Multidisciplinary Senior Design Conference
Kate Gleason College of Engineering
Rochester Institute of Technology
Rochester, New York 14623

Project Number: P19002

Gamified Rehabilitation for Independent Practice G.R.I.P. Rhythm Game

Anup Jonchhe
Biomedical Engineering

Forrest Shooster
Biomedical Engineering,
Game Design and Development

Ken Nepomuceno
Biomedical Engineering

Dominic Arcoraci
Computer Engineering

William Bates
Computer Engineering

Abstract

After implantation of a hand-based prosthetic, transradial amputees require practice to learn how to effectively use their new device. GRIP (Gamified Rehabilitation for Independent Practice) is a novel rhythm game produced with the goal of providing amputee patients an engaging and challenging means of practice for learning movements with their new prosthetic at home. The game takes design influence from past successful rhythm games, with tracks for each degree of freedom that contain notes to a corresponding song. Similar to these games, notes must be hit to improve score and measure performance. To better serve the games purpose, GRIP introduces a novel slider bar system that enables notes to be placed vertically on each degree of freedom track, dictating the position on the prosthetic required to hit a given note. Performance data including score, practice time, and raw positional data are recorded to allow for analysis of user performance using metrics like stability, percent overshoot, and lag time. Testing made possible by the development of a glove-like controller that simulates input from a prosthetic user. The game is designed to run on standard desktop PCs, and will be able to eventually interface directly with the DEKA LUKE arm.

Introduction

There are approximately 1.9 million amputees in the United States. Based on a 1:4 ratio of upper to lower extremity prostheses, there are an estimated 46,000 upper extremity amputations per year in the United States due to a variety of causes including physical injuries, congenital defects, and illnesses such as cancer or diabetes. About 14%, according to the Amputee Coalition, have upper extremity amputations below the elbow and may benefit from a transradial prosthesis [3]. The patients discussed have already been subjected to high costs of medical care for their associated conditions but their prostheses can carry significant additional costs [4]. Once these individuals have their prostheses, many of them abandon them due to difficulty of control, lack of motivation to practice, and difficulty to access practice tools [5]. Many projects being worked on today to address this utilize virtual reality (VR) including the one utilized by Case Western Reserve University (CWRU) and the Functional Electrical Stimulation (FES) lab. Setting up a personal computer (PC) with VR capability requires some degree of additional cost and a typical VR-ready PC potentially costs significantly more as compared to typical personal computers. This significant cost makes it quite unlikely that the average consumer would have the necessary materials on hand to use this tool so many of these tools are primarily confined to offices and research labs. Hence, it has become necessary to improve the continuum of care for these robotic prosthesis users by providing a practice tool that they can use at their own leisure from home (or even on the go). By providing a game practice environment with lower PC specification requirements, we increase access to treatment and improve the patient's ability to utilize regular practice to more quickly learn to use their prosthesis at a rate they are comfortable with.

Design Process

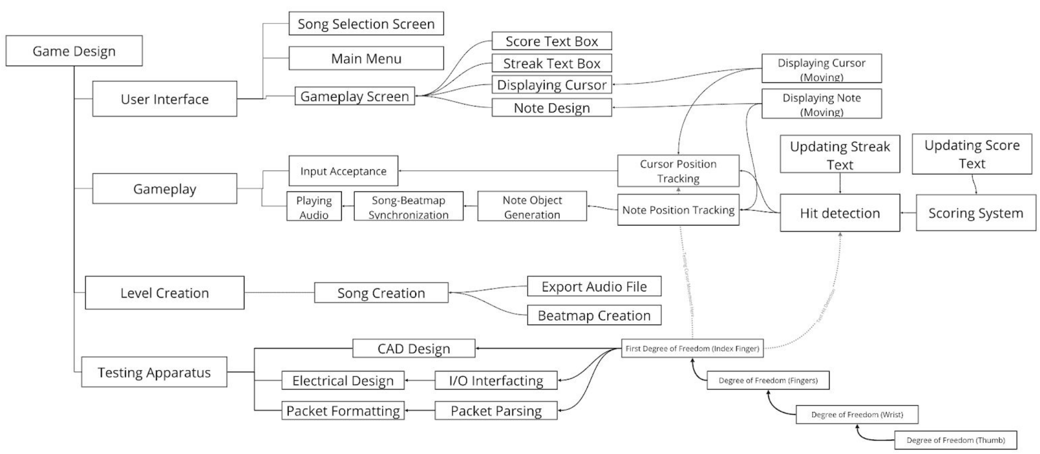

Figure 1. High Level Functional Diagram

Since there were two main components to this project, this section will be divided into two sections: one will focus on the gameplay design itself, and the second will focus on constructing the testing device meant to aid in the use and development of the game. Figure 1 shows the multiple aspects of the project, as well as interdependencies between them.

Game Design

GRIP (Gamified Rehabilitation for Independent Practice) is a video game-based training system made to be an at-home exercise to aid transradial amputees in learning to use a dexterous, high degree of freedom prosthetic hand. To design such a training system, decisions were made with careful consideration of the requirements of the project. The game needed to be able to interface directly with the myoelectric implant used by the Functional Electrical Stimulation lab so that user motor intent is the primary controller of the game. Since the system is meant to be sent with patients for at-home practice, it must run on commonplace hardware such as a mid-level desktop PC or laptop. We investigated game genres that would be able to utilize and incorporate various degrees of freedom (DoF) such that all types of motion could be practiced. GRIP was designed as a rhythm game because of the potential benefits specified in literature of this type of rehabilitative game, taking some design influence from past successful rhythm games, such as Stepmania and Guitar Hero [1]. Finally, the game had to provide a spectrum of difficulty levels to remain challenging and valuable to the patient. GRIP has an interface which allows developers to use a serial port connected glove arm to create levels and gestures for song maps and the difficulty can be adjusted by changing the note and hit bar sizes. Slider bar motion as a “hit target” has never been used in such games and those games generally require some sort of button press to signify note hits. GRIP instead detects the positioning of the slider during the note timespan to identify whether the slider is contained in the note’s range. To make development of our game swift and reduce additional development time, we chose to use the Unity 3D game engine which has

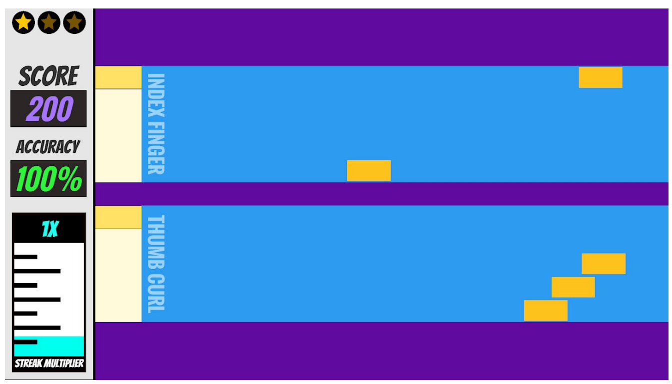

Figure 2: Main Gameplay Scene

With these considerations, it was decided that a rhythm game model would provide us the flexibility needed to train multiple degrees of freedom, provide enough controllable parameters to adjust to increase or decrease difficulty, its use of repetition, and easy mappability to step response characterization. Varying numbers of “tracks”, each one corresponding to an individual degree of motion, appear on the screen based on the number selected on the DoF selection screen. The size of the notes, tracks, and note “hit” regions are all adjustable and will function equivalently at varying scales. As notes are hit a “streak” builds up and when notes are missed it returns to “1X”.

Songs were created as a combination of music and note positions which are stored as “beatmaps”. Notes travel down each track and the user is tasked with hitting the note with a cursor that can be controlled the same way one would move the DEKA LUKE arm. Notes have a vertical placement in the track that approximately aligns with the different instruments of the music. As music plays, the incoming notes move horizontally to the left along the track, giving enough time for the user to move the cursor to the correct location. During gameplay, as notes are “hit”, score and accuracy are updated and displayed to the left.

User Interface

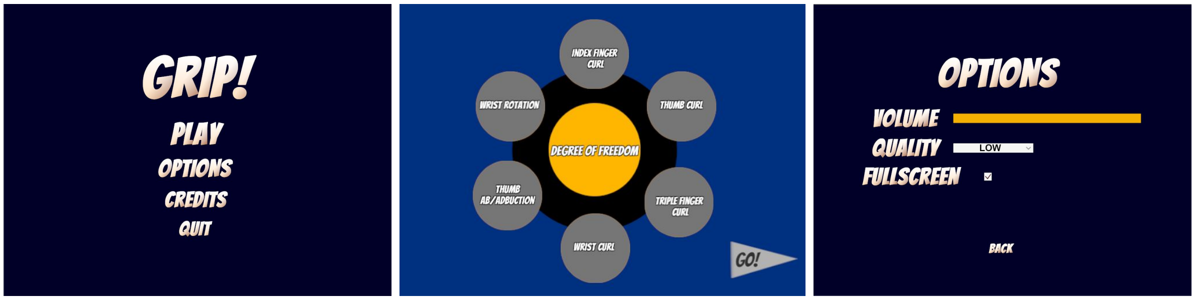

Figure 3: From left to right - “Main Menu” screen, “DoF Selection” screen, “Options” menu

The development of a few user interfaces needed to be designed in order to create the basic skeleton of the game. The game would initiate at the “Main Menu” screen, which would include four buttons with distinct functions: an "Options" button directing the player to the “Options” menu, a "Credits" button leading to the “Credits” screen, a "Quit" button to exit the game, and a "Play" button that would navigate the user to the “Song Selection” screen. From this screen, the user can select from a list of various songs with their associated beatmap. Once the user has chosen the song they desire to play, they will then be brought to the “DoF Selection” screen to determine the gameplay level of difficulty based on the amount of movements/degrees of freedom selected. The user will be given the following selection options: Wrist Roll, Wrist Flexion/Extension, Thumb Flexion/Extension, Middle/Ring/Pinky Finger Flexion/Extension, Index Flexion/Extension, and Thumb Ad/Abduction. Finally, once the player has finished playing the song, they will be brought to a Results screen that will not only show the player how they performed in the song, but also give them an idea on what they need to improve on. The interface was also designed to be colorblind friendly using variation in hue and brightness to produce effective contrast [2].

Difficulty

Difficulty was a primary consideration in the development of the game as, to remain engaging and valuable, the difficulty of the game must scale with the user’s improvement. As a new user, playing a game that is overly difficult can be discouraging. Similarly, an experienced user must be constantly presented with new challenges to prevent boredom. Failing to meet these needs can lead to infrequent use of the game, hindering a patient’s improvement. Rhythm games provide many adjustable properties that can increase the difficulty of a song.

One mechanism for scaling the difficulty with user improvement starts at the song selection screen. Song design can heavily impact the type of practice the user receives. Note density (which can vary over a song), song tempo, total range of motion, and jumps in note position are all factors that can be adjusted in the design of a beatmap to modulate the difficulty of the song. Another dimension of difficulty is provided by the degree of freedom selection. In an effort to give the patient and physicians the most control, it was decided that users would be able to select any combination of DoFs to include in a session. To start, users could focus on each DoF individually. After time and improvement, users could then practice combinations of DoFs that more closely resemble real-world movements and applications. However, this requires intelligent design of the song beatmaps. Beatmap designers must be cognizant of the combinations of DoFs such that the notes do not become overwhelming with the inclusion of more, yet they should not be unengaging when practicing them individually.

Performance

In addition to the requirements of gameplay, it is necessary for physicians and researchers to be able to track the performance of the user as they practice and improve. Metrics such as practice time, DoF stability and percent overshoot can be useful to describe the abilities and improvement of the patient. Rather than calculate a finite number of metrics within the game, GRIP was designed to be able to export the raw positional data of each degree of freedom, in addition to the score, accuracy, and song metadata, after a game session is completed. This data will include the time, desired position and actual position of each of the degrees of freedom. In this case, desired position refers to any notes that appear in the beatmap.

BeatMaps

Beatmaps are a critical component to the gameplay and exercise aspect of the training system. They must be intelligently designed to simultaneously keep users engaged while also being manageable at the intended skill level. Considerations of note-audio synchronization, overlapping degrees of freedom, and overall difficulty are important in ensuring a new addition to the game provides value to the users. Included in each beatmap is all information necessary for the generation and placement of note objects along the selected DoF tracks. Beatmaps are saved using the JavaScript Object Notation (JSON) file format, a text format able to transmit data objects consisting of attribute–value pairs and array data types. Each file contains metadata for its corresponding song and an array of note objects, described as note times, vertical placements and other attributes. These files are then read into the game and translated into Note objects which are loaded before gameplay begins. This human-readable file format and beatmap structure make the addition of new songs a simple task.

To more conveniently and accurately develop beatmaps, a tool was developed that could be run in the Unity Game Editor and would output a beatmap file after a play session. The BeatMapMaker allows the level creation team to play through a song and use the arrow keys and spacebar to place notes along a single degree of freedom. The output file contains the note times and vertical placements of each note in the beatmap JSON file format, which can be combined to produce a complete and playable song with multiple DoFs.

Testing apparatus

The testing apparatus was designed with the focus of being able to be used to test the game. Other devices previously designed are gloves that a user can wear. These keep everything in place however they might be bulky and if a user sweats in the glove it would not be pleasant for others to use or someone to use for a long period of time.

The device had be able to be able to be strapped on easily with no extra cameras or sensors outside of the wearable device.This restricted the size of the components and possible microcontrollers used. The system had three main parts. The first was the Teensy 3.2 microcontroller in a 3D printed mount [6], the second an accelerometer mounted [7] to the back of the hand, and the third was flex sensors for the finger positions. The three of these components allowed for the hand to get five of the six possible movements.

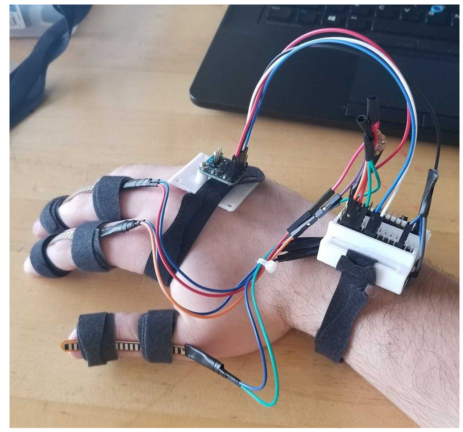

Figure 5: Testing Apparatus

Since the device would need many analog to digital ports a microcontroller with many of them was required. The teensy device has 10 analog inputs. This was sufficient since our design only has 4 flex sensors and an accelerometer, each taking an input. There are more pins available for addition of more flex sensors use to add more fingers, or a sensor for wrist flexion, extension. The teensy 3.2 also has I^2C capability. This was perfect for the addition of other peripherals such as the accelerometer. This design would hold the device in place while being adjustable for all people to use.

The next part was the accelerometer on the wrist which, as previously mentioned, interfaced with the Teensy 3.2 via I^2C communication. When the 3-axis accelerometer is held still, it records the acceleration due to gravity. This makes it an ideal sensor for fine wrist movements and measuring the rotation of the wrist. A 3D printed holder with two screws was used to mount the accelerometer in place which was then held down by velcro straps to keep it firmly in place.

The third part was the flex sensors. For this several 2.2 inch sensors were used. These were selected since they change resistance as they flex. Depending on the resistance of the flex sensors the angle of the fingers could be calculated allowing for the exact position to be computed. By placing these on the thumb, pointer and the middle finger it allows for the flexion and extension to be recorded for three degrees of freedom. Velcro straps were used to secure the flex sensors in place since they're adjustable and finger size varies from person to person. A 4.5 inch flex sensor was used to measure wrist extension and flexion since a 2.2 inch flex sensor could not pick up the full range of motion of the wrist due to its smaller size. This flex sensor was secured to the wrist using velcro straps allowing for the extension and flexion angle of the wrist to be calculated.

Testing

The game was tested with the use of the testing apparatus. This allows for us to control the game by mimicking the behavior of the Hub device. This allowed for testing without the real hub or a prosthetic hand. Without this device it would be hard to test the gameplay. The testing apparatus allowed for anyone to play the game. The first stage of testing was to connect the hand to the game and control some basic sliders. The sliders would move as the fingers flexed. This showed not only proof of concept but also how the testing device and the game can connect and interact together. Next the testing apparatus was scaled up with more sensors for more degrees of freedom. At this step the fingers were then matched to real sliders for actual game play. At this point the finger would slide the real slider and be able to move it to where notes would be.

Conclusion

GRIP provides multiple benefits to the customer, the Functional Electrical Stimulation lab and Case Western Reserve University, and users, such as amputee patients. Modeled after successful rhythm games such as Guitar Hero and StepMania, the game serves as a platform that can provide users with an engaging experience that is capable of exercising a complex variety of motions and reporting performance of those motions to researchers. Unlike other rhythm games, GRIP replaces the action of hitting a button with movement of a slider. by using myoelectric control to play the game. The program was designed to run on a standard desktop PC enabling consistent practice in the user's home. Performance data such as DoF position can be exported and analyzed for metrics such as stability, overshoot, and lag time. This performance data, coupled with the scoring system and selection of songs of varying difficulty, helps researchers, doctors, and patients recognize patient improvement and encourages further learning appropriate for their current performance. Through the design of beatmaps, patients can be guided to perform any combination of motions available on the prosthetic device. As development of the game continues, level creation can work to practice useful motions, connecting them to songs and making practice less laborious and improving continuum of care for these patients..

Future Improvements

Game

Currently, GRIP serves as a proof-of-concept that can be built on for future use cases. One necessary feature that will need to be implemented in the future is another interface with the Case Western designed "Hub" device, which is meant to read the signals of motor intent. Because the Hub is still in development, it was not possible for this important piece to be included in this project. Furthermore, the game still requires development of new levels in the form of beatmaps. New beatmaps serves as both an important part of user engagement, and an option for user progression and difficulty scaling.

In addition to beatmaps, it may be helpful to implement a set of difficulty scaling settings for the game. The game was designed with the ability to adjust slider size, note sizes and tempo but this has not been linked to a difficulty setting that the user can adjust. Exposing these would allow for finer control over the difficulty of individual songs. A recommended progression of songs accompanied by scaling difficulty parameters would be the ideal system to allow for the most complete spectrum of difficulty options.

Testing Apparatus

There some improvements that can be made to make this device better to use. One of them is the addition of the thumb abduction/adduction. By adding this motion the device would be complete in matching the motions of the DEKA LUKE arm. This is what the device was originally going to mimic so this is on of the bigger improvements.

Acknowledgements

We would like to thank the Kate Gleason College of Engineering's Multidisciplinary Senior Design Program for their financial support for our project as well as our faculty guide Cory Stiehl, PhD, and other useful references we communicated with in setting up and planning this project including Dustin J. Tyler, PhD and Ian Schreiber, MFA. We would especially like to thank Matthew Williams, PhD, our project sponsor and point of contact for the Cleveland Functional Electrical Stimulation (FES) lab. This work was supported by the DARPA BTO HAPTIX program through the Space and Naval Warfare Systems Center (Pacific contract no. NC66001-15-C-4041) and by the U.S. Dept. of Veterans Affairs RR&D Service Program (Merit Review Award #I01 RX00133401 and Center #C3819C).

References

[1] V. Bégel, I. D. Loreto, A. Seilles, and S. D. Bella, "Music Games: Potential Application and Considerations for Rhythmic Training," Frontiers in Human Neuroscience, vol. 11, 2017.
[2] B. Wong, "Points of view: Color blindness," Nature Methods, vol. 8, no. 6, pp. 441–441, 2011.
[3] The Amputee Coalition. "Limb Loss Statistics". [Online]. Available: https://www.amputee-coalition.org/resources/limb-loss- statistics/.
[4] G. McGimpsey and T. C. Bradford, "Limb Prosthetics Services and Devices - NIST," NIST Report on Limb Prosthetics Services and Devices: Market Analysis, 28-Apr-2017. [Online]. Available: https://www.nist.gov/document/239limbprostheticsservicesdevices.pdf.
[5] C. Prahm, I. Vujaklija, F. Kayali, P. Purgathofer, and O. C. Aszmann, "Game-Based Rehabilitation for Myoelectric Prosthesis Control," JMIR Serious Games, vol. 5, no. 1, 2017.
[6] Thingiverse.com, "All in one Teensy 3.2 case by Hyss," Thingiverse. [Online]. Available: https://www.thingiverse.com/thing:2431529.
[7] Thingiverse.com, "MPU6050 Mount by SrChispas," Thingiverse. [Online]. Available: https://www.thingiverse.com/thing:2911596.