variable amount each frame based on the "delta time" or the time since the last frame. This results in a smoother movement while also ensuring game objects interact with each other. The speed at which a value Lerps can be adjusted depends on the desired effect. Most Lerps are recursive, incrementing the current value towards the target value based on the frame rate.

Collisions. Each 3D game object has an invisible mesh around it to control interactions with other objects. When two meshes overlap, we can call preset functions that handle what each object does when it collides with another. Collisions are used for picking up objects, as well as detecting when a moving object has reached a target position. Some collisions are handled with a physics model and simply prevent objects from passing through each other.

Magnetization. The use of magnetization with introduced in Section 4.2.B.6. An object becomes magnetized to the patient's virtual hand when the hand collides with a specially set collision mesh surrounding the virtual object. An object will stay magnetized as long as its center position is within that collision mesh. The object will snap back to its original position if it leaves the collision mesh and the patient has not yet met the exit criteria, such as making a fist.

Symbols. Multiple symbols are used below in the equations representing game control

paradigms (Table 4.4.2). For any $\dot{}$ box, an uppercase letter represents a real world value, such

as the Hand Position ($\dot{H}$), while a lowercase letter represents an in-game object or a remapped

value.

Table 4.4.2. List of symbols for game control paradigms.

Symbols	Meaning
~	rotation as a quaternion
$\cdot$	Position as a 3 dimensional vector
$\rightarrow$	Direction between two positions as a 3 dimensional vector
$⌊⌉a \rightarrow b, t$	Lerp from vector, float, or quaternion 'a' to a new value 'b' consisting over time t
Δt	Time between frames

4.4.B.1 Ball Roll

Functional Movements: Wrist flexion and extension.

Objective: Move the paddle left and right to push the ball off of the table.

Movement Control Paradigm: The angle of the paddle is mapped to the direction of the

hand as a distance from the YZ plane. The direction of the hand $\vec{H}$ is calculated by extracting the

x component of the directional vector between the center of the palm $\dot{P}$ and the center of the

wrist $\dot{W}$ (Eq. 4.4.4). This value is multiplied by the sensitivity setting value's' and remapped to

game space (Eq. 4.4.5). This final value controls the angle $\tilde{h}$ of the paddle anchored closest to the

patient with a fast Lerp function (Eq. 4.4.6).

154

Eq. 4.4.4. $\quad \vec{H} = \angle \dot{P} x \dot{W}$

Eq. 4.4.5. $\quad h = \vec{H}(x) * \dfrac{180}{\pi} * S \qquad\qquad$ -80°$< h <$80°

Eq. 4.4.6. $\quad \tilde{h} = \{\tilde{h} \rightarrow (0, h, 0), 20 * \Delta t$

Game Control Paradigm: The speed at which the ball travels along the table is dictated by the average of the speed v of the paddle when the two game objects first collide (Eq. 4.4.7). The ball will reset every 20 seconds to prevent it getting stuck or travelling too slowly to score. When a ball leaves the table it collides with the troughs at the end of the table. The patient is rewarded with particle effects, auditory feedback and a game point for a completed trial. A new ball is then dropped from the ball dispenser in the center of the screen.

Eq. 4.4.7. $\quad v = \dfrac{\Delta h}{\Delta t} \qquad\qquad$ $1 < v$

Summary: This game is one of the simplest games we have created, but works on a fundamental functional movement. Overcoming muscle spasticity after injury is one of the more difficult problems early on in a patient's recovery. This game can be adjusted to require small changes in angle to move the paddle which can allow individuals with spastic flexion to still play the game, slowing working towards increasing their range of motion. Further, this game does not require any timing or sustained movements. This allows patients to take their time completing a trial or even rest part of the way through.

Figure 4.4.17. Ball Roll Game. In this game the patient moves their hand left and right at the wrist to control a panel (i) to push the yellow ball towards the trough (ii). When the ball reaches the trough a new ball drops from the dispenser (iii).

4.4.B.2 Turn the Dial

Functional Movements: Forearm pronation and supination.

Objective: Rotate the "player" dial to match the target dial while keeping the elbow stable. When matched, the patient must hold that rotation for a set amount of time.

Movement Control Paradigm: The angle of the player dial $\tilde{d}$ is controlled by the x and y components of a normal vector coming from the center of the patient's palm $\vec{N}$ adjusted for handedness (Eq. 4.4.8). The normal vector is orthogonal to an idealized "flat" palm, but functions as if the hand was flat even if a fist is made. Handedness is calculated by preset leap functionality that determines if the viewed hand shapes best represent a left or right hand. This is primarily determined by the indent of the palm and relative position of the thumb and pinky. The rotation of the dial is handled with a Lerp function to enhance the smoothing of the game (Eq.

4.4.9). The patient's relative elbow position $\dot{E}_p$ (calculated as the difference of x positions of the center of the palm and the elbow) is remapped (Eq. 4.4.2-3) to an elbow meter game object $\dot{e}$ on the screen.

Eq. 4.4.8. $\quad d = \tan^{-1}\left(\frac{\vec{N}.x}{\vec{N}.y}\right) * \frac{180}{\pi} \qquad\qquad d = d + 180 \ iff \ \vec{N}.x < 0$

$$d = d - 180 \ iff \ Handedness = Right$$

Eq. 4.4.9. $\quad \tilde{d} = \}\tilde{d} \rightarrow (0, d, 0), 15 * \Delta t$

Game Control Paradigm: Sample target angles are generated at the start of the game from a pool defined by the min and max angle game settings 'a_{min}' and 'a_{max}'. If the player dial angle is within a preset tolerance (10 degrees) and the elbow meter is within its maximum bounds $\pm$ 'e_{max}' from center ($\dot{E}_p = 0$), then a timer begins. A trial is complete if the timer value exceeds the hold time setting value 't'. If the player dial angle or elbow position leaves the tolerated range the timer is reset. The timer value is mapped to a UI element on the screen that fills as the patient meets the trial criteria.

Summary: *Turn the Dial* focuses on pronation and supination of the forearm by having the patient control the knob of a radio to match a target knob. Further, the patient must keep their elbow in line with their hand while matching the angle for the timer to increment. This is done to ensure that the patient is rotating their forearm and not compensating by rotating at the elbow. By directly adjusting a_{min} and a_{max} a therapist can hone in on a patient's problem areas. Finally, the hold time setting t can be reduced to a minimum to work on range of motion, or increased to higher levels to work on stretching and strength.
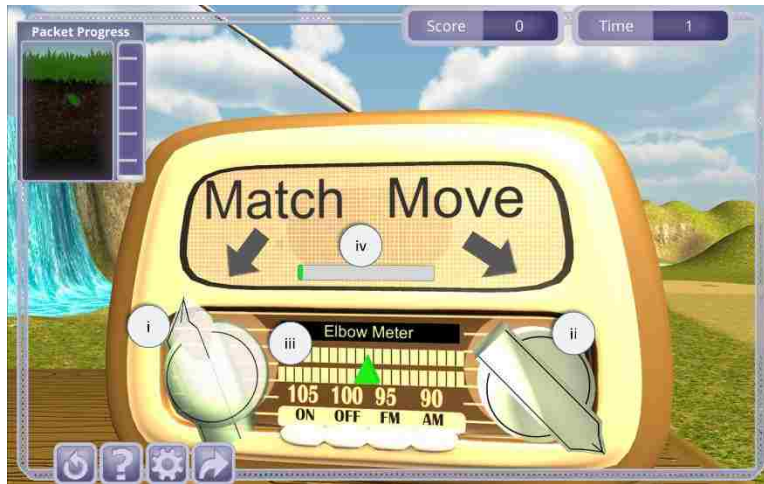
Figure 4.4.18: Turn the Dial Game. The patient controls the rotation of the right dial (ii) to match the target dial (i). The patient's relative elbow position is also tracked (iii) to ensure proper form during the movement. If the elbow and hand angle are in the proper position, a timer begins, filling a UI meter (iv).

4.4.B.3 Reach and Hold

Functional Movements: Arm range of motion and trunk stability.

Objective: Move the in-game hand towards the floating shapes and hold the position for a set amount of time.

Movement Control Paradigm: The position of a virtual game hand is directly remapped to the patient's palm position in all three dimensions based on preset game boundaries (Eq. 4.4.2-3). The virtual hand's x-y and x-z position is projected to two dimensional grids at the bottom and back of the game space to assist in visualizing the patient's current position.

Game Control Paradigm: The patient must move the virtual hand into a randomly generated target object within the three-dimensional game space. If the virtual hand's collider is in contact with these targets a timer will begin counting. While the timer is counting, the color of

the target will begin to change. A trial is complete when the timer meets or exceeds the hold time

setting *'t'*. The timer will reset if the virtual hand is no longer in contact with the target object.

The target's position is mapped to the same two dimensional grids as the virtual hand as an

additional feedback element for the patient.

Summary: *Reach and Hold* takes advantage of the Leap Motion sensor's ability to track

hand position in three dimensions. The patient can reach out in front of them within the Leap's

interaction space to touch virtual targets with their virtual hand. Using higher hold times a patient

can work their trunk stability and arm strength while in a static position. Moving between targets

also works on a patient's accuracy and reaction time to presented stimuli. Adjusting this game's

sensitivity setting requires the patient to produce smaller or larger real movements to reach the

targets. Gradually decreasing the sensitivity setting may lead to the patient increasing their range

of motion over time. Projecting the hand and target to a two-dimensional space allows the

patients to learn how to move a virtual hand in three dimensions more easily.
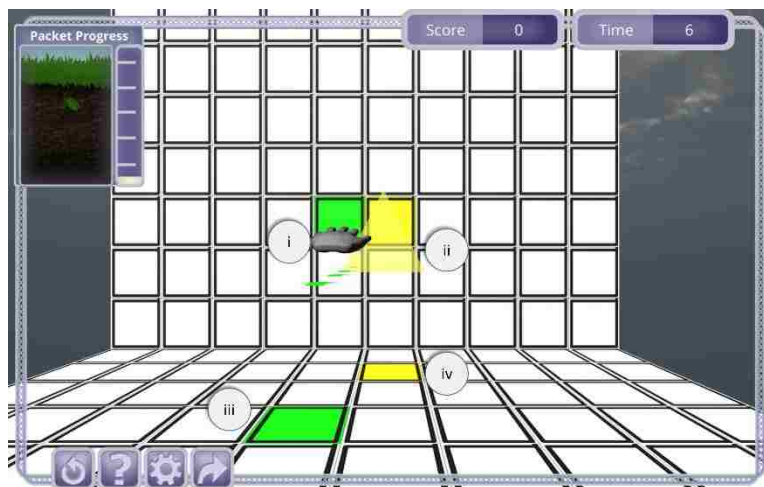
Figure 4.4.19. Reach and Dwell Game. In this game, the patient moves their virtual hand (i) in three

dimensions to reach towards target game objects (ii, yellow triangle). The virtual hand's position and

target position are projected onto two-dimensional grids (iii, green and iv, yellow respectively).

159

4.4.B.4 Reach and Grab

        Functional Movements: Arm range of motion and grip.

        Objective: Grab virtual fruit off a shelf and move it to the bowl on the counter.

        Movement Control Paradigm: The patient's virtual hand is controlled in the same way as in *Reach and Dwell*. In this module however, we have added the ability to open and close the virtual hand. The virtual hand can either be opened or closed depending on whether the patient's "grasp strength" is above (closed) or below (open) a preset threshold. "Grasp strength" is a value provided by the leap motion system that reflects how close all of the visible fingers are to the center of the patient's palm and ranges from 0 (flat palm) to 1 (fist).

        Game Control Paradigm: Six fruit are randomly generated and placed in six preset spots on the game shelf. When the player's virtual hand is near the fruit, the fruit will magnetize to the patient's hand. The patient can then make a fist to grab the fruit. While grabbed the fruit can be moved around the game space. The fruit will be dropped if the patient opens the virtual hand. Dropping fruit so that it collides with the bowl will result in a completed trial. Dropping fruit anywhere else will cause the fruit to snap back to its starting position. When all six fruit have been dropped into the bowl, six new fruit will be generated in the game shelf.

        Summary: *Reach and Grab* works on coordinated movements in a wide three dimensional space. Patients must reach out in front of them towards the shelved fruit. After magnetization, the patient must make a fist to grab the fruit and then move back towards the bowl. When they are over the bowl they can open their hand to release the fruit and drop it in the bowl to complete a trial. If a patient cannot open and close their hand, this game can be adjusted

to automatically grab the object after magnetization with a "Make Fist" setting. This, however, requires the patient move the fruit so it collides with the bowl directly to complete a trial. One of the strong benefits this game offers is the ability to have an easily repeatable grasp and move task that is not readily available in a true physical space. To emulate this game's setup, a therapist would need to replace the target objects on the shelf each time a patient completed a trial. In the home it would require the patient to reset the shelf each time, which may be impossible for the patient to complete.

Figure 4.4.20. Reach and Grab Game. In this game, the patient moves their virtual hand (i) in three dimensions to reach towards fruit on a shelf (ii). The patient can then grab the fruit and drop it into a bowl (iii).

4.4.B.5 Pizza

Functional Movements: Arm range of motion, hand grip, and pattern matching.

Objective: Grab the virtual pizza ingredients in the correct order and place them on the pizza.

Movement Control Paradigm: The control paradigm for this game is the same as in *Reach and Grab*. However, the game boundaries for remapping (Eq. 4.4.2) are restricted to the x-y plane so that patient movement is only required left-right and up-down.

Game Control Paradigm: The game mechanics are fairly similar to *Reach and Grab* where pizza ingredients are magnetized and need to be "grabbed" to be moved to the pizza crust. In this game, however, the pizza ingredients need to be added in a specific order (sauce -> cheese -> toppings). The correct object to grab is highlighted with a particle effect for feedback. Patients can grab the wrong objects, but if they try to drop them on the pizza crust the ingredient will snap back to its starting position. Adding all 5 ingredients to the pizza will cause the scene to reset.

Summary: The *Pizza* game is a functional re-skin of *Reach and Grab* with a slight pattern matching twist. Additionally, while *Reach and Grab* focuses on reaching forward and back with slight deviations to the left and right, this game focuses on left and right movements. A patient is rewarded for a completed trial when each ingredient is added to the pizza, but receives an additional visual and audio feedback reward when the pizza is completed. Settings for this game are the same as in *Reach and Grab*, allowing patient's with limited mobility to play with higher sensitivity settings.

Figure 4.4.21. Pizza Game. In this game, the patient moves their virtual hand (i) in primarily 2 dimensions to make a pizza. Patients must grab ingredients in the correct order (sauce (iii) -> cheese (iv) -> toppings (v)) and drop them on the pizza crust (ii).

4.4.B.6 Giant's Teeth

Functional Movements: Quick velocity changes (as in brushing your teeth)

Objective: Move the toothbrush quickly back and forth to remove the plaque from the giant's teeth.

Movement Control Paradigm: The patient's palm position $\dot{P}$ is remapped to a virtual toothbrush (Eq. 4.4.2). The leap boundaries are similar to those in the *Pizza* game, leading to movements primarily in the x-y plane.

Game Control Paradigm: Each auto-generated plaque object contains a collider mesh. When the running, rectified average of the toothbrush velocity $||v||_{n...}$ exceeds a set value 'v' the plaque touching the toothbrush will disappear (Eq. 4.4.10). The target velocity 'v' can be adjusted with a game setting. The running average value is only counted when the toothbrush

bristles are within the plaque's collider mesh, meaning the patient must make quick and small movements to remove the plaque. Running average is calculated over n = 10 samples or about 500ms, with variations occurring due to varying framerates. Specifying horizontal or vertical requirements in the settings menu allows for only the velocity components of the target direction to count. This is akin to substituting $\dot{P}_t$ for $P.\dot{x}_t$ or $P.\dot{y}_t$ in Eq. 4.4.10. After each of the eight plaques have been removed, they will all respawn in their original positions.

Eq. 4.4.10. $\quad |v||_{n\ldots} = \dfrac{\left(||v||_{n-1} + ||v||_{n-2} + \cdots + ||v||_1\right) + \frac{\|\dot{P}_t - P_{t-1}^{\cdot}\|}{\Delta t}}{n}$

Summary: *Giant's Teeth* has the most straightforward mapping to real world function. The acceleration of the hand as it changes from one direction to the opposite direction is necessary for a lot of movements, especially brushing teeth. By modifying the required velocity value 'v' a therapist can start their patient at easier difficulties, often allowing the patient to clear an entire row of plaque in one good movement. Increasing this value requires the patient to carefully control their movements while still moving quickly. Allowing the therapist to specify between horizontal or vertical movement further allows for patients to work on their specific problem areas.
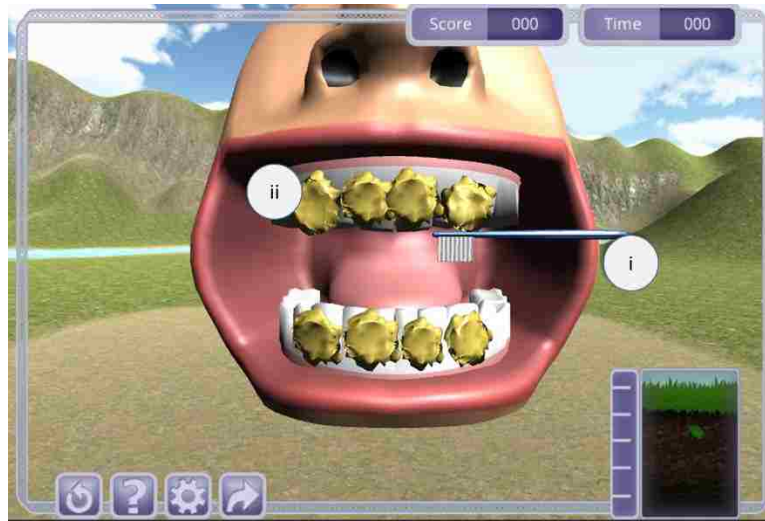
Figure 4.4.22. Giant's Teeth Game. In this game, the patient moves a virtual toothbrush (i) to remove plaque (ii) from the giant's teeth. Patients maintain contact with the bristles of the brush and the plaque while rapidly moving left and right or up and down to remove it.

4.4.B.7 Whack-A-Mole

Functional Movements: Finger Flexion and Extension.

Objective: Move the hammers up and down to hit the moles as they pop up.

Movement Control Paradigm: The relative position of each of the patient's finger tips controls the angle $'\tilde{h}'$ of five corresponding hammers pivoted around the base of the hammer's handle '$\dot{h}$'. Both the angle and height of the hammer are adjusted to induce a smooth controlled motion that does not accidentally hit a mole when it is at rest. First the fingertip position '$\dot{F}$' is remapped to game space $\dot{f}$ (Eq. 4.4.2). The y position '$\dot{f}.y$' of each finger is used to calculate an average position '$\dot{f}.y_{avg}$' (Eq. 4.4.11). We calculate hammer position using this average value along with an additional compensation that adjusts for pronation and supination of the wrist using a remapped palm position '$\dot{p}$' and the palm normal '$\vec{N}$' (Eq. 4.4.12). Hammer rotation is

directly mapped to rotation along the x axis using the height value and a few preset values (Eq. 4.4.13). Further, this mapping is directly adjustable by a sensitivity setting *'s'*.

Eq. 4.4.11. $\quad \dot{f}.y_{avg} = \dot{f}.y_{thumb} + \dot{f}.y_{index} + \dot{f}.y_{middle} + \dot{f}.y_{ring} + \dot{f}.y_{pinky}$

Eq. 4.4.12. $\quad \dot{h}.y = ((\dot{f}.y_{avg} - \dot{f}.y) - (\dot{p}.x - \dot{f}.x) * \vec{N}.x) * s$

Eq. 4.4.13. $\quad \tilde{h} = (90 - \frac{60 * \dot{h}.y}{4}, -25,0)$

Game Control Paradigm: Each hammer has a collision mesh on the hammer's head and the mole is surrounded by one collision mesh. Mole positions are randomly selected from 5 pre-determined positions immediately beneath the hammers. A trial is completed when a collision between the hammer and mole occurs and the mole respawns in a new location. The mole's position cannot be the same two trials in a row. A further option for this game allows for moles to appear above the hammers as well. In this case, the mole's position is randomly selected between all 10 positions but cannot occur at the same height twice. Additional settings prevent the mole from showing up beneath specified fingers. These settings also directly affect Eq. 4.4.11 where that removed position is removed from the calculation of the average. If only one finger is used the average position defaults to the remapped palm position. A one finger setting, however, is only useful if the mole appears above the hammers as well.

Summary: *Whack-A-Mole* works on the flexibility of individual fingers. Each finger must move independently to reach the mole as any concurrent movement will adjust the average value (Eq. 4.4.11) and hinder the movement of the hammer. The sensitivity on this game works in a similar manner as the others, but has a much larger impact on the game's requirements. Healthy individuals may even have a hard time reaching pinky targets on the lowest sensitivity

without practice since the ring finger tends to move alongside the pinky finger. This game does require patients to be able to spread their fingers slightly as the Leap camera has difficulty identifying fingers if they are too close together.

Figure 4.4.23. Whack-A-Mole Game. In this game patients must move individual fingers to control the position of hammers (i). The patient must move individual hammers to hit a randomly appearing mole (ii).

4.4.B.8 Two Hand Shape Match

Functional Movements: Bilateral hand movement, forearm pronation and supination, and grip.

Objective: Move both hands to match both the height and shape of target hands. When matched, the patient must hold the position for a set amount of time.

Movement Control Paradigm: Both of the patient's hands are remapped to game space (Eq. 4.4.2) in the y direction only. Each hand can be in three distinct states: 'chop', 'slap', and

'fist' determined by the hand's grab strength 'G' and palm normal '$\vec{N}$' (Figure 4.4.24). A chop shows the hand flat in the y-z plane, a slap shows it flat in the x-z plane, and a fist shows the hand closed.
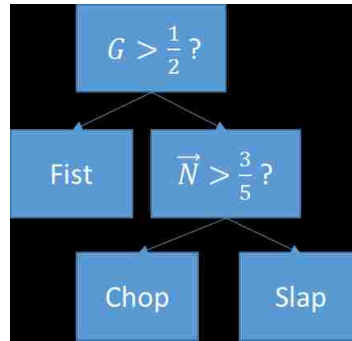
Figure 4.4.24. State diagram for Two Hand Shape Match Game. To determine which of the three states the hand could be in we first look at the grab strength and then the palm normal.

Game Control Paradigm: The patient must move their hands to match both the shape and height of two mirrored target hands. The mirrored hands can either be high or low and can be in any of the three hand states ('chop', 'slap', or 'fist'). A 'Same Shape' toggle setting forces both hands to be the same shape. A 'Match Height' toggle setting places both hands in the top position and removes the matched height requirement. When the states are matched, a timer will start that fills a center feedback meter. A trial is completed when the timer exceeds the hold time setting value. If the state becomes unmatched either from height or shape, the timer will reset. Additional height feedback elements are shown next to the timer meter. The two blocks nearest the meter show the target hand height, while the outer blocks show the patient's current hand height.

Summary: *Two Hand Shape Match* requires a considerable amount of coordination with both hands. While the states can be matched in any order (left or right first, height or shape first),

a high score comes from moving all pieces together in a coordinated manner. Moving between slap and shop states works on pronation and supination, while making a fist works on grip. Having the target shapes be different requires additional cognitive skills to view the mirrored hands and translate that to the correct patient hand movements. Modifying the sensitivity (or even turning off "Match Height") can allow the patient to focus more on shape matching if they have difficulty in keeping their arms at a certain height. Alternatively, a low hold time can allow a patient to make quick movements up and down to reach higher scores.
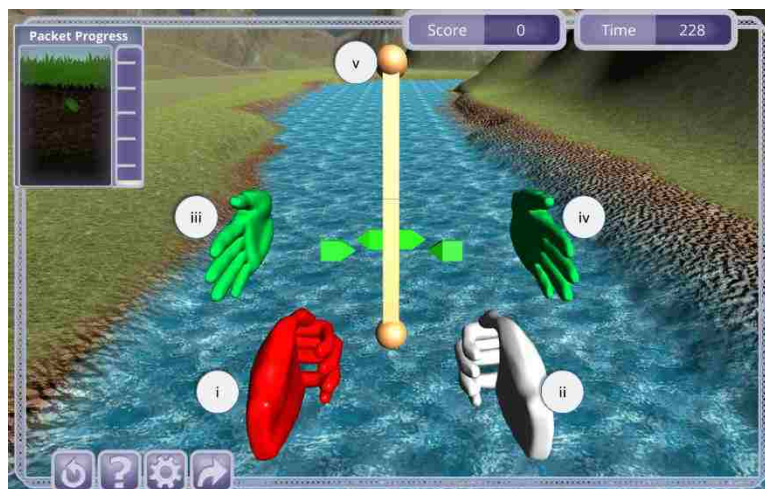
Figure 4.4.25. Two Hand Shape Match Game. In this game patients must move both left (i) and right (ii) virtual hands to match both the height and shape of corresponding target hands (iii and iv). A hold time meter with attached height indicators (v) provide additional visual feedback for the task.

4.4.B.9 Finger Position Match

Functional Movements: Finger adduction and abduction, pattern matching

Objective: Match the finger orientation of the target hand

169

Movement Control Paradigm: The tips of each of the patient's fingers and their relative locations to each other create a set of four touch states. These states estimate whether the fingers are or are not touching each other for the following combinations 1) thumb-index, 2) index-middle, 3) middle-ring, and 4) ring-pinky. A state is true if the tip positions, projected on a line perpendicular to the hand direction, are within a set threshold of each other. This threshold can be modified by a sensitivity value, but is primarily set based on the Leap's ability to detect the width of a finger. A virtual hand on the screen represents the patient's current state by modifying the relative angles of the fingers at the metacarpophalangeal joints. There are not intermediate states within this model.

Game Control Paradigm: A randomized set of touch states are generated for the target hand. The target hand is then positioned in a similar manner as the patient's virtual hand. A trial is complete if this set of states matches the patient's states after which a new set of states is generated.

Summary: *Finger Position Match* utilizes a simple matching paradigm to work on a patient's ability to spread and compress their fingers. This game has not yet had a theme treatment like the other games, but has already shown popularity amongst some therapists. It is important to note that some patients cannot produce all touch states. Using the right hand as an example, (false, true, false true) represents a hand position that is impossible for some individuals. In the current version, such a state can be skipped, but future work may exclude these entirely. Further, this game requires the patient to make a relatively flat hand, which may be difficult for some patients who otherwise need this type of exercise.
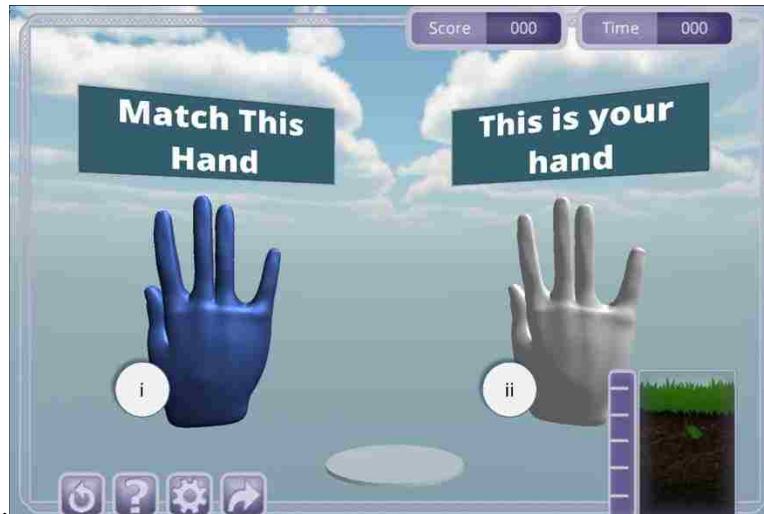
Figure 4.4.26. Finger Position Match Game. In this game, patients must adjust the positions of their fingers to control a virtual hand (ii) and match a target hand (i). Finger positions are represented in a set of finger touch states that indicate whether or not two fingers are touching each other. The above state is 1) true, 2) false, 3) true and4) false.

4.4.B.10   Pillbox

Functional Movements: Pinch grip, pattern matching

Objective: Move pills from the bottles to the correct slot in the pillbox.

Movement Control Paradigm: Patients control a set of chopstick-like objects that can either be opened or closed. Similar to the virtual hands described in previous games, these chopsticks are remapped to the patient's palm position (Eq. 4.4.2). Opening and closing the chopsticks is controlled by another native Leap parameter 'pinch strength'. Pinch strength is related to how much of a pinch movement the patient is making controlled by how close the thumb tip position is to any other finger. The thumb touching another fingertip produces a value of 1, whereas the thumb being as far away as possible produces a value of 0. If the patient's

pinch strength is above a preset threshold the chopsticks will be closed, otherwise they will be open.

Game Control Paradigm: Patients must first open one of the lids on the pillbox by tapping the bottom of their closed chopsticks (the location of a collision mesh) to the target lid. The lid will stay open for a set amount of time depending on a customizable game setting. Patients must then move the chopsticks so that they collide with one of three pill bottles. Each pill bottle has a label stating which corresponding pillbox slots it can fit into. When the chopsticks collide with the pill bottle they must be in an open state. The patient can then close the chopsticks to grab a pill from that bottle. When this action occurs the game creates a pill of the correct color and places it between the end tips of the chopsticks. Finally the patient must move the chopsticks so that the pill is above the target slot in the pillbox and then open the chopsticks to drop the pill. A trial is complete when the pill collision mesh collides with the appropriate pillbox slot. If the pill is dropped in an incorrect slot it will disappear and no points will be given.

Summary: *Pill Box* is by far the most challenging game we have developed. Patients have limited time after opening a lid to grab the correct pill from the bottle and return it to the slot. This game has a direct translation to a patient's activities of daily living, but is limited in its current scope. As an already challenging game, we have limited the pill bottles to static labels (the left bottle is always 'M, T, W' for example). A true ADL task would require us to modify the labels to show that in addition to being able to make the movements, the patient knows how to sort pills from bottles into the correct day in the pill box. Additionally, the task would work best with a limited number of pills per day. In the current iteration, a patient can continuously move pills from the Monday bottle to the Monday slot and still receive points. This is fine for

task simplicity but would need to change to make the task more functionally relevant. The chopstick visualization was chosen to more closely resemble a pinch action, but future iterations may attempt to show a real hand making a pinch movement. Finally, the game is limited in its representation of a pinch. Pinch strength represents the proximity of the thumb to any finger and thus can reach a 'closed' value even in non-pinch hand postures (such as a fist).
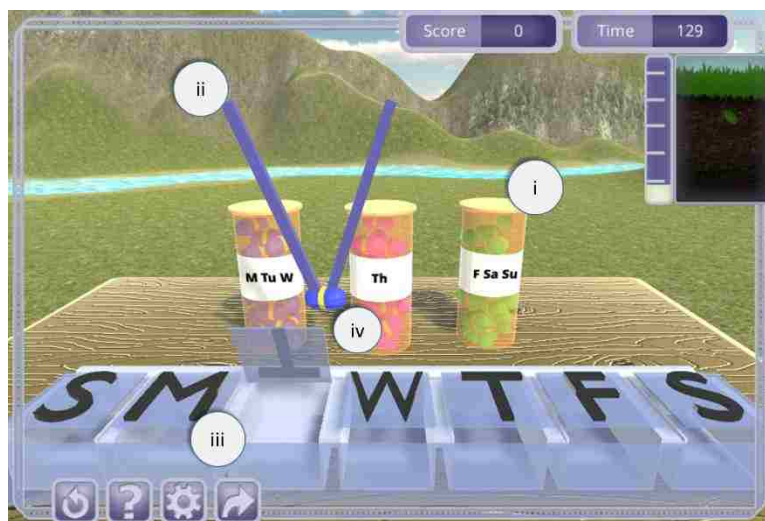
Figure 4.4.27. Pillbox Game. In this game patients must move the chopsticks (ii) to various pill bottles (i) to grab pills (iv). Patients must then drop the pills in the corresponding pill box slots (iii) to complete the trial.

4.4.B.11   Other Games Not Described Here

There are a few games that are not described in the sections above. These are primarily reskins of existing games, such as a boat game that uses the same control paradigm as *Ball Roll* to steer a boat down a river, or games that have not yet been tested thoroughly with our pilot sites, such as a two hand tai-chi type game. Other games are ones we have developed but

scrapped due to poor performance in the field. One such example is a pinching game that mapped the thumb and one other finger to balls in space. Spikes would fall from the sky that the patient could dodge by navigating through strategically placed holes. While the game idea itself worked well, we found that too many patients had a difficult time orienting their hands such that both fingers would be detected properly. This led to frustration from the patient and the therapist and has led us to remove the game from the current version until further evaluation.

We also have developed a few sandbox modules that have no direct goals or trial structure, but allow patients to either destroy block-based cities or move items around freely in a kitchen. These modules are interesting as a concept since they allow the therapist to direct patient action by saying things like "put the peanut butter next to the banana." This, however, makes providing analytics and performance measurements very difficult since we have no time markers of when target actions may have occurred.

4.4.C *Assessments*

The vHAB assessment framework was created out of the vAssess concept game described in Section 4.2. We wanted to be able to streamline our measurement and feedback capabilities from the games so therapists could obtain quick and accurate representations of a patient's hand and arm function. This streamlining involved removing some of the thematic elements of the games while also sacrificing the repetition inherent in gameplay. Assessments are meant to be performed in the clinic with both the therapist and patient present, and may function best as intermediate check-ins along a patient's progression. We intend the assessment modules to function as part of the entire vHAB system such that these assessments are done periodically, but the primary therapy is performed within the games. The assessment user

interface framework described in section 4.4.A.6, allows therapists to customize an assessment with various modules that specialize in one particular movement or measurement. This list is continuously growing based on recommendations from therapists and clinicians. In the sections below we describe the modules that are currently available for therapists to use.

Assessment Environment

All Modules have the same game scene and environment (Figure 4.4.28). This environment has a few distinct features. First, text prompts appear at the top of the screen describing the current module and the required actions. These prompts function similarly to the tutorial prompts discussed previously, advancing either by time or met physical requirement from the patient. The environment also contains various feedback elements such as a timer panel and a mock virtual Leap Motion controller. Most modules utilize a virtual hand that functions the same way as the hand in *Reach and Dwell*. The patient's real hand is remapped to the virtual hand (Eq. 4.4.2) and the game hand is projected onto front and back grids to assist the patient in locating their hand. In the assessment environment, however, we also trail this projection for a few frames so that the patient has an understanding of their hand's position history. A patient can also make a fist with their hand as in *Reach and Grab* by passing a similar grab strength threshold. In some modules the hand position and ability to make a fist is locked for simplicity.
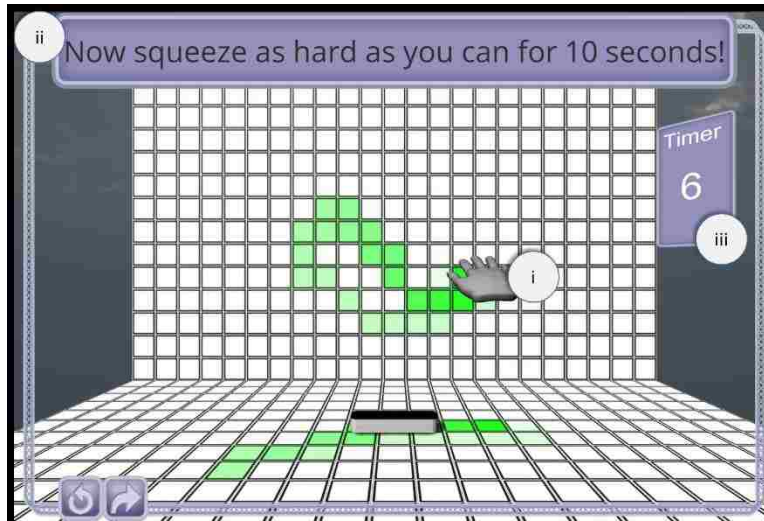
Figure 4.4.28. Assessment environment. The assessment environment is static across most of the assessment modules. In all modules, prompts are displayed at the top of the screen (i) to lead patients through the exercise. Modules can use a virtual hand (ii) to represent the patient's real hand position and a timer (iii) to assist the patient in the task.

4.4.C.1 Introductory Module

This module introduces a patient to the assessment environment. As previously discussed, many patients have a difficult time interacting with our system at first since they are not physically interacting with anything. This module explicitly addresses this by introducing the system in a piece-wise method. First the prompts describe the purpose of the system then ask the patient to place their hand 6" above the sensor. When a hand is detected, the prompts ask the patient to move their hand around, noticing the projections on the front and back grids. It then introduces the concept of a "home box", a virtual box located at sensor zero, and requests they return to the home box. This module is only intended to be used by the patient one time as an introduction, but may be returned to in special cases such as patients with dementia or after a long break from using the system.

4.4.C.2 Hand OpenCloseOpen

This module asks the patient to place their hand flat above the sensor. Next they are asked to make a fist and squeeze tightly for 10 seconds. Finally they are asked to open their hand again. This module is very simple, but provides great insight into the patient's ability. By having a patient squeeze their fist for 10 seconds, we receive a long profile of their muscle activity and how it changes over time. A single instance can measure a patient's maximum voluntary contraction (MVC), but performing this activity before and after a game or therapy session can provide insights into endurance and strength.

4.4.C.3 Arm Range of Motion

We provide three separate modules for measuring arm range of motion, a 30 second and 60 second bubble popping module and a 60 second painting module. For the bubble popping modules we present the patient with a set of procedurally generated bubbles, starting in the center around the home box. Each time the patient's virtual hand collides with a bubble, we show a popping animation, and generate a new bubble slightly further from the home box up to a maximum volume. As the patient continues to hit more bubbles the required range of motion steadily increases. By starting toward the middle we ensure that all patients can at least hit some of the bubbles. Further each bubble is a discrete object and each popped bubble can be viewed as the completion of a single trial for analytics purposes. From this we can attempt to measure things like path deviations and reaction times alongside the volumetric ranges of motion. The 30 second bubble popping module is for individuals experienced in the 60 second bubble popping module. The main challenge with the bubble popping modules is the visibility of the bubble's depth in the three dimensional environment. Experienced users are able to map where their hand

177

is relative to a given bubble, but a naïve patient will have greater difficulty in performing the task.

The painting range of motion module takes advantage of the projection capabilities of the assessment environment. Instead of trailing a history of projections we simple keep the projection grid filled in wherever the patient's virtual hand has been. The patient is tasked with filling in as many grid panels as they can within the 60 second time period. This version of the range of motion measurement does not have the same discrete task goals, but does not suffer from the same depth visualization problems as the bubble popping games.

4.4.C.4 Wrist Range of Motion

In this module we present patients with both their virtual hand and a target hand. The target hand will proceed through a series of orientations designed to test wrist range of motion, such as flexion/extension, pronation/supination, and radial/ulnar deviation. The patient is tasked with matching these angles and then holding the position for 2 seconds. As an assessment we cannot make any assumptions of the patient's true abilities. To account for this we assume a patient is "matching" the target angle if their angle along the current axis of rotation becomes closer to the target angle by a set threshold. For example, in wrist pronation and supination we first display a hand that is flat in the x-z plane and then rotate it around its center along the z-axis. We rotate the target hand 180 degrees as an ideal maximum that the patient could achieve. To advance the prompt, however, a patient only needs to make a 10 degree delta along the z axis. This method can sometimes cause patients to not reach their full range of motion potential, especially if it is their first time through the module. We have noticed, however, that very few

patients watch their virtual hand. Instead most attention is focused on the target hand and they work to reach the target angles even if they cannot actually match the true angle.

4.4.C.5 Rest Modules

These modules prompts the user to rest for 5, 10, or 20 seconds depending on the module selected. At the start of the associated rest prompt we start a timer for the corresponding time. At the end of the time we ask the patient to return to the home box. The module is finished when they return to the box, indicating they are ready to begin the next module.

4.4.C.6 Questionnaires

Questionnaires were some of the most requested features within the vHAB system. While we provide new and improved methodologies for delivering therapy, there are still many supplementary methods that can be used in tandem with our system. Further, questionnaires can be custom designed for facility studies where they may want to correlate our measurements with patient responses or other, more manual, measurements. Providing questionnaires ensures all data are placed in the same location for easy analysis. The answers to the questions can also be viewed within the data viewer for easy comparisons over time. A questionnaire consists of a set of multiple choice, number line, or input field questions. Currently we have prototyped questionnaires for pain scales, angle verification inputs, and multiple patient hand, wrist, and arm measurement forms that exist as the current standard of care such as QuickDASH [42] and the Patient Rated Wrist Evaluation [43]. The prompt format for a question is much different from a normal module since we do not require a virtual hand or other objects. For the interface we present a touch-based user interface that can be quickly navigated by the therapist or the patient (Figure 4.4.29).

179
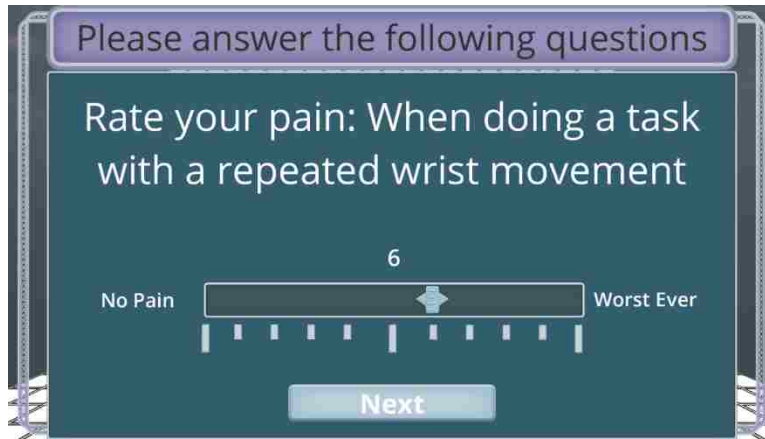
Figure 4.4.29. Sample questionnaire module interface. This sample asks the patient a single question about their pain experienced during a specific situation. This question exists within a set of questions from the Patient Rated Wrist Evaluation questionnaire. To advance, the therapist or patient must click or touch the scale to select their pain and then press next.

4.4.C.7 Combination of Modules

The true power in the assessment framework is the ability to combine a custom set of modules. For example, a sample set may consist of C1. Introduction -> C6.Pain Scale -> C2. Hand Open Close Open -> C3. Bubble Popper 60 -> C2. Hand Open Close Open -> C6.Pain Scale. In this manner fatigue can be viewed as a measure of the difference in MVC between the first and second C2 modules since the patient was tasked with performing a potentially tiring task in between. Impact of therapy on pain can be directly measured by comparing the responses to C6. Comparing this consistent module set over time further allows therapists to make informed decisions about their patient's care.

4.4.D   *Data Management*

   With the vHAB system we collect hundreds of streams of physiological and gameplay data for each game and assessment. We also track usage data for the games, assessments, and even the menus. Patient specific data are all stored on the local system under the patient's pID while usage data tracked across multiple patients and therapists is stored in a separate global directory. These data need to be available for both the analysis framework (Section 4.5) and for general validation and testing purposes. In the sections below we describe how the data are collected and stored within the vHAB system.

4.4.D.1 Directory Structure

   At the top of the directory structure is a folder named "Data" sitting within the root directory of the vHAB installation. This Data folder holds all of the pID folders alongside basic usage data files and some other operating files, such as an *unlockedLevelList.txt* that hosts an encrypted list of what levels are available for the therapist to use out of the 10+ total. In the home version, only one pID folder exists for that specific patient. Each pID folder is structured with a series of tiered folders. The top level of the pID folder holds a set of date folders containing all of the game and assessment data for that date, alongside some patient-specific information such as the *patientLevelList.txt* that holds an encrypted list of the games patients see on their game select screen. Within each date folder is set of session folders. A session is any contiguous set of games or assessments that occur without the therapist logging the patient out of the system. Multiple sessions most often occur when a patient is staying in a facility and does morning and afternoon therapy. Within each session folder is a set of folders for each game they played during that session. If a game was played more than once the folder name has a counter added that

increments by 1. Each game folder contains all of the raw gameplay and physiological data collected during the game, alongside a results folder produced by the analytics in Section 4.5. A full path to the results folder and the main scalar values may look like:

"./Data/pID/05302016/Session2/BallRoll3/Results/scalar_stats.csv".

Handles to all of the specific portions of the directory are handled with a carefully kept script with public functions for all the other data scripts. This ensures that all of the data is put in a consistent location for a session despite multiple scripts handling the various data saving functions. This overall directory structure allows for easy offline navigation during development but does present data integrity risks. If a therapist navigates to a random patient's folder they could modify specific information about a game or a session, though we expect this to be unlikely to happen in practice.

4.4.D.2 Raw Data

The vHAB system records four main streams of raw data during each game and assessment module: leap data, muscle activity data, settings data and metric data. Each of these streams has its own controller script that saves new data each frame. As mentioned previously, the Leap device updates at up to 120 Hz, and the Myo can record at up to 50 Hz. From these speeds, however, we only sample at the frame rate of the games (about 50Hz). Settings data are updated whenever the settings are changed within the game, while metric data are updated within every each frame. This frame limited approach can be modified to save missed data if it is available, but thus far has not been necessary for game control or analytics. All raw data are saved in separate comma separated value (CSV) format with headers for each data column. Each

row of data is time-stamped based on the elapsed game time and the system time. These values are useful in correlating data across data streams.

For leap data we save multiple positional points on the arm, hand and fingers, as well as hand and arm directions, grip and pinch strength, and confidence values. Most games only save one hand's worth of data for the leap (often the rightmost hand in the scene), however other games, like *Two Hand Shape Match*, save data for both hands. For muscle activity data, we save all eight streams of bipolar muscle activity data in their raw form. Settings data contain the current settings values at the start of a game, and all settings modifications. Metric data contain information about the game or assessment that is being played, such as whether the hand is holding a virtual object, when the patient completes a trial, or where a target object has been placed. Each game has a unique list of metric values that are saved.

All of the raw data are processed with a set of algorithms described in Section 4.5. These algorithms produce a set of analytics that are stored within the game directory that the raw data came from. This structure exists as a separate key-value pair file that can be quickly read into the data viewer screens.

4.4.D.3 Usage Data and Long-term Data

vHAB also saves summary data from each time a module is used. Usage data consist of the name of the module or menu, the time of day, the date, and how long the module or menu was opened. These usage data allow us as developers to understand how the systems are used to enhance therapist workflow. Further, these data can be used to identify popular games or assessments for future expansion and targeted updates. Long-term data are patient specific and contains a list of each game or assessment alongside the achieved score and time played. The

long-term data are used for quickly graphing simple analytics and metrics in the data viewer

screens and for determining if the patient has played the game before in the tutorials systems.

4.5 SYSTEM DESCRIPTION: ANALYTICS

4.5.A *Foreword: A Note on Contribution*

The following section contains work heavily attributed to Dimitrios C. Gklezakos. Conception, design, and testing were performed by Libey, Mogen and Gklezakos. Coding and implementation were primarily performed by Gklezakos. As such, in the sections below, we will focus primarily on summarizing the values and their use in relation to the vHAB system as a whole, rather than on the equations and accuracy of the measurements.

4.5.B *Introduction*

Integrated and automated analytics are a key component of the vHAB system. Engaging patients with fun games and a score wrapper is only half of the solution to solving low therapy adherence at home. With detailed analytics, patients can see small increases in function to stay motivated in ways they ca not do with traditional home therapy. For example, a patient may not be able to see a 5 degree increase in wrist range of motion from day to day, but with the vHAB system they can. Further, therapists can use analytics to better treat their patients. Monitoring multiple analytics over the course of a patient's treatment, allows therapists to identify weak spots and target their therapy to accommodate that patient.

The vHAB analytics platform is created primarily in Python. Each system contains a local installation of the Python framework alongside all of the required packages. At the end of each game or assessment module, raw data are saved to the local system in the specified directory (Section 4.4.D), and then a startup Python script is called to begin the analytics process.

This startup script is provided with the most recent module's name and the location of the raw data, which it uses to call the appropriate analytics scripts for that module.

The current set of analytics contains a variety of measurements rooted within traditional therapy paradigms, such as range of motion and reaction times, but also contains some measurements that were previously impossible to record in real time in the therapy facility. These new values present a challenge for both us as developers and for the therapy field in validation and defining appropriate use. Along these lines, we primarily present metrics therapists are familiar with as we continue to validate and collect examples of these new measures.

Throughout the vHAB system there are a few common measurements that are recorded. These values have similar names, but are not necessarily comparable across modules. In the sections below we will describe these analytics and provide examples for some of the games and assessments. A full list of the analytics calculated can be found in table 4.5.1.

4.5.C *Gameplay Analytics*

Game play statistics are the most straightforward of the analytics, but potentially the least useful on their own. To create the gameplay statistics we utilize the raw metric data streams that consist of timestamps for when a trial is completed and additional trial metadata such as target location. From these data, we create "Score", "Time Played", and "Reaction Time" statistics that represent how well a patient did in the game. Score is simply the number of completed trials in a session. Time Played is the length of that session. Reaction time is generally target based and is the mean and standard deviation of the time from the start of a trial to the trial's completion. For

example, in *Whack-A-Mole*, a ring finger reaction time of 1.5 seconds means that on average, the patient took 1.5 seconds to hit the 4th position mole with their ring finger, after it appeared.

In addition to these basic analytics for each game, some games provide the opportunity to extract specific meta-analytics. In *Two Hand Shape Match*, for example, we can compare trial completion steps between hands. If a patient has a slower reaction time in raising their left hand than the right we can present that as a normalized ratio, with 1 representing both hands moving together, and zero meaning that one hand is in position before the next hand moves. This coordination value does not have an analog in traditional therapy practice but may be very useful for patients with hemispatial neglect. In other games, such as the *Pizza* game or Pillbox game, we can derive a basic cognitive metric from incorrect trials. If the patient drops a Monday pill in the Tuesday pill box slot it will be measured as an incorrect trial to be compared against the total score the patient achieves. This value may be difficult to interpret since a failure could be caused by a cognitive challenge or by a physical inability to reach the correct target. Thus this value is best utilized alongside other physical analytics such as range of motion or tremor. Finally, we can measure analogs of endurance by examining reaction time changes over a given session. Decreasing reaction times may be a sign of fatigue and we would expect this endurance measure to increase over time. Deriving the scale of the endurance value requires comparison against a norm value. Our current model is being built on the healthy patient data collection described in Section 4.7.

Comparing gameplay metrics becomes very difficult as settings within the game change. A score of 20 with a low sensitivity represents a greater level of performance than a score of 20 with high sensitivity. Further, these metrics cannot be compared across games as the trial structures are very different. In the current version we do not make concessions on presenting

187

these data points, however a future iteration may see setting adjusted scores that use sensitivity

values as multipliers for a base scoring system. Overall, the score and time played analytics are

meant to be basic summary statistics to enhance engagement, not to drive clinical care. Reaction

time values are similarly affected by game settings, but provide interesting within-game session

comparisons. Varied reaction times between fingers in *Whack-A-Mole* may indicate poor

performance with specific fingers. This variance should decrease over time as patients recover.
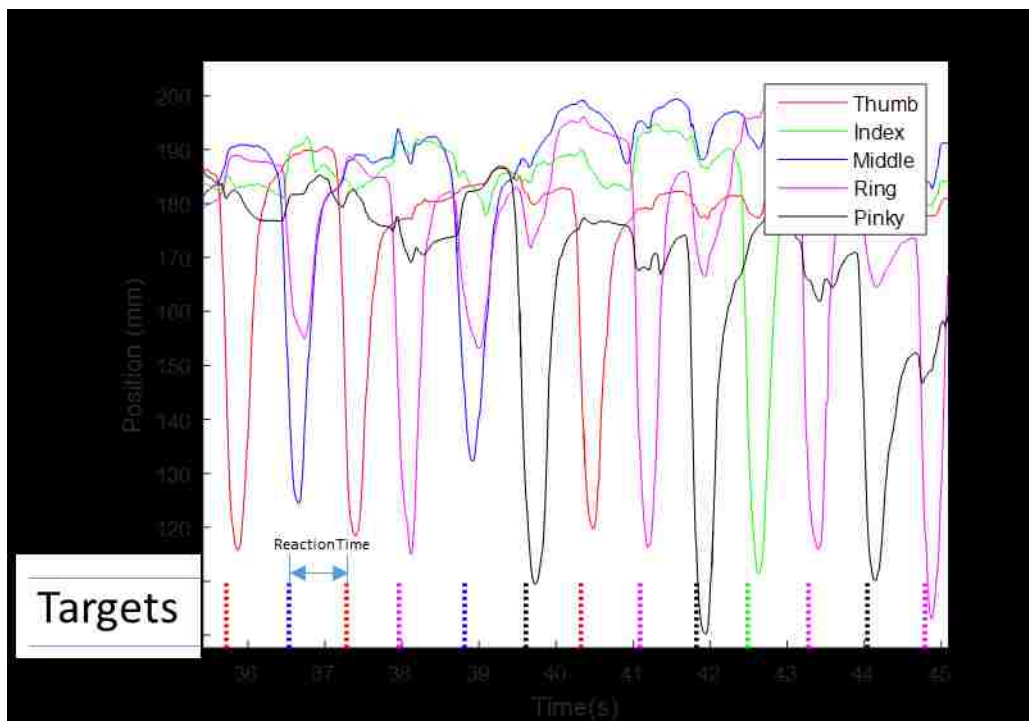
Figure 4.5.1. Gameplay analytics example for Whack-A-Mole Game. Generated targets are shown at

the bottom of the plot (dashed lines) alongside the vertical component of the fingertip position in this

idealized example. Fingertip height is not directly used to hit the targets, rather a relative position

calculation is performed in the game time (Section 4.4.B.5). In this example, extra care to keep the palm

flat and finger movements isolated to illustrate the game data components of the data analytics. Reaction

time is calculated as the time between trials, averaged over the course of the game on a per-finger basis. In the above example, the reaction time of approximately 0.8s would be attributed to the thumb.

4.5.D *Range of Motion*

Range of motion analytics describe an approximate distance or rotation around a given degree of freedom. Increases in range of motion generally track over time with a patient's recovery, but are not often measured in the clinic due to the complexity of taking measurements. In a traditional therapy setting range of motion is measured with goniometers. While digital, strain-gauge goniometers can be found on most medical product websites, the most used type consists of a set of plastic protractors. These goniometers present several challenges to use: they only have a specificity of +- 5 degrees, the measurements can vary drastically if the angle is being measured over bony surfaces, and they require that the patient maintain a static position during the measurement.

Our methods of calculating range of motion mimic that of the goniometer. We specifically look at three distinct points on the hand, arm or finger and calculate the angles along the specific axis. For example, in *Ball Roll*, we look at the palm, wrist, and elbow positions to measure angle deviations along the y axis. Some instances of our range of motion measures utilize relative distance measures. This is common amongst the finger motion games where specifying which degree of freedom along the finger to measure is related to the game play. For example, in *Whack-A-Mole* we use the relative position of the finger tips to the center of the palm and other fingers to control the hammer. For this range of motion measure we present the range of vertical distance traveled by each finger. Inverse kinematic models can be applied to

189

these values to estimate the actual angles made by the finger, but these values are not likely accurate in patients recovering from injury. The final type of range of motion we measure is a volumetric arm range of motion. This value varies from 0 to 1 and represents the total three dimensional space the patient moved their hand in out of the total possible space defined by the game and leap boundaries. The validity of this value is determined by the module's ability to ask the patient to perform all possible movements in the three dimensional space, and therefore takes a while to perform. However, no such measurement currently exists and we believe it could be a very valuable piece of information in determining a patient's ability to reach objects in front of them.

For all range of motion measures we present the mean value and standard deviation of the measurements across all trials in that module. In some user testing and pilot studies, however, these values are not easily translated into the same space as traditional goniometer measures. As mentioned previously, goniometers require that the patient hold a static position during the measurement. Our measurements are taken over more fluid motions and may end up being slightly higher than a held position. Further, a lot of the value of our measurements is in the repetition and continuous nature of the values. A full representation of a range of motion and how it is calculated is shown in the full set of data (Figure 4.5.2). This presentation does not fit into a therapists workflow and summary statistics are difficult to create that represent all of the information in just 20 seconds of data. Further, the standard deviation value or even a decay coefficient value better represent performance over time, but education and detailed documentation are required to provide therapists with the knowledge to interpret these values.
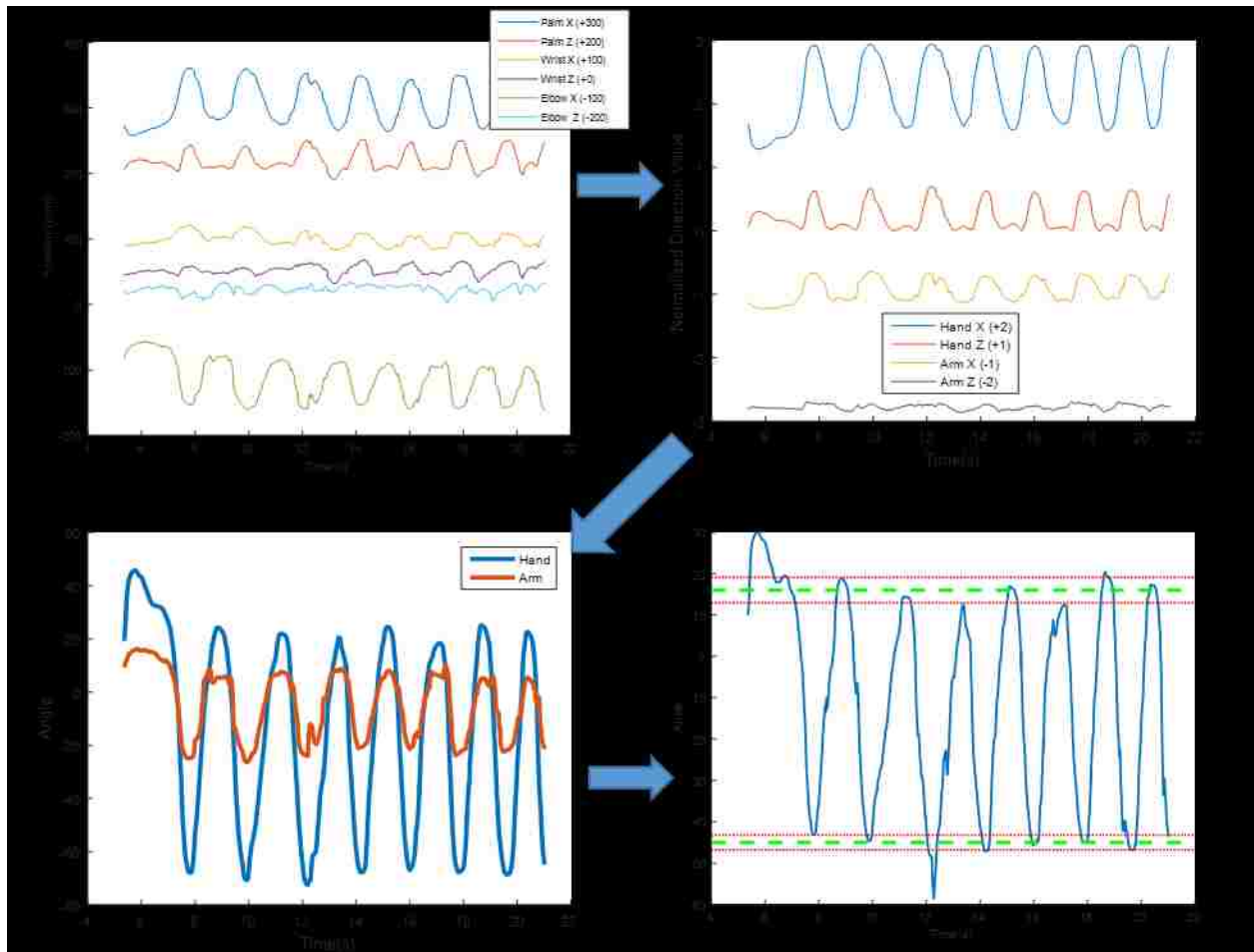
Figure 4.5.2. Sample range of motion plots for the Ball Roll Game. We start with the position of the palm, wrist, and elbow of the patient (Top Left). We use these data to calculate the normalized direction vectors for the hand and arm (Top Right) and then the angles of the hand and arm (Bottom Left). Subtracting these two angles produces our final angle value. Providing a relative value allows us to account for lateral movement of the arm that may occur during gameplay. Minimum (-45+-1.8) and maximum (16+-3.1) angles are displayed in the dashed red and green lines. It is important to note that this may not represent the true maximum or minimum the patient is capable of, but is instead the patient's ability to perform the actions we required of them.

4.5.E   *Muscle Complexity*

The muscle activity underlying hand and arm movements can showcase critical aspects of recovery from a variety of conditions. In the case of stroke there are typically several different presentations on the road to recovery including: flaccid paralysis (almost no EMG activity), increasing muscle tone, and spasticity [44]. There are a variety of techniques for processing surface EMG data to extract relevant clinical information from the complex signals. To create the most clinically-digestible numerical value we adapt the Non-Negative Matrix Factorization approach introduced by Seung and Lee in 1999 [45].

The NNMF technique attempts to build a representation of a complex data stream using a linear set of small set of basis vectors. These synergies represent common activation patterns in the muscle activity channels. Figure 4.5.3 shows an example of the NNMF algorithm applied to *Ball Roll* in a healthy patient. Previous research has shown that these synergies are relevant and stable in healthy patient upper extremity muscle activity patterns during isometric force tasks [46]. The technique also shows that synergies remain consistent across patients and change over the course of stroke recovery [47] These synergies can be linearly combined and compared to original sEMG data to understand the impact of the first more variance accounted for by few synergies, the less able the patient. To date this has been shown to be true with lower limb synergies and gait in patients with Cerebral Palsy [37]. Figure 4.5.3 shows an example of the NNMF algorithm deconstructing simple wrist flexion and extension during *Ball Roll*.
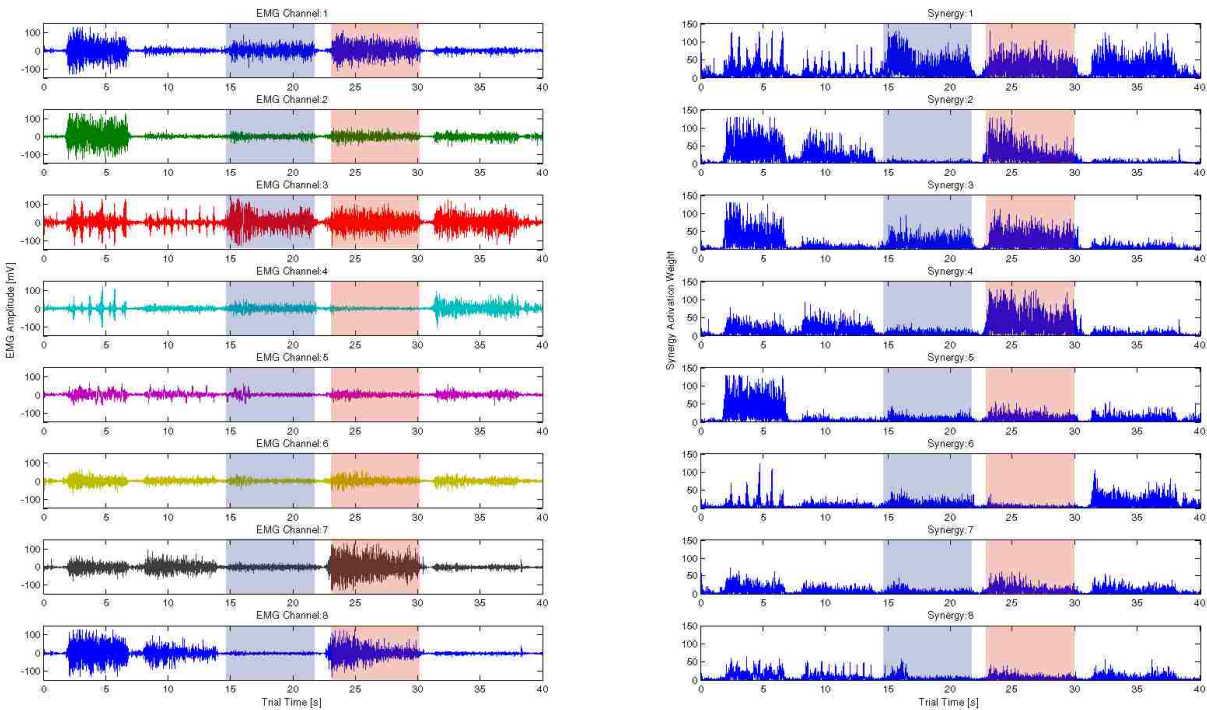
Figure 4.5.3. Example EMG Decomposition from Ball Roll Game. The left panel shows a snapshot of EMG data collected during *Ball Roll*. A sample extension epoch is highlighted in blue and a sample flexion epoch is highlighted in red. In the left panel of raw, randomly ordered EMG traces there are several channels that there are selectively active for flexion (Channels 7 and 8) compared to extension. Synergy activations plotted on the right are ordered by magnitude from 1 to 8. It is important to note that synergy 1 captures both flexion and extension activation patterns and has the greatest magnitude throughout, reflecting the contribution from all contributing electrodes. Subsequent synergies capture extension or flexion more dominantly.

4.5.F   *Tremor Characterization*

The fine kinematic tracking of hand position in relation to game object positions and timing allows us to explore quantifying and subtyping the status of different tremors as well. This will be detailed in cowriter's theses.

193

Table 4.5.1. All analytics for each module.

Module	Section	Analytics
Ball Roll	4.4.B.1	Wrist Flexion/Extension Range of Motion, Reaction Time, Muscle Complexity
Turn the Dial	4.4.B.2	Forearm Pronation/Supination Range of Motion, Reaction Time, Muscle Complexity
Reach and Hold	4.4.B.3	Arm Volume Range of Motion (3D), Reaction Time, Tremor
Reach and Grab	4.4.B.4	Arm Volume Range of Motion (3D), Reaction Time, Tremor
Pizza	4.4.B.5	Arm Volume Range of Motion (x-y plane), Reaction Time, Tremor, Failed Trial Ratio
Giant's Teeth	4.4B.6	Reaction Time,
Whack-a-Mole	4.4.B.7	Finger Flexion/Extension Range of Motion, Reaction Time,
Two Hand Shape Match	4.4.B.8	Reaction Time, Comparative Tremor, Coordination
Finger Position Match	4.4.B.9	Reaction Time, Finger Adduction/Abduction Range of Motion
Pillbox	4.4B.10	Reaction Time, Arm Volume Range of Motion, Failed Trial Ratio
Hand OpenCloseOpen	4.4.C.2	Maximum Voluntary Contraction, Muscle Complexity
BubblePopper	4.4.C.3	Arm Volume Range of Motion, Reaction Time
Painting ROM	4.4.C.3	Arm Volume Range of Motion
Wrist Range of Motion	4.4.C.4	Multiple Wrist Range of Motion

4.6 USE CASE: PILOTS AND EARLY FEEDBACK

4.6.A *Introduction*

We began testing the vHAB with patients and therapists early in the development of the system. This initially took the form of one-on-one sessions with therapists and professionals in the rehabilitation medicine space as described in Section 4.2, but this type of feedback was not enough to finish development of the system. From these sessions we received conflicting reports of healthcare regulations, needs of the therapists, and patient ability in relation to the system. While all of this information was useful, it made it very clear that we had to see the system in use to prioritize development milestones. To accomplish this, we set up partnerships with a commercial entity, the local science center, a hospital in-patient facility, and a skilled nursing facility. These pilots were not focused on collecting or validating data, but were highly organized use-case studies setup to collect usage statistics and end-user feedback. In the sections below we describe the genesis of each pilot, the state the vHAB system was in at the time, and the end results from the pilot, both in terms of system changes and end-user workflow.

4.6.B *Pilot 1: Skyline Retirement Community*

4.6.B.1 *Introduction and Methods*

Skyline Retirement Community (SRC) in Seattle, Washington, is a skilled nursing facility where we conducted our first pilot test of the vHAB system. Skilled nursing facilities present an interesting challenge in end user workflow integration, and our first pilot study elucidated many issues with our early user interfaces and system capabilities. For this pilot we used an early vHAB system without a muscle activity sensor. One system was deployed for a

three month period after a one hour training session with the therapists. Multiple follow up visits were scheduled during the course of the pilot to collect feedback and deliver updated software builds. The vHAB system was set up in SRC's therapy gym where a majority of their therapy was performed. Patients consisted of geriatric patients recovering from a variety of injuries, but no specific demographic information was collected.

4.6.B.2 Outcomes

As a first pilot, we received invaluable feedback relating to the use of vHAB in a skilled nursing facility. In retrospect, it is clear that the system was not yet ready for this level of testing and deployment, both in terms of software and system education. The system was used for three months and then reclaimed to collect and analyze the data; however due to a poorly implemented patient management structure we were unable to analyze patient specific data. The pilot officially ran from January 2015 until we reclaimed the system in September 2015, but the system was not heavily used during any time period. The reasons for this low usage became clear through the follow up sessions with their therapists.

Primarily, we had created a system that was too complex to use, containing too many modules and options. The therapists found the user interface difficult to navigate and the game objective too unclear. Our training protocol at the time consisted of one training session over lunch before leaving the system in the hands of the therapists. During this hour, we demonstrated the system, but provided no formal training, hands-on experiences, or left behind documentation. Further, this system did not have built-in tutorials or settings descriptions. These small design changes were fixed over time, but still left the primary issue of training and education. The therapists continued to use a patient account named "Test" instead of creating individual patients

because it was easier to do in their workflow, and, to their knowledge, individual patient accounts did not provide a clear benefit. The in-game settings were confusing and there were too many games to choose from. In a given day, they may have remembered how to use one of the games, but it may not have been useful for that specific patient. It was clear from this experience that training and continued education of the system are just as important as creating a useful system. Each subsequent pilot and experiment has contained a stronger focus on this aspect and has led to better experiences.

4.6.C *Pilot 2: Tacoma Lutheran Retirement Community*

4.6.C.1 Introduction and Methods

The Tacoma Lutheran Retirement Community (TLCR) in Tacoma, Washington, is a similar facility to SRC in both patient population and therapist composition. This pilot began a few months after the SRC pilot, which allowed us to refine both the training protocols and the actual vHAB software. For training on the system, we began with a 1 hour lunch training, but scaled back the amount of information we tried to cover and only overviewed three games. We then shadowed the therapists throughout the afternoon as they used vHAB with their patients. We answered questions and received feedback from all users. We also tested a few new software features, such as system auto-start, where vHAB would automatically launch if the system ever shut down. We returned to TLRC multiple times over a six month period to gather additional feedback and update their system with new features.

4.6.C.2 Outcomes

This pilot lasted from July 2015 to December 2015. The versions of the system utilized in this pilot were more sophisticated than that of the first pilot, but the success of the pilot came from the modified training protocols. Limiting the scope of training to just a few games made the therapists feel comfortable with the system and allowed them to focus on what patients in their case-load could benefit from using vHAB. This pilot led to the creation of the Pillbox game and the Finger Position Match game through personal conversations and iterative feedback. Difficulties in this pilot primarily arose from the distance of the facility from the system's developers. Updates were slow to deliver and it was difficult to troubleshoot the system over the phone with the therapists when something went wrong. These issues are inherent in a developing system and as such we sought a closer partner for our final pilot study.

4.6.D   *Pilot 3: Harborview Medical Center*

4.6.D.1 Introduction and Methods

Harborview Medical Center (HMC) in Seattle, Washington, hosts an inpatient therapy center that sees primarily stroke, spinal cord injury, and traumatic brain injury patients of varying demographics. This pilot saw quite a different set of users than the previous, skilled nursing facility, pilots and provided new opportunities in testing the limitations of the system. We initiated this pilot as a precursor for an at-home study (Section 4.8) to familiarize the therapists with the system so they could feel comfortable referring their patients to the study. Additionally, it allowed us to validate our previous changes made through the first two pilots for both our training protocols and system deployment. For this pilot, we deployed two systems to approximately 20 therapists over two separate training sessions. We followed a similar, limited

game set model, as in the TLRC pilot, but we further supplemented training with scheduled

hands-on experience for the therapists. We paired up the therapists and had them use the system

together, with one acting as the therapist and the other acting as the patient. This method drove

them to ask usage questions we had not heard before and expanded the types of conversations we

could have during the pilot.

4.6.D.2 Results and Outcomes

This pilot lasted approximately 4 months (December 2015 to March 2016) and saw over

30 patients of varying injury presentations. Though the system was used significantly more than

in the previous pilots, the numbers were not as high as we expected from a high capacity facility.

During the first few months, low usage was driven by the system setup in the facility. Due to

security reasons, the therapists locked up the tablets overnight. This meant that in order to use the

system, they needed to take it out of the locker, wait for the computer to start, plug in the

peripherals, and then start the vHAB software. This increased the chance of human error in the

setup, and, more importantly, took too long to easily include in a normal workflow. For the last

month of the pilot we added a tablet lock to the system and worked with the therapists to

establish a dedicated space for the system in their therapy gym. This increased visibility of the

system and reduced the workflow burden in using the system, leading to increased use during the

last month of the pilot.

4.6.E   *User Experience: Pacific Science Center*

4.6.E.1 Introduction and Methods

In July 2015 we were approached by the Pacific Science Center (PSC) in Seattle, Washington, to assist in the creation of an educational exhibit demonstrating new technologies from the University of Washington. We created a special build of the system consisting of a modified user interface that allowed users to select and play 1 of three games (*Shelf Grab, Whack-A-Mole,* and *Reach and Hold*). The primary purpose was to demonstrate how science and engineering can be used to help people in need, but we were able to collect general feedback from users to influence system design. We deployed one system in September 2015 and the exhibit ran until March 2016. Working with the PSC we designed a set of questions designed to help us collect feedback. For each game we asked "Tell us how much you did or did not enjoy each game" as a multiple choice and then "What can we do to improve the games?" as a freeform answer. We also asked for demographic information and "Do you have any other feedback?" to capture things we may have missed.

4.6.E.2 Outcomes

For these questions we collected the demographic information and general system satisfaction of 14 users (Figure 4.6.1). The system was used more than 14 times, but feedback was not a required component of the exhibit. There was not a measurable correlation between any of the demographic and user satisfaction, but this is likely due to the low user count.

Pacific Science Center Demographic Information

Pacific Science Center Feedback Summary
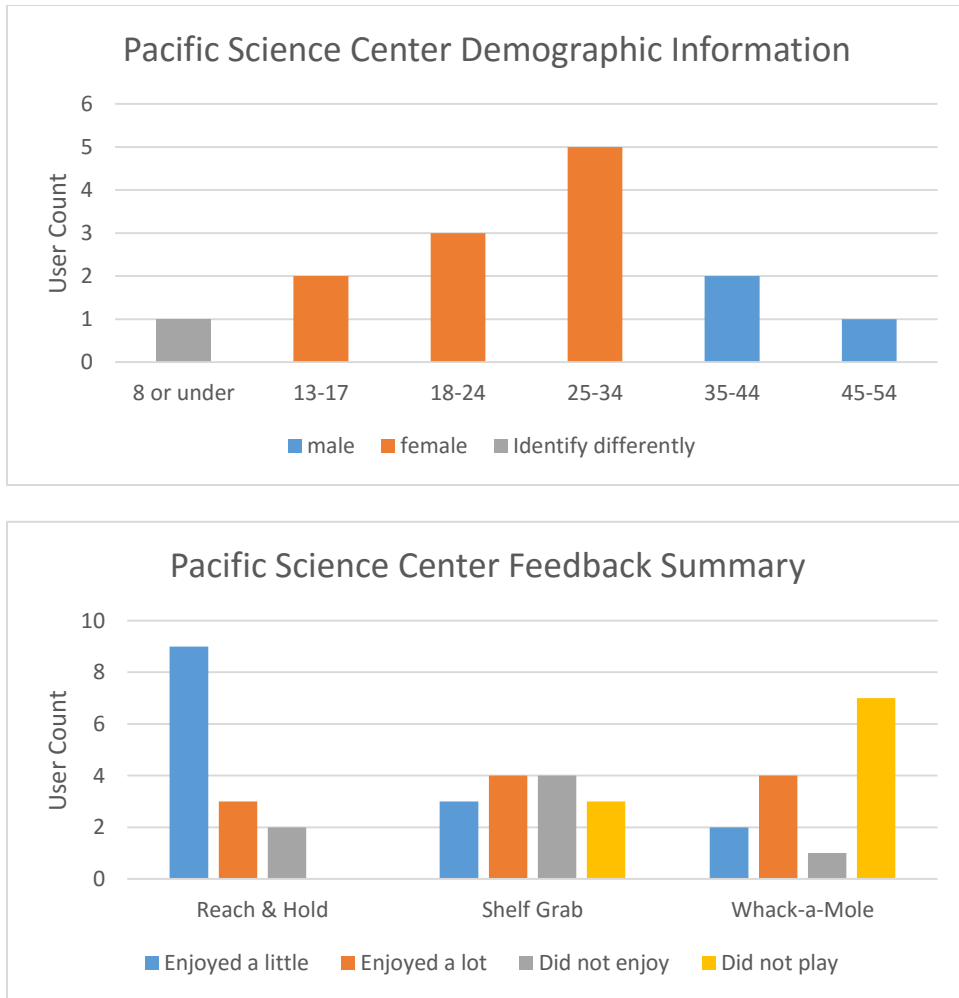
Figure 4.6.1. Pacific Science Center Feedback. Survey information was collected from 14 users over the course of 8 months of system deployment in an exhibit at the Pacific Science Center. Top. Demographic information from PSC survey. Bottom. Feedback summary information from PSC survey.

The special user interface did not contain tutorials nor did any user undergo training for the games. Brief descriptions of the games were provided in the exhibit, but there are no data as to how many people read the descriptions before attempting to play. Of the 14 surveys we received the feedback was primarily positive for each game, and complaints of the system were in the same vein as what we learned in the pilots with users stating:

"Instructions or rename. I had no idea what to do at first. I am used to touching things quickly in computer games."

"More explanation of how you're supposed to place your hand - this is for fingers, right?"

"Did not explain what you needed to do, I did not get it."

This feedback reinforced the need for robust instructions within the game. On the other end, however, some users were able to immediately understand the system and even develop strategies. This positive feedback was validating, but did not overshadow the need for system education.

4.7 USE CASE: HEALTHY SUBJECTS

4.7.A *Introduction*

The vHAB system aims to deliver therapy in an alternative way to traditional methods. Through vHAB's ability to automatically track hand kinematics and muscle activity we can create complex analytics (Section 4.5) that fully describe hand and arm function. Therapists, however, are not accustomed to seeing these types of measures, which presents a challenge in implementing the system in therapy facilities. One method of addressing this is by validating the measurements we take against traditional therapy measurements. For example, we could attempt to claim that the wrist angle minimums and maximums in *Ball Roll* are analogous to goniometer measures. This is quite difficult as *Ball Roll* provides the patient multiple attempts to reach this range of motion in a continuous fashion, while goniometer measures are performed under statically held conditions. Further, some analytics we have created, such as endurance curves, muscle complexity, and tremor do not have traditional analogs. To combat these challenges we set out to create a database of healthy user data.

This healthy user database (HUD) will eventually consist of age, gender, and handedness matched measures for most, if not all, of our analytics. Building this database will take considerable time and effort, but is the only reliable way to build maximum value benchmarks for recovery. Creating minimum values for the database is not likely possible as a zero value is possible in most analytics, but does not necessarily represent zero function. It is likely that through the analysis of recovery data we may be able to build models that future patients could be compared against. For initial deployments of the system, we can assume a linear model between zero and the average values in the HUD.

In this section, we describe preliminary attempts at building this healthy user database. Healthy subjects played multiple games in the vHAB system and demographic information and summary analytics were recorded. This work is by no means comprehensive and will require grander scale to fully implement. Further, some analytics, such as muscle complexity and tremor were not adequately collected in these initial results and had to be discarded from the HUD. Finally, it should be noted, that improvements to the system and the analytics were made in response to the collections of these values. This does not necessarily exclude them from the HUD, but instead will weight them less in the averages as we bring in newer data.

4.7.B *Methods*

All recruitment, data collection, and data storage methods were approved by the University of Washington Institutional Review Board prior to study onset.

4.7.B.1 Subject Recruitment

Subjects were recruited through public workplaces at the University of Washington campus in Seattle, WA. No exclusion criteria were used in this study, though subjects needed to be able to lift their hand above the sensor to collect the data. Subjects were recruited with a standard recruitment script explaining the purpose, risks, and benefits of the study and were consented before beginning the study.

4.7.B.2 Data Collection

Age, gender and handedness were collected prior to using the vHAB system. Subjects were seated in a chair and asked to adjust their position so that they were comfortable holding both of their hands six inches above the Leap sensor. Subjects performed all exercises below

with both hands, the order of which was chosen at random. Settings were consistent across all subjects.

First, subjects were led through an early version of our assessment platform that contained the 60 second Bubble Popper Module and the Wrist Range of Motion module. Subjects were then led through three separate games: *Whack-A-Mole* and two others chosen at random. Subjects were given the ability to become familiar with the game mechanics and hand movements in an unrestricted practice mode during which the researchers would provide instruction and answer questions from the subject. Practice was completed when the subject indicated they were comfortable with the game's mechanics and objectives with both their dominant and non-dominant hand. Practice mode was not repeated during the second hand play through. Data were then collected for 60 seconds of gameplay during which subjects were instructed to perform at maximum effort. This procedure was repeated for each of the three games. Subjects were allowed to rest as long as they wished in between each 60-second recording.

4.7.B.3 Data storage

Data were stored in a slightly modified version of the pID structure previously described. As one time users of the system there was no need to store their pIDs in a patient list for a specific therapist. Instead, all healthy subject demographic data were stored alongside a temporary healthy patient ID. No identifiable information was stored for these subjects.

4.7.B.4 Data Analysis

All raw data were processed in the methods described in Section 4.5 when the appropriate data were available to complete the analysis. Analytics were discarded if not enough trials were completed or if the analytic method changed significantly during the course of subject recordings. No muscle activity data were recorded due to technical issues at the time of recording. This primarily left range of motion, reaction time, and other gameplay related analytics. Standard deviations are used within the analytics to define a subject's trial-to-trial variance. These are either utilized to trim or smooth the values, as in range of motion measures, or are presented as is next to the mean value. Standard errors were used when comparing subject to subject variability in the measurements.

Comparisons of performance between which hand was used (left or right, dominant or non-dominant) and practice vs performance gameplay are calculated using a two sample t-test. Comparisons between left and right hands split the performance metrics based on the hand used during the module. Hand dominance compares performance metrics between dominant and non-dominant hands. Practice-Performance compares all practice values to performance values during the 60-second tests. Practice mode could be completed with either hand. Therefore the comparison between some metrics is excluded as time could be spent switching hands.

4.7.C *Results*

4.7.C.1 Demographics

Data were collected from 17 subjects (11 Female; 14 right handed; Age 42.53±14.4). Subjects played 12 different modules at unequal frequencies. Aside from challenges with our

randomization methods, some games, like the Trace Game, were removed from the game pool due to difficulties in system performance. The intersectionality of demographics and game types prevents any significant analysis comparing precise subject groups partially due to all male subjects being right handed. Handedness was utilized to define whether the patient was using their dominant or non-dominant hand. Additional patient populations will be required to build a HUD that can truly be age, gender and handedness matched.

4.7.C.2 Gameplay Analytics

Data were collected across all 17 patients to produce 31 different gameplay analytics (Table 4.7.1). Population means were not significantly different across either of the three comparisons (hand dominance, play order, or practice) in 30 of the 31 analytics, though it is important to note that some statistics could not be computed due to missing information or low trial numbers. The one analytic that showed a significant difference was *Reach and Hold reaction time* comparing practice to performance (p= 0.021). Across all gameplay analytics, standard errors were relatively low, indicating that these values are a good benchmark for the creation of an early version of the HUD.

4.7.C.3 Range of Motion Summary

Range of motion analytics were only performed on four modules (Table 4.7.2), primarily due to changes in data saving and analytics structures that occurred during or after this study that significantly changed the collected values. Measurements were similar across patients, with relatively low standard errors, indicating that these values represent a basis for comparison with the HUD. However, a number of the angles from the wrist ROM were either lower or higher than literature reported goniometer values [48]. As previously discussed, this could be caused by the

207

differences in measurement techniques, but either way warrants further investigation. Statistically significant differences were apparent in both the left-right hand and dominant-non dominant hand comparisons for the Thumb range of motion measurements (p = 0.025 and p= 0.035 respectively).

4.7.C.4 Other Measurements

During *Two Hand Shape Match,* we were also able to measure tremor (Table 4.7.3). These values had a much higher standard error than the other measures indicating that the analysis metrics may need to be refined before including these values in the HUD. Further the actual standard deviation measures with a subject are higher than the mean values, indicating that the variability in the measure is too high to be of use in a clinical setting. Adjustments to these values have been made since this test to improve these measurements for future tests.

Table 4.7.1. Gameplay Analytics for Healthy Subjects. All presented values for reaction times are in seconds and all other values are unitless.

Game	Gameplay Metric	Mean ± ste	Total Count	p-values (Two Sample t-test)		
				Left-Right	Hand Dominance	Practice-Performance
BallRoll	ReactionTimeMean	3.7 ± 0.3	10	0.590	0.669	0.606
	ReactionTimeStd	1.5 ± 0.3	10	0.318	0.426	0.738
	Score/Minute	16.7 ± 1.3	10	0.591	0.684	-
Turn the Dial	ReactionTimeMean	5 ± 0.8	6	0.250	0.250	0.877
	ReactionTimeStd	4.2 ± 1.6	6	0.478	0.478	0.959
	Score/Minute	17.2 ± 2.9	8	0.216	0.216	-
Pizza	ReactionTimeMean	10.7 ± 1.9	6	0.860	0.860	0.116
	ReactionTimeStd	7.7 ± 3.2	6	0.757	0.757	0.893
	Score/Minute	5.4 ± 0.9	6	0.641	0.641	-
Reach and Hold	ReactionTimeMean	3.8 ± 0.2	4	0.147	0.147	**0.021***
	ReactionTimeStd	1.8 ± 0.4	4	0.179	0.179	0.103
	Score/Minute	15.7 ± 0.9	4	0.162	0.162	-
Whack-A-Mole	IndexReactionMean	0.3 ± 0	30	0.863	0.679	0.861
	IndexReactionStd	0.4 ± 0	30	0.880	0.782	0.547
	MiddleReactionMean	0.4 ± 0	30	0.630	0.550	0.742
	MiddleReactionStd	0.4 ± 0	30	0.236	0.266	0.950
	PinkyReactionMean	0.3 ± 0.1	27	0.673	0.435	0.383
	PinkyReactionStd	0.3 ± 0.1	27	0.481	0.285	0.257
	RingReactionMean	0.4 ± 0	29	0.529	0.390	0.086
	RingReactionStd	0.5 ± 0.1	29	0.434	0.171	0.098
	ThumbReactionMean	0.4 ± 0.1	27	0.148	0.176	0.730
	ThumbReactionStd	0.5 ± 0.1	27	0.613	0.523	0.109
	Score/Minute	52.3 ± 3	34	0.203	0.561	-
Two Hand Shape Match	ReactionTimeMean	3.9 ± 0.4	8	0.327	0.369	0.697
	ReactionTimeStd	2.3 ± 0.6	8	0.301	0.375	0.971
	Score/Minute	16.4 ± 1.3	8	0.439	0.557	-
Bubble Popper	BubblesPopped	30.8 ± 2.6	18	-	-	-
Giant's Teeth	Score/Minute	55.6 ± 10.5	10	0.876	0.876	-
Trace Game	Score/Minute	3.9 ± 1.2	4	0.764	0.764	-
Pinch Fall	Score/Minute	16.1 ± 0	2	-	-	-
Reach and Grab	Score/Minute	5.7 ± 0.8	14	0.929	0.929	-

209

Table 4.7.2. Range of Motion analytics for healthy subjects.

Module	Gameplay Metric	Mean ± ste	Total Count	p-values (Two Sample t-test)		
				Left-Right	Hand Dominance	Practice-Performance
BallRoll	'MaxAngle'	54.4 ± 3.6	10	0.157	0.979	0.649
	'MinAngle'	-50.9 ± 5.4	10	0.970	0.828	0.851
Whack-A-Mole	'IndexRom'	126.8 ± 5.6	30	0.478	0.332	0.628
	'MiddleRom'	134.8 ± 5.3	30	0.520	0.158	0.411
	'PinkyRom'	108 ± 5.2	30	0.261	0.423	0.596
	'RingRom'	127.6 ± 5.7	30	0.120	0.077	0.232
	'ThumbRom'	107.1 ± 5.4	30	**0.025***	**0.035***	0.222
Wrist ROM	'HorizontalExtensionMax'	58.4 ± 3.2	17	-	-	-
	'HorizontalFlexionMin'	-62.3 ± 3.5	17	-	-	-
	'SupinationMax'	74.9 ± 9.5	17	-	-	-
	'SupinationMin'	-61.6 ± 9.1	17	-	-	-
	'VerticalExtensionMax'	55.1 ± 3.9	17	-	-	-
	'VerticalFlexionMin'	-79.9 ± 2.4	17	-	-	-
	'UlnarMax'	42.3 ± 2.8	17	-	-	-
	'RadialMin'	-38.5 ± 2.2	17	-	-	-
Bubble Popper	'ReachVolumeFraction'	0.6 ± 0.1	17	-	-	-

Table 4.7.3. Tremor Analytics from healthy subjects.

Game	Gameplay Metric	Mean ± ste	Total Count	p-values (Two Sample t-test)		
				Left-Right	Hand Dominance	Practice-Performance
Two Hand Shape Match	'LeftHandTremorMean'	26.9 ± 10	8	0.403	0.633	1.000
	'LeftHandTremorStd'	31.8 ± 15.8	8	0.368	0.375	0.906
	'RightHandTremorMean'	21.8 ± 7	8	0.233	0.797	0.259
	'RightHandTremorStd'	29 ± 9.6	8	0.216	0.956	0.167

4.7.D   *Discussion*

4.7.D.1 Standard deviations vs standard errors

Most of the gameplay and tremor analytics contain both a subject mean and standard deviation and a population mean and standard error. This population does not represent the true population in a traditionally statistical sense, but is the current standard to build against for the HUD. While standard errors across subjects tended to be low, this does not mean that there is not significant variability within a subject over the course a game or assessment. In fact, many of the standard deviations we recorded were within close range of the value means. This variability is to be expected in many of the games for first-time users and is likely to decrease with continued play. Models may need to be adjusted to account for amount of time each game has been played to build accurate models of learning vs recovery of the system.

4.7.D.2 Value comparisons

Most of the values collected did not show significant differences in any of our three test conditions. Of the two analytics that did show significance, it is unclear what may have caused this difference. Low trial numbers put the *Reach and Hold reaction time* value into question, but the difference does make sense in terms of overall game difficulty compared to the other games. *Reach and Hold* is the only included game that requires three-dimensional hand movement, and as mentioned previously, learning how to map hand depth to the virtual space is one of the most challenging aspects of the system to learn. The *Whack-A-Mole thumb range of motion* analytic is a more curious case as it was significant for both the right-left and the dominance comparisons. This may be influenced by the fact that 14 of the patients were right handed, but p–values for the other metrics do not show a direct relationship between the two comparisons. Slight differences

between reaction times and ability are likely to occur between hands, but these variations would require much further testing to determine. Instead, these differences could be caused by positioning of the hands above the sensor during the gameplay and the thumb moving in a different plane from the other fingers. This could be verified by systematically testing these values with the hand at varying points over the sensor.

4.7.D.3 Building model of healthy patient data

The above data represent a significant start towards collecting comparative data to use within the vHAB system to help therapists and clinicians interpret vHAB data. As our first implementation of many of these analytics at a large scale it became clear that many of them did not perform as expected, causing system crashes and lags. This experiment also exposed a flaw in the muscle activity recording system, which was good for system development, but bad for building a HUD. Regardless of these missteps, we were still able to collect data for 51 different analytics over 17 subjects. Many of these values had low standard errors, leading us to feel confident in the creation of an early version of the HUD. This set of data only represents one set of vHAB settings which will impact all comparisons of the gameplay analytics. While it may not be feasible to test all settings for all games, a few additional settings need to be tested to finalize an accurate model.

Our population was inherently biased and not immediately relatable to all of our end-user demographics. Our mean age of 42 does not represent the traditional age of a stroke (66% > 65) [49] nor does it match users in a traditional skilled nursing facility. Further, individuals were recruited from workspaces around a university, which may have produced individuals more familiar with technology than the average population. These biases and the issues mentioned

above need to be accounted for in building the full model. While we are still far from

implementing the HUD within the vHAB system, this early version can be used to perform

internal testing and validation.

4.8   USE CASE: HOME ADHERENCE STUDY

4.8.A   *Introduction*

Adherence to a therapy regime after an injury drops off significantly when the patients are sent home [50]. Traditionally, they are sent home with sheets of paper instructions that provide pictures and diagrams of their therapy exercises. In addition to this format lacking any form of enjoyment, patients are not given any feedback as to whether they are performing the exercises correctly or if they are getting any better [51, 52]. In a set of key surveys many patients indicated that they would be more likely to complete their therapy if alternative, engaging, exercises were available [50].

vHAB was created to solve this problem, but there are a few key questions left to answer. First, does vHAB provide access to engaging exercises at home, or are they as boring as traditional methods? Second, does vHAB actually increase adherence to therapy? Finally, does using vHAB improve outcomes for these patients? This question is much more difficult to answer, as developing proper population sizes for outcome studies after injury is a time intensive and expensive task.

Our main driving hypothesis is that **increased adherence through engaging home exercises will lead to better outcomes** simply because patients are performing exercises that would otherwise be skipped. Since vHAB was built directly from existing therapy exercises, it is likely that using vHAB will be equivalent to patients performing these exercises on their own. It is possible that vHAB will lead to better outcomes in the home setting since the system provides immediate feedback and measures of progress in real time, further enhancing the patient's ability to target problem areas of their movement. However, proving that vHAB is better than traditional

exercises is not a line of inquiry we intend to pursue due to the challenges listed above. Instead, we have created a home adherence study that primarily looks at utilization data of the vHAB system and relies on patient feedback to measure system engagement.

In this study, patients who recently suffered a neurological trauma, such as stroke or spinal cord injury, received a vHAB system to use at home over an 8 week period. During this time, we collected usage data, a set of analytics on gameplay and function, and verbal feedback relating to their use of the system. vHAB system use was then compared to how long the patients performed traditional therapy over the same length of time using a custom designed tablet log system. This study is still ongoing at the time of this writing, but below we present the preliminary results for two patients who have completed key milestones within the study.

4.8.B   *Methods*

4.8.B.1 Study Criteria

Study participants had a recent injury causing upper extremity impairment and were within six months of being discharged from an inpatient facility at the start of the study. Patients were referred directly through their physician, therapist, or other official caregiver, who believed the patient could be a good candidate for extended therapy with the vHAB system. Patients could not participate in the study if they 1) had contraindications for using the muscle activity armband, such as implantable devices (pacemaker, Baclofen shunt) or skin lesions or rash on the forearm, 2) had cognitive deficits as a result of their injury, demonstrated by scoring 5 or more errors on the Pfeiffer Short Portable Mental Status Questionnaire, or 3) vision or hearing impairment. The following study was approved by the University of Washington Institutional Review Board prior to subject recruitment.

215

The study was designed to compare usage data between the vHAB system and traditional therapy. In addition to asking patients to recount the duration and time of day of their exercises we designed a custom therapy logging system (Figure 4.8.1) that could be sent home with the patient. This tablet log consisted of a tablet computer and custom software that allowed patients to track how often, and how long they performed their therapy exercises. The software emulated a simple stop watch, where patients would press "Start" when they began their therapy, and "Stop" when they were finished. The amount of elapsed time was then saved to a log file on the local system for future analysis. Patients could also "Pause" the timer if they were temporarily stopping their therapy exercises but intended to continue.

Figure 4.8.1 Tablet Log software. This software was sent home with patients during the Table Log phase of the home adherence study. Patients could use the timer function of the software to track when and how long they performed their therapy exercises.

4.8.B.3 Study Design

The study utilized a delay cross-over method to maximize potential results with smaller patient numbers (Figure 4.8.2). In this method, patients were divided into two groups. In the first 8 weeks Group 1 received the vHAB system and Group 2 received a tablet log. After eight weeks Group 1 received the tablet log system, while Group 2 received a vHAB system. Each group used these systems for 8 weeks. In this method, usage data can be compared both across and within each group. Upper extremity function was assessed periodically during the study using traditional assessment techniques, such as the Wolf-Motor Function Test, the Box and Blocks test, and a dynamometer measurement. Assessments were performed by a therapist trained in each method. In the event that patients could not perform the minimum requirements for the tests, these assessments could be skipped. In addition to these motor measurements, the Assessment phases included a verbal questionnaire relating to the use of the vHAB system and general function.

4.8.B.4 vHAB Customization

The vHAB software was modified slightly for this study to allow for easier level navigation. Patient and therapist management user interface screens were removed so that when launched, the patient was presented with the game select screen (Figure 4.4.10). A smaller, simplified version of the data visualization screen was added to the game select screen so that patients could easily view any collected analytics. Settings were set during the first Assessment period (A1) and adjusted at each based on the patients current ability. The garden wrapper was utilized with the game point to progress point ratios set to optimally grow the garden over eight weeks to align with the study length. Finally, patients were presented with a limited set of games

based on an early assessment of their functional ability. For example, we did not include *Whack-A-Mole* in their game set if they could not move individual fingers. This game list would be reassessed during each of the Assessment periods in the study.
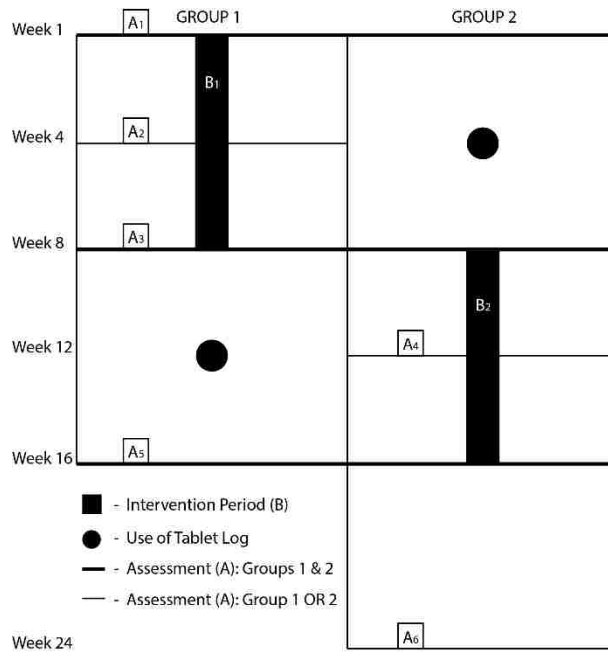
Figure 4.8.2. Home Adherence Study Design Diagram. Patients were divided into two Groups which dictated what system they received first (vHAB/Intervention) or the tablet log. Assessments were performed periodically during the course of the study.

4.8.B.5 Analysis Methods

Usage data consisted of both how long and how often the subjects utilized vHAB or the Tablet Log systems. Cumulative use time for the vHAB system only included time spent within a game module, but did include "rest" periods where the hand was not above the sensor but returned within 5 minutes. Usage of the Tablet Log consisted of all timer log data since patients could pause the system if they were not performing their exercises. vHAB analytics were tracked over the entire 8 week usage period.

4.8.C *Results*

4.8.C.1 Demographics

      Two patients with complete spinal cord injuries (C3 and C4) have participated in the

study at the time of this writing. While this does not provide enough data to make reliable

conclusions regarding system adherence, the data collected so far provides valuable insights into

system use. Both patients were unable to perform the physical assessments during the

Assessment phase of the study, but game data, analytics, and questionnaire data were still

collected.

      Both patients were assigned to group 1 and received the vHAB system first. Assessments

did not take place at 4 and 8 weeks due to scheduling conflicts. For P01, A2 occurred at day 58,

and A3 at day 100. For P02, A2 occurred on day 65 and A3 at day 106. These days are very far

outside our study protocol, but may be the norm in trying to schedule visits for individuals with

such high levels of injury as it requires high levels of coordination between care staff and the

patient.

4.8.C.2 Adherence Data

      Both patients utilized the system more before the second assessment (A2) than after

(Figure 4.8.3). Reasons for this are further explored in the discussion points below. Early data

(Day<50) show increasing usage over time as patients become familiar with the system. Usage

days are low (P01:8, P02:14), but each day can contain multiple sessions with different game

sessions.
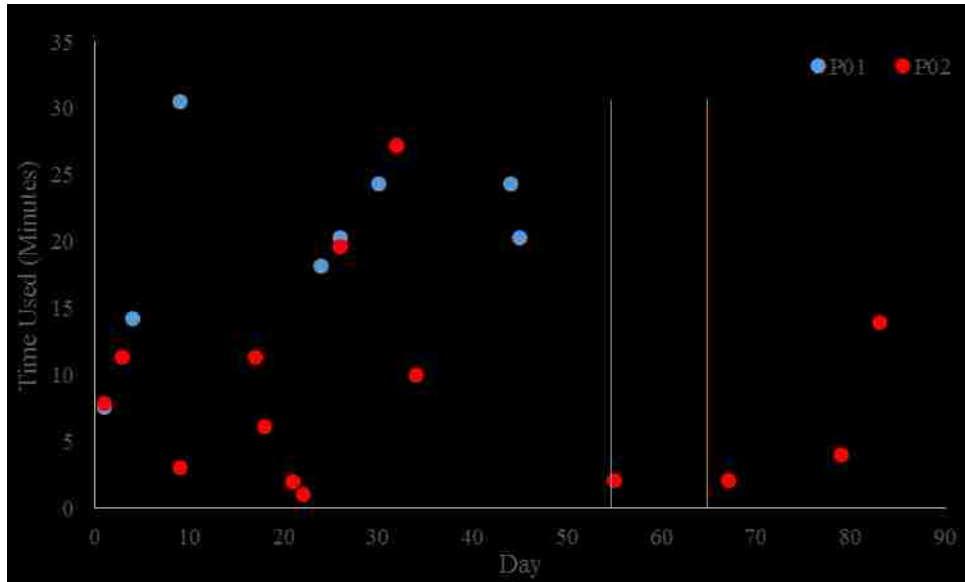
Figure 4.8.3. vHAB usage data in home adherence study. Usage days for P01 (blue) and P02 (orange) were composed of all the individual sessions for that day. Day to day usage was sparse, but some days showed high usage, such as P01 day 9, where the patient used the system 3 separate times playing multiple games each time. Assessment 2 (A2) bars are shown for reference for each patient. It is important to note that P01 did not use the system at all after A2, and P02 barely used the system. P02 preferred *Ball Roll* heavily with 48 play instances, while P01 spilt their time between five different games nearly equally.

4.8.C.3 Analytics Summary

Analytics were performed on most of the sessions for both patients (Table 4.8.1). Some analytics, such as *Ball Roll Max Angle*, could not be calculated for all sessions due to short play times or poor data quality. Other analytics, such as *Turn The Dial* range of motion were not calculated due to an error in the data recordings. The standard deviations of the analytics were relatively high, likely due to both fatigue and eventual learning of the tasks. Most games were not played often enough to extract significant changes over time. Figure 4.8.4 shows *Ball Roll*

220

angle measurements for P01 (left) and P02 (right) for every play session. Figure 4.8.5 shows the

reaction times for those same sessions for P02, who played the game enough times to show

decreases in reaction time over time.

Table 4.8.1 Home subject analytics data.

Game	Metric	P01 Mean ± Std	N	P02 Mean ± Std	N
Ball Roll	Max Angle	34.5 ± 14.1	16	56.2 ± 16.8	45
	Min Angle	-45.8 ± 22.5	16	-44.8 ± 15	46
	Reaction Time Mean	9.4 ± 3.9	16	3.4 ± 2.4	48
	Reaction Time Std	10.1 ± 4	16	2.5 ± 3.2	48
Turn The Dial	Reaction Time Mean	9.5 ± 4.2	18	3.4 ± 1.6	4
	Reaction Time Std	11.1 ± 5.6	18	10.4 ± 10.6	4
Reach and Grab	Reaction Time Mean	16.8 ± 10.7	14	10.5 ± 2.8	3
	Reaction Time Std	11.4 ± 11.5	14	5.9 ± 2.7	3
Pizza	Reaction Time Mean	5.4 ± 1.9	14	9.1	1
	Reaction Time Std	5.4 ± 3.9	14	3.9	1
Reach and Hold	Reaction Time Mean	6.1 ± 5.1	15	9.4	1
	Reaction Time Std	6.4 ± 8.7	15	3.0	1
Two Hand Shape Match	Left Tremor Mean	18.6	1	-	-
	Left Tremor Std	34.0	1	-	-
	Reaction Time Mean	4.3	1	-	-
	Reaction Time Std	2.2	1	-	-
	Right Tremor Mean	7.5	1	-	-
	Left Tremor Std	8.0	1	-	-
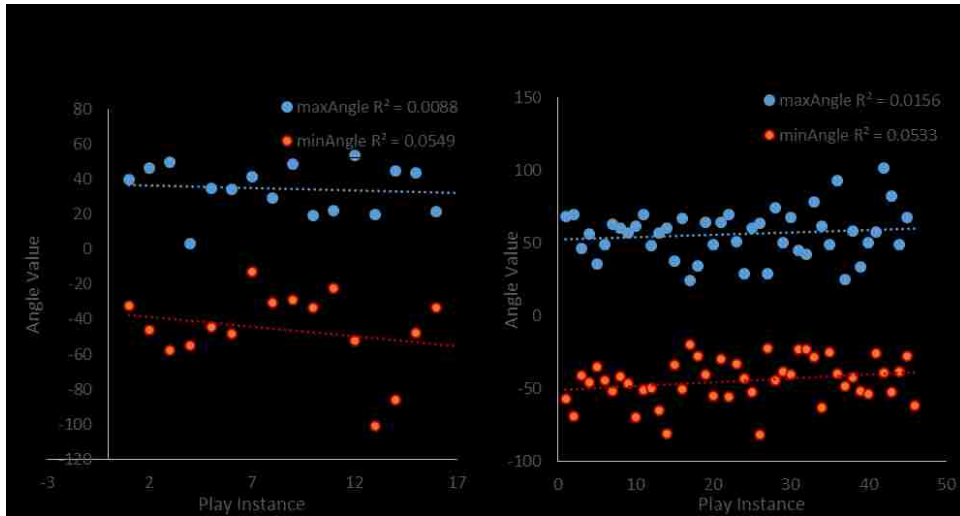
Figure 4.8.4 Ball Roll wrist angle measurements during vHAB home use. Neither the minimum or maximum angles for wrist flexion-extension in *Ball Roll* increased significantly over time for either patient. It is important to note, however, that both patients played the game at the highest sensitivity settings throughout the entire course of the study. Thus, it may have been that they did not need to achieve angles greater than displayed (~50 degrees) to achieve the game's objective.

Figure 4.8.5 Ball Roll Reaction Time measurements during vHAB home use. Here we present the only measurable change in game performance. The change in reaction time mean is accompanied by a decreasing change in the standard deviation. This means that the trial-to-trial variability within a given

game session decreased as well. It is difficult to tell, with limited data, if this decrease is caused by increasing familiarity with the game or by actual functional improvement.

4.8.C.4 Tablet Log Use

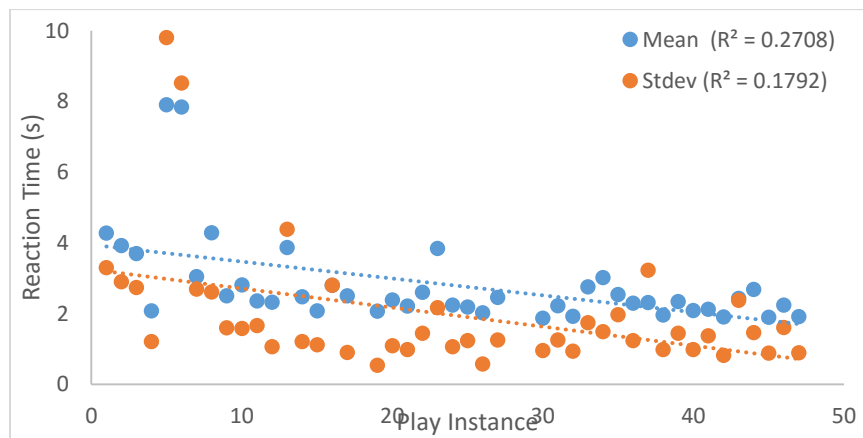Neither patient utilized the tablet log system during the control portions of the study. This does not mean that they are did not perform therapy during this time.

4.8.C.5 Survey Data

Survey data were collected from both patients regarding their experience with the vHAB system. Table 4.8.2 presents some of the more relevant responses. Questions relating to injury, demographics, and their perception of gamified therapy were removed as the responses are not relevant at this time. Both patients appeared to have positive experiences with the vHAB system and both would recommend using the vHAB system to others. A common theme among P02, was that the system was great "When it works." Hardware issues, discussed below, may have influenced this opinion. P01 was very motivated by the garden metaphor, which he found surprising. Multiple suggestions for improvements came from this survey data as well, such as new tutorial and educational components, and assistive control elements.

Table 4.8.2. Survey response questions from vHAB home use

Phase	Question	P01	P02
Before Using vHAB	How often do you engage with your prescribed home therapy tasks?	once a month	no set time, when able
	What is the most difficult part about carrying out your prescribed home therapy tasks?	adding the therapy time into daily routine	frustration about not being able to do things including exercises
	What would make your engagement with your prescribed home therapy tasks easier or more enjoyable?	keep a log, held accountable	faster results, seeing results
After first use	Did you enjoy using the system? Why or why not?	yes - encouraging to do exercise	Yes!
A2	Describe your experience with the vHAB system thus far:	Biggest struggle is daily routine. "Surprisingly motivated by garden metaphor!" Notice getting fatigued during it, depending on time of day. Try not to get frustrated with reach and grab. Run through all games 2x (not two hand shape match), second time through is slower due to fatigue. Get tense and activate other muscles not involved with the task -> has had same issue with different rehabilitation tech.	Great when it works! Can feel it exercising his body, can feel the benefits
	Would you recommend vHAB to someone in a position similar to yourself for their home therapy needs? Why or why not?	Yes, one of the big perks is doing it at home. Time management makes it much easier on the daily routine.	Yes engaging and fun, when it works
A3	Recommendations for improvement	While doing exercises it would be helpful to have someone to consult with to say "remember to activate your shoulder while relaxing your back for movement x". Prompts on the screen could also work.	Voice activated vHAB control. Smooth experience without bugs would make it better.
	Did the vHAB system keep you motivated to do your daily therapy exercises? Why or why not?	More successful when it was set up as a station. Surprised by how motivating the garden aspect was	It is a 2 person project to get started, but great when it works
	Would you continue to use the vHAB system once this study has ended for your home therapy needs? Why or why not?	If I could be shown what muscles to work on for the particular exercise it would be more helpful.	I would
	Would you recommend vHAB to someone in a position similar to yourself for their home therapy needs? Why or why not?	yes I would, particularly if people are already independent then this would be an even bigger motivating factor	yes, something different and new to do

4.8.D   *Discussion*

4.8.D.1 Comparing to model of healthy patient data

Section 7 of this chapter aimed to create a healthy user database of vHAB metric data. This accumulated data are still incomplete, but provides an interesting benchmark to start comparing the two home patients to. *Ball Roll* minimum and maximum angle values for both patients are slightly lower than the healthy user data but still within 1 standard deviation from the healthy users. Reaction time data for the home patients were higher in almost all cases than healthy users. As demonstrated by Figure 4.8.5, however, P02 was able to work to get closer to this norm value. These tools will be very helpful in the continuation of this study and the future use of the vHAB system.

4.8.D.2 Study Populations

Initial protocols for this study called for unilateral injuries and study enrollment within 2 months of being discharged from an inpatient facility. Patient recruitment with these constraints was nearly impossible for our small team. We were interested in these enforcing these limitations since a unilateral injury increases the likelihood that the patient can set up the system on their own. The shorter discharge time was to truly monitor whether they were increasing adherence significantly with the vHAB system when they would otherwise be likely to do therapy. In our case, both patients were close to 6 months post discharge and were not previously engaged in therapy answering – "When I have time" and "Once a month" – to the pre-study question how often do you do therapy exercises now. We definitely increased the frequency of therapy compared to these answers, but it may have had even more impact if it had been used sooner.

4.8.D.3 Low Usage Data

The two patients we have seen thus far had severely limited function in their hand and arm. They were able to play many of the simpler games at high sensitivity settings; however, we were not able to reliably perform traditional physical assessments throughout the course of the study. This also meant that these patients had a difficult time navigating the vHAB UI on their own. Both patients utilized home health care professionals to help them navigate the vHAB system at home, but this may not be possible for many of our end users. Future iterations of a home-use vHAB system may contain alternative level selection methods, such as voice controls, to assist individuals with more severe impairments.

P01 noted that the system was much easier to use during the first few weeks since it was "setup as a station." At the beginning of the study, our team traveled to his location to set up the system on a table in the living room. For A2, the system was dismantled and brought in for an update and then sent back with P01. Though it is not possible to know for sure, it is likely the system was never set back which explains the lack of usage data. P02 may have had a similar issue post-A2. He reported hardware failures that we attempted to troubleshoot over phone, but were not able to get consistently working. The usage in Figure 4.8.3 does not show multiple 0 score attempts throughout days 60-100, where it is likely the system was not working. This could have been from poor hardware setup, or from as yet undiagnosed software issues. Either way, hardware assembly processes will be increasingly important in providing a system patients can truly use at home.

4.8.D.4 Tracking traditional adherence

Tracking adherence without a digital system is inherently difficult. It may be weeks in between traditional physician or outpatient visits and relying on a patient's memory of how long and when they did there therapy is not a reliable method for gathering true data. Additionally, patients may not provide accurate results due to various social pressures that accompany injury. These problems presented challenges in designing this study. Providing scientifically accurate values of adherence would require an unbiased observer to watch the patient perform their traditional therapy exercises. It is important to note that the presence of the observer could further influence the patient's adherence to the study. Having the patient fill out pen and paper values can also be unreliable since they may fill them in all at once, just before the study organizers view the data.

Our solution was to implement the Tablet Log system. If patients attempted to rack up multiple sessions immediately prior to an assessment phase, we could examine the time stamps on the data to discount the data. Additional patterns of activity can also be flushed out of the collected data, such as what a "normal" session would look like for that patient. However, this system still has flaws. The presence of the Tablet Log may influence adherence to therapy itself, as the tablet hardware serves as a reminder to perform the therapy. Further, patients may find the timer useful in ensuring they perform 15 minutes of therapy, when they previously may have stopped short at 12 minutes. Finally, if patients truly desired to, they could start the timer while performing other tasks and set an alarm to remind them to stop the system. It is unlikely that incentives would drive patients to this extreme, but it is nonetheless important when examining the significance of the data.

It may be that outcome measures are the only true way to measure actual adherence, which is contrary to the goals of our central hypothesis. True adherence comparisons may be a good corollary to outcomes, but adherence may prove too difficult to measure with traditional exercises. Alternative methods will continue to be explored as we develop new studies with the vHAB system. One partial solution would be to include a leap motion sensor with the Tablet Log, and have patients perform their traditional exercises over the sensor. This would allow for direct study of the impact of the gamification, movement feedback, and real time analytics on adherence to therapy.

4.8.D.5 Impact of System for Home Use

Based on survey responses, both patients enjoyed using the system at home during the study. Taking this alongside the usage data, indicates that the system needs to be perfected from not only the software and hardware side, but from a setup and accessibility standpoint. With limited data, both from small subject size and low usage data, it is difficult to extract outcomes or any main conclusions regarding adherence.

4.9   FINAL THOUGHTS

Here we have described the creation and testing of vHAB, a gamified therapy and assessment platform designed to enhance rehabilitation after neuromuscular trauma. vHAB consists of both custom software and carefully selected commodity hardware, crafted to meet six design principles aimed at making the system accessible to as many people as possible. Early pilot studies exposed some of the key challenges with implementing any sort of new technology into a health care environment. Multiple iterations of the vHAB design have prepared us to overcome these barriers, allowing vHAB to impact the lives of countless patients. In the following sections, we further examine these barriers and present solutions that may become integral parts of vHAB in the future.

4.9.A   *Importance of Commercialization*

Reaching new patients and facilities is a challenge with any new product. From an academic environment, we have reaped many benefits in making new contacts, both from end-users and field experts. This, however, did not necessarily lead to usage of the systems. Anyone can accept a "free" pilot study, but that does not necessarily mean they will use the system. When entities, especially high-efficiency therapy facilities, have to pay for something the incentives become aligned in using the system. It takes more work upfront to justify the use of the system to the facilities, but through the process everyone understands the benefits and limitations of the system. Then, each month of the system's use is carefully monitored for efficiency and benefits to ensure that it is worth paying for. This may be risky early on in a product's development, but would provide more detailed feedback and usage data for continued

iterations. It is only through commercialization of vHAB that we will be able to reach this level of engagement with our end-users.

Another key benefit of commercialization of vHAB is the long-term scalability and reach of the system. An academic project cannot manufacture, distribute, and provide the support infrastructure for a system like vHAB. When thinking of vHAB within a corporate infrastructure, however, there are many other key engineering factors that come into play. Cost of the system was always an important consideration, but now the cost models need to be sustainable and accessible. Shipping and system setup become even more important, since we will not be able to personally set up systems as in Section 4.8. Finally, technical support and system updates require additional engineering development to ensure the system's longevity. These topics are not the key focus of this writing, but are worth mentioning to fully describe the future of the system.

4.9.B   *Cost and efficiency in healthcare*

We set out to design vHAB to be accessible in price for facilities and end-users at home, but changes in healthcare legislation make this challenge a moving target. For facilities, therapists charge insurance based on skilled therapy reimbursement codes often billed in 15 minute increments. The definition of "skilled" therapy is murky and based more on a facility's experience with reimbursement rejections and documentation practices than an actual true definition. In talking with therapists, it ideally means activity or exercise in which the patient is trying, challenged, and/or engaged. This also means that there is one therapist per patient to monitor the patient and adjust the task when appropriate. Facilities tend to operate at very high efficiencies to maximize profits, meaning 80-90% of the day must be providing skilled therapy

for each therapist. Anything that detracts from this time, such as documentation, system setup, or training may not be adopted.

vHAB can be the exercise that the patient is doing during skilled therapy, but current codes do not allow for extra reimbursement to cover the cost of vHAB. Instead, vHAB must provide additional benefit over traditional therapy to fit into the facilities. These benefits must also outweigh any setup time that may not be present in traditional modalities. Currently, vHAB's main additional benefits come from increased engagement and better insights into recovery. Increased engagement may allow patients to complete their therapy when they otherwise would not perform skilled therapy for a full 15 minutes. Recovery insights through our assessment and analytics systems may allow therapists to better target areas of difficulty or keep patients motivated, but otherwise does not provide direct financial benefit. This, however, is changing.

With recent legislation, many facilities are undergoing dramatic changes in reimbursement processes and resultant therapy practices. The first change comes from the Affordable Care Act's sec. 2706 which established Accountable Care Organizations. We will not go into full detail of the ACA or ACOs here, but it reinforced a growing mantra in the therapy communities of moving from "experience based care" to "evidence based care". Experience based care revolves around established therapists knowing how best to treat patients because they have the experience concerning outcomes resulting from their treatment. This type of care is problematic for multiple reasons, but mostly because it is difficult for insurance companies to establish proper care pathways for their patients that lower risk for re-admittance. Evidence based care aims to solve this in two parts. First, care should be quantifiable with outcomes and incremental progress measures. Second, these measures can be used to define ideal

care pathways or ideal outcomes for any given patient. This is a challenging task, given the diversity of patient presentations, but represents a great opportunity for the vHAB system.

The second set of changes comes from the push towards telemedicine. Reimbursement codes for home equipment have existed for a long time, but recent thought-leader shifts may allow for more healthcare technology to replace traditional splints and bands. In this case, vHAB would be prescribed by the therapist and paid for by the patient's insurance. Therapists and physicians can then use another recently added reimbursement codes to view and analyze the data collected at home. Telemedicine legislation is continuously being presented, which may allow for additional opportunities for facilities to see additional benefits for home use of the vHAB system.

Future improvements to the vHAB system will add more benefits to the facilities to enhance system adoption. Data analytics collected during each module are already helpful for defining a patient's care path. Automated documentation and reporting of these analytics, and even usage statistics, could save therapists time and reduce the risk for declined reimbursements. In the short term, we plan on having exportable text blocks that describe the care that was delivered in the 15 minute session. This text can be copied into a patient's documentation, so the therapist does not need to type out anything. Prototypes of this system have already been created, but defining the ideal information to present still needs user testing and feedback. The long-term vision of this automation will directly integrate this documentation into the patient's electronic medical record. This integration presents additional security and workflow challenges, but should further increase the value of the system to facilities.

4.9.C *Benefits of continuity of care*

In an ideal care model, patients will be exposed to the vHAB system in a facility and then be sent home with their own system. This continuity of care allows patients to be immediately familiar with their home exercise program and provides an unbroken record of their performance. Currently, patient progress is only tracked when they return for follow up outpatient visits. Therapists are under the same pressures described above during these follow-up visits and may not perform a robust assessment. This leads to sparse data points of a patient's recovery, making evidence based care decisions nearly impossible. With vHAB, measurements are taken during every therapy exercise and assessments can be performed quickly during follow up visits. Not only does this provide more information for the therapist and motivation for the patient, but it represents a huge shift in tracking recovery statistics across patients. Comparing measurements across subjects can help a therapy facility adjust their practices, or help insurance companies identify ideal care pathways that minimize re-admittance. These data may be of additional use to the scientific community for tracking the differences in spontaneous recovery and therapy induced recovery after stroke, or for introducing new biochemical treatments that work alongside traditional therapy practices where adherence data are paramount in validating the treatment's efficacy.

4.9.D *Conclusions*

The vHAB system represents a new opportunity for therapists and end-users to enhance their therapy practices. vHAB can be used in a therapy facility and then be sent home with patients to continue their rehabilitation, enabling an unprecedented monitoring of recovery statistics and adherence. Early pilot studies assisted in the development of the vHAB system and

further studies will explore the additional benefits that vHAB can provide. End-user surveys

showed that patients enjoyed using the system and provided early evidence that vHAB does

improve adherence to therapy at home through its engaging games and wrapper. This writing

represents the beginning stages of vHAB. It is our hope that through further development and the

commercialization of vHAB that we will be able to establish vHAB as the new de-facto

paradigm for upper extremity therapy after neuromuscular trauma.

4.10 BIBLIOGRAPHY

1.   Duncan, P.W., and S.M. Lai, *Stroke Recovery.* Top.Stroke Rehabil, 1997. **4**(3): p. 51-8.

2.   Richards, L. and P. Pohl, *Therapeutic interventions to improve upper extremity recovery and function.* Clinics in Geriatric Medicine, 1999. **15**(4): p. 819.

3.   van der Lee, J.H., et al., *Forced use of the upper extremity in chronic stroke patients - Results from a single-blind randomized clinical trial.* Stroke, 1999. **30**(11): p. 2369-75.

4.   Hiraoka, K., *Rehabilitation effort to improve upper extremity function in post-stroke patients: A meta-analysis.* Journal of Physical Therapy Science, 2001. **13**(1): p. 5-9.

5.   Brunnstrom, S., *Motor testing procedures in hemiplegia: based on sequential recovery stages.* Phys Ther, 1966. **46**(4): p. 357-75.

6.   Voss, D., M.K. Ionta, and B.J. Myers, *Proprioceptive Neuromuscular Facilitation - Patterns and Techniques.* Rehabilitation Literature, 1985. **46**(9-10): p. 296.

7.   Bobath, B., *Adult Hemiplegia - Evaluation and Treatment.* Rehabilitation Literature, 1979. **40**(5-6): p. 160.

8.   Luke, C., K.J. Dodd, and K. Brock, *Outcomes of the Bobath concept on upper limb recovery following stroke.* Clinical Rehabilitation, 2004. **18**(8): p. 888-98.

9.   Schmidt, R.A., *Motor learning & performance: From principles to practice*. 1991: Human Kinetics Books.

10.   Page, S.J., *Intensity versus task-specificity after stroke: how important is intensity?* Am.J.Phys.Med.Rehabil., 2003. **82**(9): p. 730-2.

11.   Galea, M.P., K. J. Miller, and S. L. Kilbreath. *Early task-related training enhances upper limb function following stroke*. in *Annual Meeting of the Society for Neural Control of Movement*. 2001. Sevilla, Spain.

12.   Winstein, C.J., et al., *A randomized controlled comparison of upper-extremity rehabilitation strategies in acute stroke: A pilot study of immediate and long-term outcomes.* Arch Phys Med Rehabil, 2004. **85**(4): p. 620-8.

13.   Whitall, J., et al., *Bilateral and Unilateral Arm Training Improve Motor Function Through Differing Neuroplastic Mechanisms: A Single-Blinded Randomized Controlled Trial.* Neurorehabilitation and Neural Repair, 2011. **25**(2): p. 118-29.

14.   Taub, E., G. Uswatte, and R. Pidikiti, *Constraint-Induced Movement Therapy: a new family of techniques with broad application to physical rehabilitation--a clinical review.* J Rehabil Res Dev, 1999. **36**(3): p. 237-51.

15.   Hakkennes, S. and J.L. Keating, *Constraint-induced movement therapy following stroke: A systematic review of randomised controlled trials.* Australian Journal of Physiotherapy, 2005. **51**(4): p. 221-31.

16.   Ploughman, M. and D. Corbett, *Can forced-use therapy be clinically applied after stroke? An exploratory randomized controlled trial.* Archives of Physical Medicine and Rehabilitation, 2004. **85**(9): p. 1417-23.

17.   Thieme, H., et al., *Mirror therapy for improving motor function after stroke.* Cochrane Database of Systematic Reviews, 2012(3).

18.   Volpe, B.T., et al., *Robot training enhanced motor outcome in patients with stroke maintained over 3 years.* Neurology, 1999. **53**(8): p. 1874-6.

19.   Lum, P., et al., *Robotic devices for movement therapy after stroke: current status and challenges to clinical acceptance.* Top Stroke Rehabil, 2002. **8**(4): p. 40-53.

20.   Saposnik, G., et al., *Efficacy and safety of non-immersive virtual reality exercising in stroke rehabilitation (EVREST): a randomised, multicentre, single-blind, controlled trial.* Lancet Neurol, 2016.

21.   Haas D., P.S., J. Yu, *Kinect Based Physiotherapy System for Home Use.* Current Directions in Biomedical Engineering, 2015. **1**(1): p. 180-3.

22.   Archambault P.S., et al., *Towards Establishing Clinical Guidelines for an Arm Rehabilitation Virtual Reality System*, in *Replace, Repair, Restore, Relieve – Bridging Clinical and Engineering Solutions in Neurorehabilitation*. 2014. p. 263-70

23.   Shubert TE, B.J., A. Chokshi, M. Barrett, and R. Komatireddy, *Are Virtual Rehabilitation Technologies Feasible Models to Scale an Evidence-Based Fall Prevention Program? A Pilot Study Using the Kinect Camera.* JMIR Rehabil Assist Technol 2015. **2**(2).

24.   Kang X., *Methods for providing telemedicine services* 2014, Rehabtics LLC.

25.   Mousavi Hondori, H. and M. Khademi, *A Review on Technical and Clinical Impact of Microsoft Kinect on Physical Therapy and Rehabilitation.* J Med Eng, 2014. **2014**: p. 846514.

26.   *RespondWell Telerehabilitation Software*. 2016; Available from: http://respondwell.com/.

27.   Kurillo, G., et al., *Tele-MFAsT: Kinect-Based Tele-Medicine Tool for Remote Motion and Function Assessment.* Stud Health Technol Inform, 2014. **196**: p. 215-21.

28.   Cameirao, M.S., et al., *Neurorehabilitation using the virtual reality based Rehabilitation Gaming System: methodology, design, psychometrics, usability and validation.* J Neuroeng Rehabil, 2010. **7**: p. 48.

29.   *Ges Therapy.* 2016; Available from: http://gestherapy.com/.

30.   Friedman, N., et al., *MusicGlove: motivating and quantifying hand movement rehabilitation by using functional grips to play music.* Conf Proc IEEE Eng Med Biol Soc, 2011. **2011**: p. 2359-63.

31.   Song, H. *RAPAEL: Wearable Technology and Serious Game for Rehabilitation.* in *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems.* 2016.

32.   Baker, K., S.J. Cano, and E.D. Playford, *Outcome measurement in stroke: a scale selection strategy.* Stroke, 2011. **42**(6): p. 1787-94.

33.   Mathiowetz, V., et al., *Adult norms for the Box and Block Test of manual dexterity.* Am J Occup Ther, 1985. **39**(6): p. 386-91.

34.   Oxford Grice, K., et al., *Adult norms for a commercially available Nine Hole Peg Test for finger dexterity.* Am J Occup Ther, 2003. **57**(5): p. 570-3.

35.   Bohannon, R.W., *Adequacy of hand-grip dynamometry for characterizing upper limb strength after stroke.* Isokinetics and Exercise Science, 2004. **12**(4): p. 263-5.

36.   Rios, D.C., et al., *NeuroGame Therapy to improve wrist control in children with cerebral palsy: a case series.* Dev Neurorehabil, 2013. **16**(6): p. 398-409.

37.   Steele, K.M., A. Rozumalski, and M.H. Schwartz, *Muscle synergies and complexity of neuromuscular control during gait in cerebral palsy.* Dev Med Child Neurol, 2015. **57**(12): p. 1176-82.

38.   Weichert, F., et al., *Analysis of the accuracy and robustness of the leap motion controller.* Sensors (Basel), 2013. **13**(5): p. 6380-93.

39.   LeapMotion. *Leap Motion SDK and Plugin Documentation.* 2016; Available from: https://developer.leapmotion.com/documentation/.

40.   Stuart, S., *Graph Maker for Unity®.* 2016.

41.   Imphenzia, *Universal Sound FX for Unity®.* 2016.

42. Beaton, D.E., et al., *Development of the QuickDASH: comparison of three item-reduction approaches.* J Bone Joint Surg Am, 2005. **87**(5): p. 1038-46.

43. MacDermid, J.C., et al., *Patient rating of wrist pain and disability: a reliable and valid measurement tool.* J Orthop Trauma, 1998. **12**(8): p. 577-86.

44. O'Dwyer, N.J., L. Ada, and P.D. Neilson, *Spasticity and muscle contracture following stroke.* Brain, 1996. **119 (Pt 5)**: p. 1737-49.

45. Lee, D.D. and H.S. Seung, *Learning the parts of objects by non-negative matrix factorization.* Nature, 1999. **401**(6755): p. 788-91.

46. Roh, J., W.Z. Rymer, and R.F. Beer, *Robustness of muscle synergies underlying three-dimensional force generation at the hand in healthy humans.* J Neurophysiol, 2012. **107**(8): p. 2123-42.

47. Roh, J., et al., *Alterations in upper limb muscle synergy structure in chronic stroke survivors.* J Neurophysiol, 2013. **109**(3): p. 768-81.

48. Delp, S.L.e.a., *Maximumisometric moments generated by the wrist muscles in flexion-extension and radial-ulnar deviation.* Journal of Biomechanics. **29**(10): p. 1371 - 1375.

49. Hall, M.J., S. Levant, and C.J. DeFrances, *Hospitalization for stroke in U.S. hospitals, 1989-2009.* NCHS Data Brief, 2012(95): p. 1-8.

50. Shaughnessy, M., B.M. Resnick, and R.F. Macko, *Testing a model of post-stroke exercise behavior.* Rehabil Nurs, 2006. **31**(1): p. 15-21.

51. C, G., *Issues of parent compliance: what the clinician and researcher should know.* Physical and Occupational Therapy Pediatric 1991. **11**: p. 73-77.

52. Touillet, A., et al., *Assessment of compliance with prescribed activity by hemiplegic stroke patients after an exercise programme and physical activity education.* Ann Phys Rehabil Med, 2010. **53**(4): p. 250-7, 257-65.

Chapter 5. ADDENDUM: PREFRONTAL CORTEX

5.1   INTRODUCTION

Control signals for traditional BMI systems are usually recorded from movement related areas of the cortex, such as the primary motor cortex (M1), premotor cortex (PMC), and parietal cortex (PC). In primate studies, these neural signals are often correlated with overt motor behaviors prior to BMI integration. In this way, researchers have predetermined inclinations that the animal is able to volitionally control an end effector, and can simply replace the BMI for the original motor behavior. In human applications of BMI, however, there is no way to determine a pre-injury cortex-movement map. Successful BMI systems have been implemented in human subjects that place electrodes over anatomically determined motor areas, in which either motor imagery or complex training paradigms are utilized to achieve control. Motor imagery has been described as exhausting, and complex training paradigms may take too long to successfully teach a patient to use a BMI system. Further, patients suffering from a stroke or TBI may not have neural signals available in any of the traditional motor BMI areas due to tissue damage. As an alternative to these traditional areas, we propose the use of the prefrontal cortex for BMI systems.

The prefrontal cortex is reported to be the center of executive control of motor behavior [1]. Single-neuron activity in the dorsolateral prefrontal cortex (dlPFC) has been shown to modulate with motor planning [2, 3] and the integration of sensory information [4]. A recent study showed for the first time that neurons throughout the lateral prefrontal cortex can be operantly conditioned to varying extents[5]. These results encourage further examination given the potential applications of dlPFC controlled brain computer interfaces. These neurons may

prove easy to adapt to BCI control without the need for initial motor calibration or training, since their activity is related to the idea of an overt action instead of directly driving the action. In this work, we attempt to train monkeys to volitionally modulate the activity of single neurons and local field potentials in the dlPFC using a variety of recording and training techniques.

5.2   METHODS

5.2.A   *Single Neuron Conditioning*

Two monkeys (V and T) were implanted with arrays of 12 microwire electrodes [6] over the dorsolateral prefrontal cortex (Figure 5.1). Wires were advanced individually to find neurons. The monkeys were restrained in a primate chair in a booth and neurons were discriminated using a dual time-amplitude window discriminator. Increased firing rates caused a cursor on a screen in front of the monkey to rise towards a target box. When the firing rate was sufficiently high, such that the cursor was in the target box, applesauce was delivered to the monkey. Experiments were divided into epochs where food reward was available (R), indicated by cursor presence and an auditory cue, and epochs where food was unavailable (NR), indicated by lack of visual and auditory cues.

5.2.B   *Local Field Potential Conditioning*

Two monkeys (V and X) were bilaterally implanted with "dual" electrodes over motor, premotor, and prefrontal cortex (Figure 5.1). Monkey V was re-implanted 8 months after the previous wire implant had failed. Dual electrodes allow for simultaneous recording of intracortical and epidural local field potentials and are less surgically invasive than arrays of intracortical microwire electrodes. Differential local field potentials were recorded between the

subdural and epidural sites on each dual electrode. These signals were used as the main control

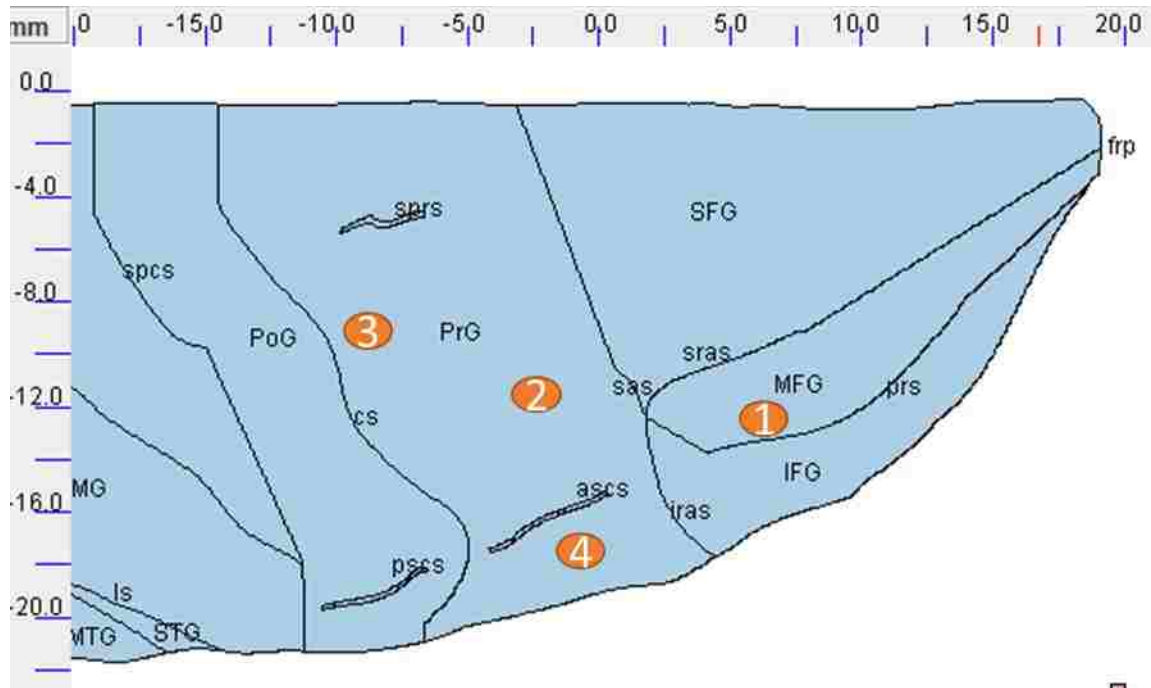signal for all conditioning paradigms described below.

Figure 5.1 Approximate electrode implant positions for prefrontal cortex conditioning. Electrode

areas 1-4 are overlaid on a sample Macaca fascicularis atlas provided at http://braininfo.org. Positioning

has not been validated post-mortem. Monkey V and T received one microwire array over area 1

(dorsolateral prefrontal cortex). Monkey V was re-implanted with dual electrodes over areas 1, 2 (dorsal

premotor), 3 (motor) and 4 (ventral premotor) on the left hemisphere and 1, 2 and 3 on the right

hemisphere. Monkey X was implanted bilaterally over areas 1, 2, and 3.

Both monkeys were trained on a one dimensional center-out task where wrist torque

controlled a cursor and the monkey needed to move the cursor left or right to hit a target box.

Food reward was delivered when the monkey returned to center after hitting a target. The

baseline signal of all electrodes were recorded and categorized during periods of rest and the

center-out task. These baseline signals were used to set target thresholds for an operant conditioning task. In the main conditioning tasks, multiple control paradigms were tested to enhance the monkey's ability to volitionally control local field potential within the prefrontal cortex. In each case, the monkey's neural activity would move a cursor up or down in a similar paradigm to the center out-task. Hold times, thresholds, and power calculation methods were modified throughout the course of the experiments. Initial learning experiments were performed exclusively with 1:1 mapping of reward to trial completion. Eventually, to provide evidence of volitional control, we implemented an R:NR epoch schedule at 2 minutes per epoch.

Beta Power (Monkey V and X)

Power modulations in the beta frequency band (13-33Hz) were recorded from an electrode over prefrontal cortex and was used to control the position of the computer cursor up and down. Power was calculated by summing the filtered and rectified signal and directly mapped to cursor position. This mapping was adjusted with a multiplier that allowed the monkey to achieve targets early on, but would ideally decrease over time.

Bi-Directional Beta Power (Monkey V)

Early results with Monkey V showed strong correlations between motor cortex activity and prefrontal cortex activity during task performance. Further examination of the torque traces revealed that the monkey was moving his wrist to control the task. With continued experimentation this factor may have dropped out on its own. In order to expedite this dissociation, we modified the task to control the cursor with differential power such that a zero value was where motor cortex beta power and prefrontal cortex beta power were equal, and

deviations in relative activity would drive the cursor up or down. Implant failure prevented full exploration of this paradigm. Retrospectively, this correlation may have been desired for proving the viability in BMI applications. It would have been a more validating approach to watch the correlations over time, but move towards an R:NR experimental paradigm.

Gamma Power (Monkey X)

This method was the same as the Beta Power methods, but utilized LFP signal power from 70-110 Hz. Additionally, since the bursts of the signal were shorter, we implemented a smoothing function (400ms trailing average of RMS value) to assist in cursor control.

5.3   RESULTS

Volitional control over prefrontal cortex was not significantly demonstrated in these experiments. Our primary measure of volition compared task completion rates and signal composition during R and NR periods. If R rates were significantly higher than NR rates it signified that the control signals in R were sufficiently different from baseline (random) modulations of cortical signals. For the single neurons, we were not able to retain any given neuron for longer than 1 session and the implants both failed within 3 months of implantation after 25 sessions with Monkey V and 10 sessions with Monkey T. Monkey V was re-implanted, but would not work well in the booth due to discomfort caused by a posthumously diagnosed MRSA infection in his margin. This limited us to only 20 total sessions of limited and decreasing performance. Monkey X provided better results overall. Monkey X received rewards for modulating beta power, but failed to show differences between R and NR epochs. Gamma power control was more promising, but equipment failure throughout the course of experiments made

243

day to day performance or learning comparisons impossible. Further, this equipment failure often took the form of passing incorrect control signals to the cursor, which made learning a set task difficult. Additional work is being performed that solves these issues by additional members of the Fetz lab, (Bioengineering student Camille Birch). In the sections below we present a few interesting results from some of the "good" experiments. These results should be viewed only as anecdotal, but should spark interest in the potential of using the prefrontal cortex as an alternative source for BMI control signals.

5.3.A *Single Neuron*

Both monkey T and V experienced implant failure early in the experimental process. Initial results indicate that volitional control over neurons in the prefrontal cortex is possible. As indicated in Table 1, of the 33 conditioning sessions across both monkeys, seven showed firing rate differences between R and NR epochs. One experiment in particular showed a 258% increase above baseline threshold (Figure 5.2). Further examination of this neuron was impossible since it was not present during the next day of testing. Many of the failed attempts were due to noisy signals or the loss of neurons within experiment session.

Table 5.1 Prefrontal cortex single neuron conditioning. Of 33 total sessions across 2 monkeys, only 7 sessions showed evidence of volitional control. These 7 successes used 6 unique neurons of the 19 total neurons that were conditioned.
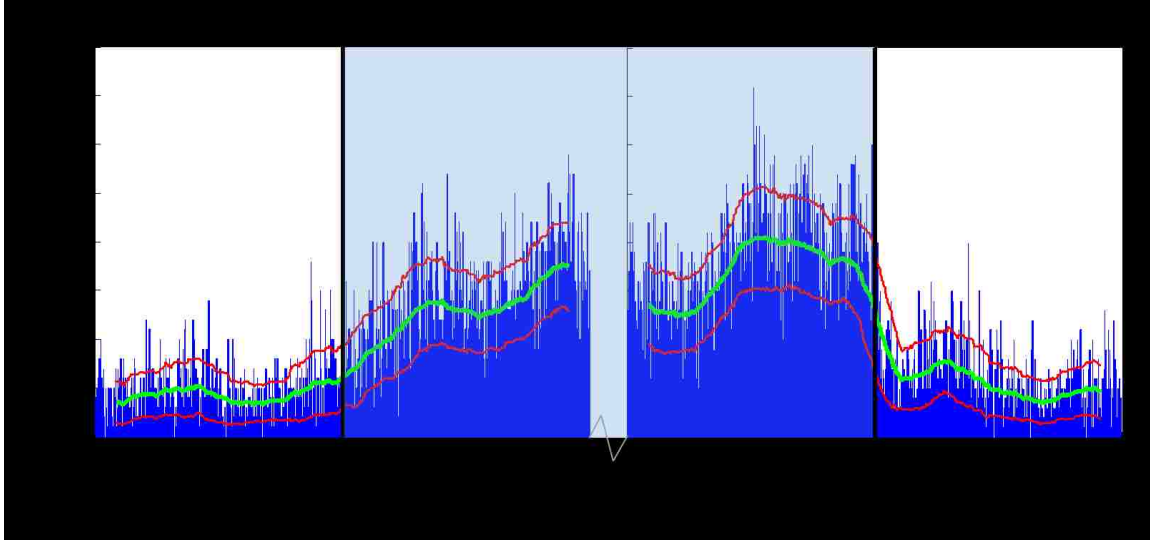
Figure 5.2. Single Neuron Conditioning in Prefrontal Cortex. In this sample experiment we demonstrate increases in single neuron firing rate at NR-R transitions.

5.3.B *Local Field Potential Beta Power Conditioning*

Monkey V was able to perform the beta control task in a 1 trial to 1 reward ratio (all R), but we were not able to reach the R:NR stage of experimentation due to a detour towards a dissociation study as mentioned above. In many of the experiments, power across all of the recording electrodes showed correlations in beta power (Figure 5.3 A, B). This was even present in the contralateral hemisphere in some sessions, though to a lesser extent (Figure 5.3 C). To further examine this phenomenon we looked at the phase and amplitude relationships of the signals by plotting each signal triggered off of the peak of an oscillatory event in the control electrode (Figure 5.3 D). The relationships between amplitude and phase were not consistent across sessions, but may warrant further examination.

Monkey X had similar power increases across channels and was also able to receive reward during the tasks using electrode L10 during the training period. This time was also used to fine-tune some of the task parameters, such as hold time, gain, and target size. When moving to the R:NR paradigm however, we were unable to establish reliable evidence of volitional control (Figure 5.4). We tested two different control sites within the prefrontal cortex over 3 months in the R:NR paradigm. The epidural electrode at L10 eventually lost recording fidelity. This may have led to apparent increases in trials/minute statistics over time as increases in signal power were no longer differential. We also experienced a 30 day loss (day 45-75) in recording ability due to a MRSA infection with Monkey X and resulting treatment and quarantine. When switching to L8 we experienced our first hardware failure with the Grapevine recording system, where a broken cable was used to connect the recording equipment with the task equipment. This calls into question much of the data presented in Figure 5.4.
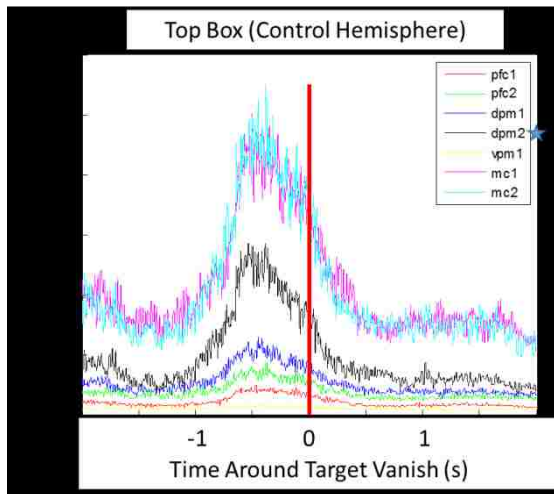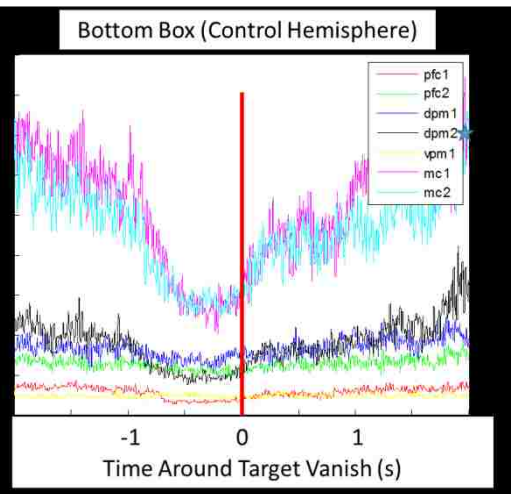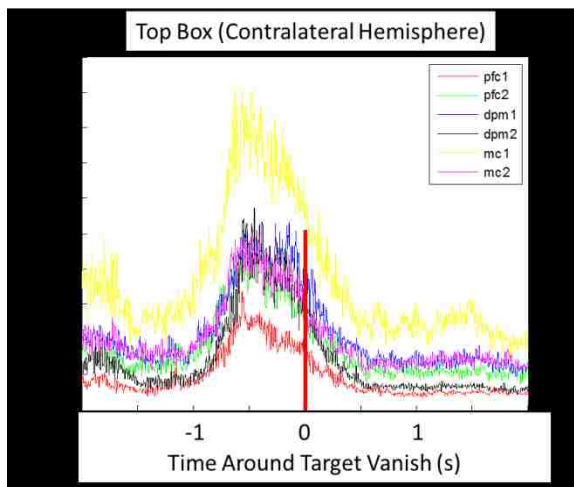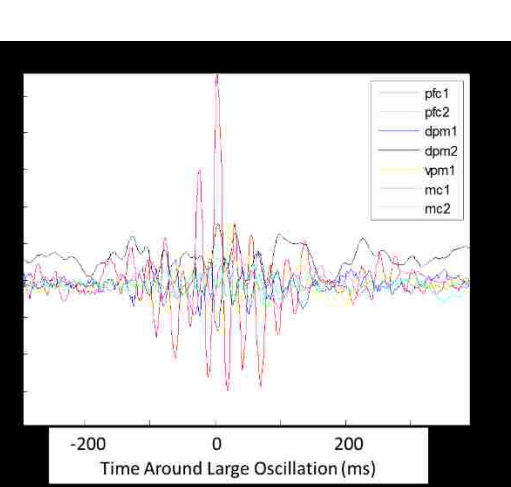
Figure 5.3. Beta Power correlations during PFC LFP Conditioning. In all sessions, pfc = Prefrontal

Cortex, dpm = dorsal premotor cortex, vpm = ventral premotor cortex, mc = motor cortex.

In the data presented in previous chapters we saw R:NR ratios larger than 50:1 for single

neurons (Chapter 2) and up to 3:1 for motor field potentials with only 14 total sessions and less

control over fine tuning thresholds and task parameters (Chapter 3). Comparatively, successful performance in PFC beta conditioning was not obtained. There are many possible explanations for this difference. First, this could have been due to limitations in our task design and setup. Differences in processing speeds between data acquisition units required that recording signals be decimated before the power values could be calculated. Further, the screen refresh rate was variable during the task and data packets were of varying length, due to limitations on the computer's processing speed and the data acquisition unit sampling rate. Another hardware related explanation comes from task confusion from all of the false task parameters from equipment malfunction. It would be difficult for the monkey to learn the task if the control signals are not consistently mapped to task performance.

It could be that a 1.8:1 ratio between R:NR events is as good a performance as we could physiologically achieve. If this is the case then beta power PFC control of a BMI system would not be very robust to false positives. With failing electrodes and hardware malfunctions it is difficult to conclude one way or another. It is also possible, that beta power in prefrontal cortex has too much spontaneous high power oscillation, and our task algorithms were not finely tuned enough to pick up volitional vs non-volitional differences between epochs. Finally, and equally likely, is that during the NR periods, the monkey would often struggle in his chair, leading to increased movement artifacts and potential behavior confounds in the power signal. Future work with robust equipment, carefully controlled algorithms, and a calm monkey could examine these issues further.
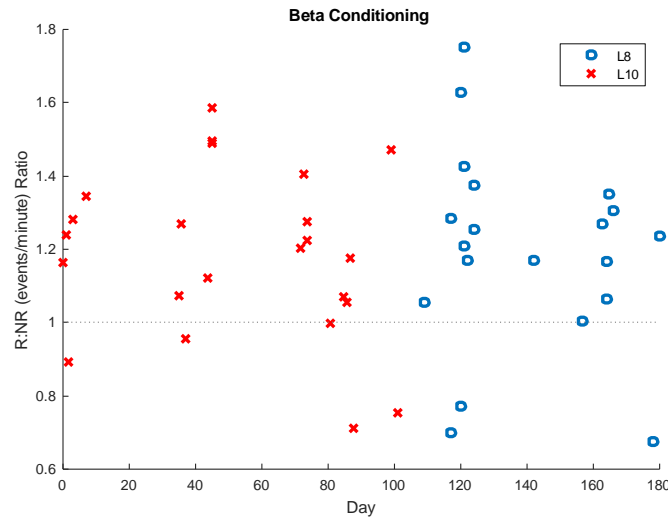
Figure 5.4. Beta Power LFP Conditioning task performance. Ratios were calculated by first calculating the average performance in R epochs (events/minute) and then in NR epochs. Presented values are the ratio between R and NR epochs (R/NR).

5.3.C *Local Field Potential Gamma Power Conditioning*

Gamma field potential conditioning suffered from similar struggles as the beta conditioning. Task completion in an all-R paradigm was achieved quickly (potentially spontaneously), but R:NR comparisons proved slightly more difficult. Initially, we experienced promising performance increases (Figure 5.5), even as we fine-tuned the task parameters to account for movement artifacts and increased hold time to remove false positives. However, during this training period, we experienced 2 additional hardware failures – one from a broken ground and the other from a bad connection between the recording equipment and task controller. These problems were found weeks after they could have occurred, but we believe it happened after day 60 when equipment was temporarily moved out of the lab. This again calls into question the validity of Figure 5.5, but could explain the strange performance past day 60.

Task performance varied in both R and NR periods, but was rarely larger than 5 rewards/minute in any given epoch.
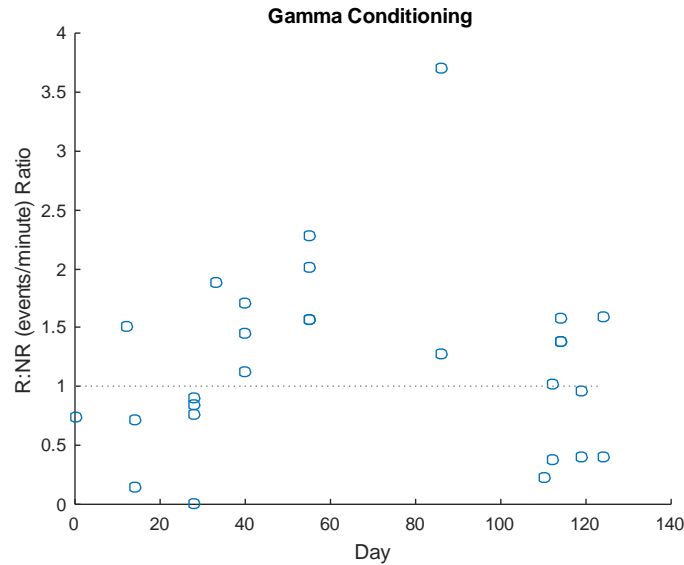

Figure 5.5. Gamma Power LFP Conditioning task performance. Ratios were calculated by first calculating the average performance in R epochs (events/minute) and then in NR epochs. Presented values are the ratio between R and NR epochs (R/NR).

5.4   DIRECTIONS FOR FUTURE STUDY

5.4.A.1   *General Recommendations*

One of the key limitations in the above experiments, aside from hardware errors, is the infrequency of task performance. Training occurred more than shown in figures 5.4, 5.5 as in addition to R:NR experiments we tested a similar bidirectional control paradigm and did some R only training sessions. However, this was not necessarily on a relevant learning timeline for BMI control. It would be ideal to set up a static schedule for the experiments, such as training with R only paradigms every day, but R:NR every other day. This would likely enhance the monkey's

ability to learn the task quickly and could lead to better results. This was difficult in our case due to troubleshooting equipment failures, and Monkey X's grand ability for removing his cap and destroying connectors. It would also be interesting to test longer R, NR epochs during early trainings. In Chapter 3 we utilized 5:10 minute epoch ratios for learning the task which seemed to aid in performance.

5.4.A.2   *Proving Volitional Control*

Many BMI papers look at reaction times and shuffled trial control analysis to prove control over the task, but the tasks themselves tend to be more complex such as 8 target center out tasks with 2 directions of control. In our experiments, our best case was to see high R:NR ratios, but we also wanted to see nearly 0 NR events. Zero NR events would indicate a complete lack of false positives which would be ideal in a real BMI implementation. Increasing task difficulty, by adding additional targets or control signals could decrease this false positive rate, but would be inherently more difficult to learn. It would also be beneficial to design a study comparing learning rates between PFC control and Motor cortex control. However interpretation is challenging since there may be strong correlations between these signals; emulating the exact control parameters and experimental paradigms as existing BMI literature could be a better place to start.

5.4.A.3   *Cross Cortex Correlation*

The electrodes implanted for this study allow for the examination of signals across multiple cortical areas. By examining the correlations between these electrodes both in amplitude and phase could provide interesting insights into how a PFC BMI may be controlled. It could be possible to validate some initial findings of top-down control from the PFC. Further, it could

lend evidence towards the use of PFC as an alternative, easy to learn BMI control paradigm. The

data collected thus far is ready for this analysis, but not within the scope of these studies.

5.5   BIBLIOGRAPHY

1.   Tanji, J. and E. Hoshi, *Role of the lateral prefrontal cortex in executive behavioral control.* Physiol Rev, 2008. **88**(1): p. 37-57.

2.   Fuster, J.M., *Behavioral electrophysiology of the prefrontal cortex of the primate.* Prog Brain Res, 1990. **85**: p. 313-23; discussion 323-4.

3.   Shima, K., et al., *Categorization of behavioural sequences in the prefrontal cortex.* Nature, 2007. **445**(7125): p. 315-8.

4.   Ono, T., et al., *Single neuron activity in dorsolateral prefrontal cortex of monkey during operant behavior sustained by food reward.* Brain Res, 1984. **311**(2): p. 323-32.

5.   Kobayashi, S., W. Schultz, and M. Sakagami, *Operant conditioning of primate prefrontal neurons.* J Neurophysiol, 2010. **103**(4): p. 1843-55.

6.   Jackson, A. and E.E. Fetz, *Compact movable microwire array for long-term chronic unit recording in cerebral cortex of primates.* Journal of Neurophysiology, 2007. **98**(5): p. 3109-18.

Chapter 6. FINAL THOUGHTS

In the preceding chapters, we've presented multiple approaches towards new methodologies for exploring BMI systems. In addition to showing the efficacy of brain stimulation reward in Chapter 2, we provided evidence that false positives are more likely to occur in an unconstrained environment than traditional constrained environments, indicated by higher event counts during NR epochs in in-cage experiments. In work described in Chapter 3, we built and tested systems to translate traditional experiments to a free-behavior environment with the inclusion of a cage-mounted feeder and a behavior monitoring system. These systems provide a foundation for expansion into untethered paradigms, while showing that such a system can be used for BMI experiments. In Chapter 5, we presented preliminary findings suggesting that the prefrontal cortex can be used in BMI systems. While this paradigm needs further exploration, it represents an exciting alternative to motor-cortex-dependent BMIs. Taken together, these three chapters represent my challenge to the BMI community to rethink traditional approaches to BMI systems.

The vHAB system and its subsequent commercialization have exposed me to the challenges of technology adoption for patients with neuromuscular trauma. Factors such as market sustainability, development funding, usability and training are not only crucially important to reaching patients, but are often overlooked in the BMI field. Further, the inclusion of product cost, public perception, and end-user needs are rarely considered at early stages of research, but are fundamental to a product's success. Applying these development principles to the BMI field elucidate additional challenges.

Low mobility or locked-in patients may be a great first target for BMI technology, but for it to be sustainable in the market, fundable in research, and impactful for large populations, we must develop systems that could assist patients with all types of neuromuscular trauma. This means working directly with end-users throughout all stages of the research and design process to understand their needs and the challenges with using a BMI system in everyday life.

Much of this work is made possible by advances in technology that have allowed for smaller recording systems such as the Neurochip, and low-cost commercial sensors like the Kinect. This new technology does not originate from the neuroscience field, as is often overlooked by its members when designing new experiments. Staying at the front of all technology fields, especially new commercial technologies, may provide easy access to tools for researchers. It provides inspiration for new BMI systems that may have tremendous impact on the end-user. For example, combining a binary BMI control signal with eye tracking and a context sensitive smart home could easily allow patients to navigate their homes and interact with their environment. Designing these transdisciplinary systems will require extensive work, but may provide the most effective, sustainable, and cost effective solutions for patients with neuromuscular trauma.