UNIVERSITY OF MIAMI

AUDITORY FITTS TESTING OF MUSIC CONTROLLERS

By

Mark Gill

A THESIS PROJECT

Submitted to the Faculty
of the University of Miami
in partial fulfillment of the requirements for
the degree of Master of Science in Music Engineering Technology

Coral Gables, Florida

May 2015
Auditory Fitts Testing of Music Controllers

Abstract of a Master’s Research Project at the University of Miami

Research Project supervised by Dr. Christopher L. Bennett
Number of Pages in Text: 56

Motion based music controllers are becoming increasingly more popular. These instruments draw influence from instruments such as the Theremin and Potentiometre d’espace by allowing the musician to create music in free space. MIDI, a protocol used to communicate between digital music devices, is often used in these controllers to allow easy incorporation into a musician’s work environment. Complex mapping schemes are often provided to the musician to allow customization over their controller’s functionality. Motion controllers utilize accelerometer, gyroscope, and magnetometer sensors to capture the musician’s motion. Motion can be quantified by the distance traveled between two targets and the travel time between the targets. Fitts’ law is a motor function model that can be used to describe information transfer using these motion quantifications. Fitts’ law returns a measure of efficiency for each input device tested through the measures of movement time and task difficulty. Typical Fitts’ law testing incorporates a visual feedback system that is not ideal in measuring devices created for a musical purpose. An auditory feedback testing environment was devised to provide a more accurate testing environment for the motion based controllers.

The test was composed of three main components including an Ableton Live set, a hardware platform to serve as the MIDI controller, and an OS X application to conduct the Fitts’ test. The Ableton Live set featured several tone generators to provide the auditory feedback. The Ableton Live set received MIDI information from the OS X application to alter the pitch of the tone generators. One tone generator was used a destination pitch and was played in the right ear. The controllers were used to tune the tone playing in the left ear. In the experiment the test time was held as the independent variable, and completed mappings was dependent. A subject completes a mapping by tuning the controlled tone generator to the destination pitch. Subjects completed the experiment with a motion MIDI controller and a potentiometer MIDI controller.

The testing was validated using a t-test to determine the significance of the finding. The index of performance (IP) scores returned from the auditory Fitts’ testing was used as the test variable. Eight subjects were tested and their IP scores were averaged. The motion controller’s IP score was higher than the
potentiometer’s score, indicating a more efficient performance. The t-test returned a confidence interval of 96% that the results were significant. The tests were averaged to find an average movement time savings of 1.6 seconds per mapping.

This research shows that a motion based music controller was more efficient when compared against a potentiometer controller. The research was also able to show that Fitts’ law is appropriate when providing auditory feedback. The research was conducted in a 1D test environment. Future testing of motion controllers could utilize their multidimensional properties and extend this research to two or three dimensions.
ACKNOWLEDGMENTS

First and foremost I’d like to thank my committee for their ideas, contributions, and inspiration. Everyone brought something unique to the table that fueled me along. Big thanks to the FORE Center for all the resources provided to me. My time researching there prepared me for this paper as I studied the MAPR inside and out. Thanks to all the previous hardware designers on the MAPR. We have come so far with this thing! Special thanks to Dr. Colby Leider for introducing me to Dr. Chester Udell. His eMersion platform was a big inspiration to me in this research study. Thanks to Tommi for all the hot meals and preliminary testing to make sure the project was working smoothly. I cannot thank my parents enough for putting me through school and letting me follow these dreams I have had of working with audio electronics. Big thanks to the guys at Livid Instruments for getting me interested in controllerism.
# TABLE OF CONTENTS

**ACKNOWLEDGMENTS** ........................................ iv  
**LIST OF TABLES** ............................................. vii  
**LIST OF FIGURES** ........................................... viii  

**CHAPTER**

1 *Introduction* ................................................. 1  
1.1 Focus and Aims ............................................. 2  
1.2 Motivation .................................................. 3  
1.3 Anticipated Outcomes .................................... 3  

2 *Background* .................................................. 5  
2.1 Previous Work ............................................. 5  
2.1.1 Theremin ................................................. 6  
2.1.2 MIDI and MIDI Controllers ............................. 7  
2.1.3 Mapping .................................................. 8  
2.1.4 Inertial Measurement Units and Attitude and Heading  
Reference System (AHRS) .................................. 9  
2.1.5 Fitts’ Law ................................................. 15  
2.1.6 Fitts Testing Procedures and Applications .......... 18  

3 *Proposed System* ............................................. 23  
3.1 Chapter Three ............................................. 23  
3.1.1 Ableton Live Set ..................................... 23  
3.1.2 Hardware Platform .................................... 26
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.3 Software Platform</td>
<td>29</td>
</tr>
<tr>
<td>4 Testing and Evaluation</td>
<td>42</td>
</tr>
<tr>
<td>5 Results</td>
<td>46</td>
</tr>
<tr>
<td>6 Discussion</td>
<td>53</td>
</tr>
<tr>
<td>6.0.4 Conclusion</td>
<td>53</td>
</tr>
<tr>
<td>6.0.5 Variation from Visual Tests</td>
<td>53</td>
</tr>
<tr>
<td>6.0.6 Future Work</td>
<td>55</td>
</tr>
<tr>
<td>LIST OF REFERENCES</td>
<td>57</td>
</tr>
<tr>
<td>APPENDIX</td>
<td></td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MIDI Mappings in Ableton</td>
<td>26</td>
</tr>
<tr>
<td>2</td>
<td>Motion Packet Packing</td>
<td>29</td>
</tr>
<tr>
<td>3</td>
<td>Potentiometer Packet</td>
<td>29</td>
</tr>
<tr>
<td>4</td>
<td>t-test inputs</td>
<td>46</td>
</tr>
<tr>
<td>5</td>
<td>t-test Output</td>
<td>46</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MOOG Theremin</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Generic IMU Depiction</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>Theremin</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>WISP</td>
<td>14</td>
</tr>
<tr>
<td>5</td>
<td>Source Audio’s Hot Hand</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>Fitts’ Tapping Test</td>
<td>19</td>
</tr>
<tr>
<td>7</td>
<td>Fitts’ Original Tapping Device</td>
<td>20</td>
</tr>
<tr>
<td>8</td>
<td>Ableton Simpler Patch</td>
<td>24</td>
</tr>
<tr>
<td>9</td>
<td>Ableton Clip View</td>
<td>25</td>
</tr>
<tr>
<td>10</td>
<td>Uncased MAPR</td>
<td>29</td>
</tr>
<tr>
<td>11</td>
<td>OS X Application GUI</td>
<td>30</td>
</tr>
<tr>
<td>12</td>
<td>System Signal Flow</td>
<td>31</td>
</tr>
<tr>
<td>13</td>
<td>MIDI Signal Flow</td>
<td>37</td>
</tr>
<tr>
<td>14</td>
<td>Test Instructions</td>
<td>44</td>
</tr>
<tr>
<td>15</td>
<td>Motion and Potentiometer MAPR Devices</td>
<td>45</td>
</tr>
<tr>
<td>16</td>
<td>Motion Regressions</td>
<td>48</td>
</tr>
<tr>
<td>17</td>
<td>Potentiometer Regressions</td>
<td>49</td>
</tr>
<tr>
<td>18</td>
<td>Comparison of Regressions</td>
<td>50</td>
</tr>
<tr>
<td>19</td>
<td>P-Score Regression</td>
<td>51</td>
</tr>
<tr>
<td>20</td>
<td>Average Movement Time</td>
<td>52</td>
</tr>
</tbody>
</table>
1

Introduction

An inertial measurement unit (IMU) is a device that records motion data from which position or orientation can be estimated. IMUs are often designed as a wearable device, so that a person can use the device to record raw accelerometer, gyrometer, and sometimes magnetometer data of their own kinematic motion. Wireless IMUs provide an attractive solution for motion tracking due to their noninvasive properties[Bruckner et al., 2012]. An ideal IMU is small and light-weight to allow a user free motion.

IMUs are used as digital instruments or music controllers, where motion sensing systems are used to control sound processes or synthesize audio[Miranda and Wanderlay, 2006]. Although these devices are used to create artistic expressions, the need for scientific evaluation is crucial for the development of existing gestural controls[Wanderley and Stravinsky, 2001].

Evaluation criteria for this research included measuring a potentiometer controller’s efficiency as a benchmark, and testing a motion controller against the benchmark. A controller’s efficiency is a time based metric based on its usability. A more efficient controller will allow the user to complete more mapping tasks within a given period of time. To evaluate existing gestural controls this study compared the efficiency of potentiometer based music controllers versus gestural or motion controllers. **Efficiency is a critical aspect of a music controller**
as they are used in real time music applications where users desire to be able to execute many commands in quick succession. The efficiency of the two controllers was measured using an index of performance (IP) metric described in the Background chapter. A controller with a higher IP can be said to be more efficient. This study aims to assess the sensor and algorithm technology in its current state of the art.

1.1 Focus and Aims

Users desire flexibility from music controllers, and researchers have found that users are more satisfied with a controller as its mapping scheme becomes more complex [Bellona, 2014]. 3D motion controllers allow for a more intricate controller versus one dimensional potentiometer based controllers. This study aims to compare the usefulness of motion controllers in real time music performance applications versus potentiometer based controllers. To complete this task an appropriate hardware sensor system, OS X MIDI software, and Fitts Law tests were designed to compare these two controller variations. The hardware sensor platform used in this study is the Multi-Axial Profile Recorder (MAPR) developed by researchers from the Music Engineering Technology Program at the University of Miami. The most recent revision of which was created by Rehrig [Rehrig, 2014], who utilized the MAPR for controlling sound location through wave field synthesis. The objectives for this study can be summed by the following points:

- Aim 1: Design hardware platform, OS X software platform, and Ableton
Live Set to conduct auditory Fitts’ test.

- Aim 2: Conduct 1D Fitts’ law testing to compute Index of Performance (IP) of motion controllers and potentiometer based controllers.

1.2 Motivation

Motion sensor controllers are an emerging technology and need to be scientifically tested against their predecessors in terms of usability. Artists have embraced the possibility of musical expression through motion controllers by funding Kickstarter projects such as the eMotion Emersion[Udell, ], and incorporating consumer products such as the Source Audio Hot Hand[Sou, 2013] into their music creation process. This study was in service to those musicians working with these new technologies to ensure that their on stage performances and in studio recordings are equipped with the most efficient controllers. The testing will conclude that either the controllers have no difference in efficiency, or one controller will be more efficient than the other.

1.3 Anticipated Outcomes

It is anticipated that this study will determine if there is or is not a difference in motion and potentiometer controller efficiency in a 1D control environment. Some bias may be encountered as subjects will likely be more accustomed to potentiometer input devices. We expect subjects to become accustomed to the motion controller through multiple tests, and perform better as more tests are taken. The study will serve as a pilot to more field appropriate
studies that utilize the motion controller’s 3D capabilities. The study may have different results as more controls are added to a test. Fixing multiple control parameters may prove more efficient when using motion controllers as multiple mappings can be completed with a single peripheral.
2

Background

2.1 Previous Work

In the 1940s Hugh LeCaine built the Electronic Sackbut, which is largely considered to be the first voltage controlled synthesizer. Control voltage would later be repurposed into modular and standalone synthesizers to allow users extended control over electronic instruments. Control Voltage is presented as a CV input jack that can be routed to any CV output jack. Control voltage terminals allow signals from modules to be routed to any other module at the users’ will. These terminals were the original form of generalized music control. Control voltage was largely superseded by Musical Instrument Digital Interface (MIDI) invented by Dave Smith of Sequential Circuits. The user can assign MIDI outputs from MIDI hardware or software to any desired MIDI input. MIDI has been incorporated into standalone devices known as MIDI controllers. These devices are equipped with USB and DIN connection terminals to transmit MIDI packets to other MIDI devices. Traditional MIDI input interfaces include potentiometers and buttons. MIDI is used in this thesis as the basis of communication between motion and music control. Motion and music control have been linked together since the 1920s when the Theremin was created. Since then, IMUs have been introduced as a form of motion music controller.
2.1.1 Theremin

The Theremin is a gesture controlled synthesizer created by Leon Theremin in the 1920’s. The Theremin was unique among all other instruments as it requires no physical contact to be played. The Theremin consists of two metal rods which are used to control an oscillator’s pitch and volume. The synthesizer is controlled based on the musician’s proximity to these two rods. Higher pitched sounds can be produced by moving closer to the pitch rod, and louder sounds can be produced by moving away from the volume rod. The Theremin can be viewed as a controller with two output data points. The two input rods act as permanent outputs to the controls pitch and volume. This alternative view of the Theremin pictures it as a motion music controller. The Theremin offers precise sound control, but requires the musician to be in the vicinity of the Theremin to play it. Wireless IMU controllers give the user a new found freedom of expression where this limitation is removed.

Pierre Schaeffer created a digital rendition of the Theremin device with his Potentiometre d’espace invention. The device features 5 passive inductor rings each assigned to a speaker within the performance room. The rings are controller by a magnet held in the musician’s hand. The musician uses the magnet to direct the sound around the room by placing it closer to a desired ring. Similarly to the Theremin, the Potentiometre d’espace is single purposed and does not allow for generic control.
2.1.2 MIDI and MIDI Controllers

As companies started to create digital instruments it quickly became evident to Dave Smith that discrepancies between manufacturers communication protocols could keep digital instruments from being compatible with each other. Dave Smith’s first MIDI capable synthesizer, the Prophet 600, hosted a sea of rotary potentiometers with dedicated purposes. This product design is influential as most MIDI controllers continue to feature rotary potentiometers as well as slider potentiometers. The potentiometer quickly became synonymous with music control. MIDI still stands as the leading control protocol and is still implemented in modern music workstations and control devices. The adaptiveness of MIDI makes it a perfect platform for this research as the created motion controller can be used for any music creation purpose.

MIDI controllers have taken various forms including drum pads, ribbon
controllers, and keyboard controllers. Drum pad controllers feature an array of buttons. Drum pads are largely intended for sending note on/off messages accompanied by velocity and note number. Drum pads can be used to trigger samples, initiate sequences, and finger drum. Ribbon controllers feature touch sensitive ribbon sensors to create an arbitrary controller. The user can slide along the ribbon to transmit continuous control data, similarly to using a sliding potentiometer. Keyboard controllers feature a piano key roll to transmit note on/off messages. Keyboard controllers sometimes feature weighted keys and velocity sensitivity. Aftertouch can be incorporated, which allows the musician to press onto the keys after the initial key engagement to send an additional signal at the note number. Keyboard controllers often feature drum pads and potentiometers control surfaces as well.

2.1.3 Mapping

A map is described by the map-territory relation which states that a map is a representation of an object [Bateson, 1972]. In music control mapping can be described as the process that takes gesture input data and translates it to sound [Bellona, 2014]. When the motion is used to control an effect, the motion is manifested as sound as a modulation. The mapping of movement data is not a direct one-to-one scale. The energy used to express some music control does not directly correlate to the output of the control system. Input data requires some data modification to be relevant in its application. Bellona proposes the following data modifications between input and output: interpolation, filtering, offset,
**smoothing, and scaling** [Bellona, 2014]. These mathematical relations create some meaningful output from a controller’s input system, and in many cases require data modification or sensor data conditioning to occur before mapping can be completed. The sensor data conditioning strategies used are described in the IMU section of this chapter. Mapping schemes can be described as one of the four strategies one-to-one, one-to-many, many-to-one, or many-to-many [Chantasuban and Thiemjarus, 2009]. These strategies describe the number of data inputs-to-number of data outputs. This relationship describes the number of sensor data points used to determine another value which will be used as control data. A one-to-many mapping would take a 1D accelerometer reading and extract frequency and amplitude [Chantasuban and Thiemjarus, 2009]. To create a many-to-one mapping, a 3D accelerometer can be used to extract total acceleration. Total acceleration is an average of the three accelerometer axes with gravity subtracted.

### 2.1.4 Inertial Measurement Units and Attitude and Heading Reference System (AHRS)

IMUs utilize accelerometer, magnetometer, and gyroscope data to record motion data. Accelerometers register gravitational acceleration and external acceleration together. Gravitational acceleration is distributed across the X, Y, and Z axis determined by the orientation of the sensor to the Earth’s gravitational field. In rotational algorithms the gravitational force is desired and the external force is detrimental. The gravitational acceleration distribution
describes the sensor’s orientation without heading, while the external acceleration component represents user motion. AHRSs are similar to IMU, but differ in that on board processing is completed to compute rotational data.

Gyroscopes measure angular velocity and are described in units rad/sec. Gyroscope readings are not affected by any external forces making them a reliable sensor. Magnetometers measure the strength and direction of the local magnetic field. The local magnetic field is comprised of the Earth’s magnetic field and any magnetic field induced by nearby objects. In motion detection applications the local magnetic is often discarded to try to measure the Earth’s magnetic field alone. Removing local magnetic readings requires calibrating the magnetometer data to separate local and global data.

Magnetometer readings are only required in applications that desire a heading data point to describe the relation of the sensor’s orientation in reference to the Earth. This relation is not necessary for music control applications as relative motion can be used to create control data.

x-IMU

The x-IMU[x-i, 2012] developed by x-io is a versatile and well documented open source IMU sharing many features with the MAPR. It relies heavily on rotation quaternion found using the open source Attitude Heading Reference System (AHRS) algorithm developed by Madgwick[Madgwick et al., 2011]. The AHRS algorithm is presented on the x-IMU website and will be utilized in this project. The algorithm’s derivation can be found in Magdwick’s cited paper. The
Algorithm utilizes a gradient descent gyroscope error estimation. His algorithm estimates orientation by integrating the estimated rate of change quaternion which is calculated from gyroscope or angular velocity measurements. Integrating velocity yields a position quaternion. Madgwick compares the gradient descent method to the popular Kalman filter orientation estimation, which is considerably more complex mathematically. Kalman filter implementations can demand high sampling rates [Madgwick et al., 2011], making it a suboptimal choice for a Bluetooth Low Energy implementation. Madgwick has concluded that his technique performs as well as Kalman filter implementations. An addition to Madgwick’s equation can be added to account for gyroscope bias drift [Madgwick et al., 2011]. This filter referred to as the Mahony filter, can track gyroscope bias over time allowing for the IMU to be calibrated only once.

Figure 2: Typical IMU Considered Axes. The box represents an IMU in the x, y, z Cartesian space. The measurements reported by the sensors are in reference to the x, y and z dimensions.
when turned on. Failing to track gyroscope bias drift will cause an error in the quaternion estimation over time. This error will be compounded as the algorithm requires an integration of the gyroscope data.

The x-io AHRS implementation determines orientation for a single IMU. More complex methods use the heading provided from a magnetometer to relate sensor motion within a sensor network. The x-IMU features many open source projects on its website including its use as a MIDI controller. A more abstract demonstration shows the x-IMU tracking oscillatory position data through the use of double integration of an accelerometer signal. The double integral technique is criticized due to small errors in quaternion calculation or sensor noise being accumulated by double integration.[Slifka, 2004]. Position is an elusive measurement that would be useful in this application, but is unreliably obtained. It would allow users to control with vertical or horizontal displacement motion, adding an additional type of motion to measure along with rotation.

WISP

The WISP[Till et al., 2007] is an IMU featuring the same type of sensors as a MAPR. It has shown sufficiency in music control purposes and has been used to monitor motions of fire dancers. It is able to sample at 40 or 80 Hz depending on sensor count over the wireless transport. This provides some notion that the MAPR can be used in similar applications as it transmits samples at 50 or 100 Hz. The WISP features an on-board DSP to convert raw sensor values to quaternion. The MAPR has opted to place all signal processing in the backend of
a software application, or to take quaternion data directly from a sensor capable of producing these data. This provides the advantage of quick debugging, increased memory allowance, and allows for a smaller sensor package.

**Source Audio Hot Hand**

The Source Audio Hot Hand is a commercially available IMU used for music control. The Hot Hand consists of a 3-axis accelerometer that is worn on the user’s hand. The device communicates wirelessly to a computer with Hot Hand Software Editor installed. The software’s scheme allows users to create six processes, which can be mapped to a total of 10 MIDI send signals. The processes are created from one of the three accelerometer axis. Each process can have its own center, invert, depth, and smoothness setting. The MIDI send signals can be assigned continuous controller, pitch bend, or aftertouch. The
minimum and maximum MIDI signal can also be adjusted. The product is highly stable and able to be used in real time audio applications. The setup of the device and software encourages users to move their hands on a single axis to produce a desired result. Demonstration videos show users rotating and repositioning their hands to control music software. These demonstrations serve as inspiration for mapping techniques for the MAPR.

Figure 4: Wireless Inertial Sensor Package (WISP) [Till et al., 2007]. An academically made IMU with an open source community.

Figure 5: Source Audio Hot Hand (Source Audio, Woburn, MA). A commercial IMU with extensive mapping capabilities.
2.1.5 Fitts’ Law

Fitts’ Law states that the time required to move to a destination is logarithmically proportional to the distance traveled, and is logarithmically, inversely proportional to the width of the destination. This test can be used to test the motor response of a test subject when asked to perform aimed movements [Fitts, 1954]. Fitts’ original relationships and laws has proven to be one of the most robust and adopted models from experimental psychology [MacKenzie, 1992]. A popular application of Fitts’ Law is to test input devices by conducting tapping tests. The test is conducted by instructing a user to tap between two visual points as quickly as they can within an allotted time period. The distance between the center points of the tapping destinations is measured, and the width of each tapping destination is measured. This definition hints at a visual component to the testing where the subject fixates on the intended final location.

To conduct an auditory version of Fitts aiming test the metrics of distance and width must be defined for the auditory variation of the test. In Fitts’ model equation (2), distance and width are inserted into a logarithm, implying they must carry the same unit. In this research the subject is providing MIDI values in the range 0 to 127 as the input. This input unit can serve as the basis for the parameters distance and width. The subject will track two MIDI destinations separated by some MIDI distance. The target they are tracking will have some error margin described by a MIDI width. For the potentiometer and motion
device this has a direct correlate to a distance in position as used in previous Fitts’ tests. The rotation of the controllers is linearly correlated to the output MIDI value. The MIDI and perceptual data scale symmetrically. **The subject is still aiming, but is not using their eyes to determine the final location but their ears.** The visual and audio feedback from the tests may differ, but the motion of the user is analogous. This difference in visual and audio feedback was tested by Friedlander [Friedlander, Naomi Schlueter and Mantei, 1998], and he found that Fitts’ Law did not fit his testing of auditory feedback when traversing a bullseye menu. This previous research may not be applicable in this case as the feedback in Friedlander’s tests was short beeps to designate accessing a new portion of a menu. These beeps were sudden and did not give the subject any indication of how close they were to reaching the target. Fitts Law may be more applicable for this application as the auditory feedback used to judge distance from the target is continuous and instantaneous.

Another previous study found that Fitts’ Law has been used to create models of several input devices to compare the group of devices [MacKenzie et al., 1991]. This study verifies that some significant data may result from conducting this study. In this study the researchers merely compared mean movement time to determine which device is most efficient. This is possible by preserving the test conditions width and distance for all subjects.

Fitts devised the metric to measure index of performance (IP) to measure the information capacity of the human motor system [MacKenzie, 1992]. IP is a
measure of the index of difficulty (ID) of a given task and the movement time (MT) to complete the task. IP is defined by equation 1, and is in units of bits/sec. The human motor system is said to be transmitting information to the computer by completing a task. The index of performance describes the rate of information transmission. The index of difficulty shows the relation that acquiring a smaller target at a larger distance is more difficult than tracking a large object at a small distance.

\[ IP = \frac{ID}{MT} \]  \hspace{1cm} (1)

ID can be solved for directly in equation 2 where \( D \) is the distance between tapping destinations, and \( W \) is the width of each destination. IP is linearly proportional to \( ID \) and linearly, inversely proportional to \( MT \). It suggests that the relationship between MT and ID is linear [MacKenzie, 1992] even though ID is solved for logarithmically.

\[ ID = \log_2 \left( \frac{2D}{W} \right) \]  \hspace{1cm} (2)

ID can also be solved for over the regression described in equation 3 where \( T \) denotes the average time for completion, \( a \) and \( b \) are model parameters.

\[ T = a + b \log_2 \left( 1 + \frac{D}{W} \right) \]  \hspace{1cm} (3)

The fraction \( \frac{D}{W} \) can be described as the task difficulty, term \( a \) can be described as the reaction time, and the parameter \( b \) describes the system input device.

Parameter \( b \) is defined as the reciprocal of \( IP \), and demonstrates performance of
the device by comparing movement time to task difficulty. Typically, and in this study, the distance $D$ describes a one-dimensional distance. Specifically for the auditory variation of the test, $D$ is **MIDI distance** and $W$ is **MIDI Width**. The definitions for $a$ and $b$ are preserved from the visual test to the auditory test. They describe the *device* under test and not the test being performed. This test will determine if IMUs can be used to complete a single music control task as efficiently as using the traditional potentiometer control surface.

**2.1.6 Fitts Testing Procedures and Applications**

Studies including Fitts’ original paper on his theorems[Fitts, 1954], Epps’ comparison of cursor control devices [Epps, 1986], Mackenzie’s comparison of input devices[MacKenzie et al., 1991], and Ware’s and Mikaelian’s evaluation of eye tracking devices[Ware and Mikaelian, 1986] are highly documented and cited test procedures incorporating Fitts’ law into their test analysis. These studies focus on visual feedback systems where test subjects use their eyes to use the provided input devices. A generic Fitts tapping test is shown in figure 6.

Fitts’ original testing device is shown in figure 7. This device has two target plates that can be set to 1 of 4 widths. The target plates have adjacent plates to capture undershoot and overshoot taps. The plates were tested at 4 different distances. Fitts’ tested two stylus held by the user which varied in weight and size. The index of difficulty varies over the range 1 to 7. Each subject completed 32 trials by fully crossing the width variations with the distance variations, and completing each sequence twice. Each trial lasted 15 seconds.
Figure 6: **Generic Fitts’ Tapping Task Set Up.** The distances for each destination are equal, and the distance between the two is measured from their center points.

During the trial the subject was instructed to alternate tapping between the two target plates. Fitts observed a mean error rate of 1.2% and 1.3% for the lighter and heavier stylus respectively, and notes that practice has a relatively small effect on subject performance. Both stylus had relatively similar index of performance scores. Fitts analyzed three other motor skill analyzation tasks within the research and expressed similar results. Fitts was able to conclude that his proposed relation between speed, amplitude, and tolerance may be general and usable for other applications.

Epps extended Fitts tapping tests to digital input devices. He tested Fitts’ model as well as Jagacinski’s and Kvalseth’s motor function models. Jagacinski’s and Kvalseth’s models vary slightly from Fitts’ original model in equation 3. Each model uses the same variables MT, ID, and IP. Epps tested 12 subjects on 6 input devices including 2 track pads, 1 optical mouse, 1 trackball, and 2 joysticks. Each input device was tested over 200 trials by each subject. Epps’ test differs form Fitts’ as his target acquisition method is not a true Fitts’ law task.
Subjects completed a 2-axis acquisition task versus a single axis task. Epps found the Fitts model to be an appropriate fit of the tasks completed. Epps conducted ANOVA analysis to determine the significance of his data and fit his data to the three models to determine which most accurately described his data set.

Mackenzie’s experiment is similar in structure to Epps, but more closely resembles Fitts’ original tapping tests as his test software is similarly designed to Fitts’ tapping apparatus. Two on screen rectangles are shown to a user and they are to complete the task as quickly as possible ten times. Mackenzie’s test differs to Fitts’ as the test time is variable and tap count is constant, and in Fitts’ test tap count is variable and test time is constant. Mackenzie used 4 widths and 4 distances to create 16 trials per subject by fully crossing the widths and distances. Mackenzie tested 12 subjects. Subjects completed the trials on three inputs devices including a Macintosh mouse, a Wacom tablet and stylus, and a Kensington trackball mouse. Subjects participated in a dragging test and also a pointing task. In the dragging task a virtual object is dragged from one target to
the other, and in the pointing task the targets are alternatively clicked.

Mackenzie analyzed mean movement times for each subject and device pair to demonstrate which device is faster. Mackenzie notes in his results high correlation between MT and TD. This correlation reinforces Fitts’ original documentation on motor system response.

Ware’s and Mikaelian’s test of eye tracking devices poses a new application of Fitts’ tests as the input devices are not operated with the hands. Fitts has previously suggested that the model should work generally for any motor characterization [Fitts, 1954]. This input device is of special interest to this research since the research tests a prolonged fixation over a target as a tapping mechanism. The paper supports the notion that all previously discussed tests are visual in nature as they will presumably fixate on the desired target before acquiring it. The researchers tested a fixation interval of 0.4 seconds, pressing a keyboard key to indicate the tap, and an on screen button that is fixated on after fixating on the target. Ware’s and Mikaelian’s allowed testing time to be variable and acquisitions to be non-variable. The researchers tested 4 subjects over 35 trials each. The first 7 trials for each subject were withdrawn from analysis. They found that fixation ”tapping” and a hardware tap were equally fast tapping measures. This suggests that fixation can be used in this research.

This research was adapted from the discussed studies. The movement time was held as the independent variable and completed tasks dependent. This architecture is pulled from the Fitts’ study. 4 widths and distances were chosen
to create a 16 trial test. The test was conducted on two input devices. The input
mechanical tapping mechanism, so a fixation time is used similarly to Ware’s and
Mikaelian’s test. Subjects were not given a preparatory test as Fitts showed that
this did not improve results. Eight subjects were tested which was an appropriate
amount of subjects as demonstrated by the previous studies. The test asked
subjects to complete a single axis tapping test with auditory feedback.
Proposed System

3.1 Chapter Three

An Ableton Live set and hardware and software systems have been prepared to conduct an auditory Fitts test. The Ableton Live set produced the audio the test taker used to take the Fitts test. The hardware system (MAPR) was responsible for collecting raw data values and wirelessly transmitting them to the software system. The software system was responsible for creating MIDI packets from the raw data, transmitting the MIDI packets to a digital audio workstation (DAW), and conducting the Fitts test.

3.1.1 Ableton Live Set

An Ableton Live Set is an Ableton save file containing tracks, virtual instruments, effects, and samples. The set used for the experiment required three tracks, a tone generator on each track, and a MIDI clip for each track. The three tracks represented the motion controller, the potentiometer controller, and the destination sound. Two tracks are used at a time to complete a test. The destination track was always used and one of the controller tracks was used during a test. Each track contained an identical virtual instrument, Ableton’s Simpler. The Simpler patch is shown in figure 8. Ableton standard does not come with any tone generators, so the sampling instrument Simpler was used as it comes with Ableton. A sinusoid was generated using wavtones.com and loaded into the sampler. The Simpler changes the sample rate of the sinusoid to create...
different frequencies. Zero crossing must be used for the first sample frame, and the sample before the next zero crossing was the last frame. Each track’s Simpler used the same sample and settings. The Simpler had a MIDI settable parameter labeled transposition that was controlled by the MIDI devices. The transposition parameter was tunable to 96 musical notes on the Western scale. 127 corresponds to the range of MIDI continuos control values, so the MIDI controllers produced more values than the transposition knob needed. Ableton mapped the 127 data values down to the 96 settings for the transposition parameter. Due to the low resolution of a 7-bit parameter, each increment or decrement in the transposition parameter effected the sound in a ’step’ manner. The jump in pitch was similar to touching notes on a keyboard roll. Alternatively a 14-bit MIDI message system could be used and a frequency parameter could be tuned. This would eliminate the stepping quality of the tracking parameter. This would result in a more difficult test and cause the movement times to increase. To further differentiate between the destination and the controlled tone generator, the destination tone was panned hard right and the controlled tone was panned hard
left. Ableton’s grid view was used to completed the testing session. The grid view allowed for looping of the audio and can loop for an arbitrary amount of time. An Ableton scene was prepared for each MAPR type. An Ableton scene launches clips assigned to that scene. Each track had one clip of repeating quarter notes on middle C. In the motion MAPR scene the destination and motion track clips were activated, and in the potentiometer MAPR scene the destination and potentiometer tracks were activated. Each scene was activated to perform a motion or potentiometer Fitts test. A complete overview of the grid view and scenes is shown in figure 9. Ableton’s MIDI mapping tool was used to correspond MIDI CC messages to certain controls or parameters within Ableton. The MIDI mapping scheme is shown in Table 1. The first three mappings were to control the transposition parameter on the tone generators (TG) for each

![Figure 9: Ableton Tracks in clip view. Two tracks were controlled by the MIDI controllers and the third track was the destination track.](image-url)
audio track. The last two mapping were used to start and stop audio playback through Ableton’s built in transport.

### 3.1.2 Hardware Platform

Rob Rherig designed the MAPR with the following goals: 1) a way to communicate with a remote computer wirelessly; 2) sub-sensors that measure multiple dimensions of movement; 3) a way to manage power to the board; and 4) a microcontroller to handle all of the onboard management [Rehrig, 2014]. These design goals mostly line up with the hardware platform requirements for this study. To further expand these goals, the hardware system must be capable of 5) sampling a potentiometer, and 6) provide sufficient raw data for data fusion. Through many circuit revisions Rherig created a small and power efficient Bluetooth Low Energy IMU. Rherig’s board provides access to the microcontroller’s analog-to-digital converters (ADC) for sampling the potentiometer, and has been configured to sample at 50 Hz. These features meet requirements 5 and 6. The MAPR’s microcontroller was the BlueRadios BR-LE4.0-S2A single mode low energy chip. This chip is a Texas Instruments CC2540 with an added antenna. BlueRadios provides an API to interface with

<table>
<thead>
<tr>
<th>MIDI Control</th>
<th>Path</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TG Motion</td>
<td>Transpose</td>
</tr>
<tr>
<td>2</td>
<td>TG Potentiometer</td>
<td>Transpose</td>
</tr>
<tr>
<td>3</td>
<td>TG Destination</td>
<td>Transpose</td>
</tr>
<tr>
<td>4</td>
<td>Transport</td>
<td>Start</td>
</tr>
<tr>
<td>5</td>
<td>Transport</td>
<td>Stop</td>
</tr>
</tbody>
</table>

Table 1: MIDI Mappings in Ableton
the chip. The chip’s SPI port was utilized to connect the Bosch Sensortech BMC-050 magnetometer and accelerometer chip and STMicroelectronics L3G4200DTR Gyrometer chip. The BMC-050 provides configurable 10-bit accelerometer data. The acceleration data can be configured to the acceleration ranges $\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$. The smaller g ranges are useful for capturing less intense motions and larger g ranges can be used to capture more abrupt acceleration changes. The acceleration range $\pm 2g$ was used in this research since it allowed the limited 10-bit data range to be used for precise motions. The BMC-050 also features a low-pass filter (LPF) which was utilized. The cutoff for the filter was set to 1 kHz. The LPF eliminated some noise from the system. The gyrometer features 16-bit resolution raw data values with selectable degrees per second (dps) scale. The gyrometer can be set to $\pm 250\,\text{dps}$, $\pm 500\,\text{dps}$, or $\pm 2,000\,\text{dps}$. Similarly to the accelerometer, the lowest range of $\pm 250\,\text{dps}$ was used to capture more precision. In addition to the BLE chip and sensors, the board featured a 2-way LED used for monitoring connections and for signaling communication events from the OS X device. The LED periodically flashes orange when not connected, and green when connected.

The BLE standard allows for connection intervals as low as 7.5 ms. The CC2540 is capable of connections this quick, allowing it to not be a bottle neck in communication throughput. In the application layer of the firmware, a timer was configured to sample the sensors every 20 ms and they were then queued for BLE transmission. One transmitted packet included X, Y, and Z axis data from the
accelerometer and gyroscope sensor, and also included a packet count. The packet count allowed for tracking of packet reception on the OS X device. The system must have a low packet drop rate for integration with sensor fusion algorithms. These algorithms require constant sampling as the sampling rate is used as part of the computation. When integration is used constant sampling is also required to minimize integration error. To ensure optimal transition between the MAPR and the OS X device, bit packing was implemented to minimize packet size. The packet was reduced by 2 bytes by reducing the accelerometer data from 6 bytes to 4 bytes. The packet layout is shown in Table 1.

The MAPR must first collect the appropriate raw data samples. Two separate MAPRs were prepared to act as either a potentiometer or motion device. A potentiometer device transmitted a single reading from an ADC shown in table 2, while a motion device transmitted the packet shown in table 1. By creating different packet structures, data transfer can be as efficient as possible. These packets were transmitted on separate Bluetooth Profiles. The potentiometer and motion Profiles had unique Services and Characteristics to allow the OS X device to differentiate between the two MAPR types. Using two MAPRs was also more appropriate for the motion testing, as the motion MAPR will not have a connected potentiometer while testing. Both potentiometer and motion characteristics were created as notify style characteristics versus read style characteristics. Notify characteristics do not require a handshake between the BLE devices. This transmission technique is required to allow for higher
<table>
<thead>
<tr>
<th>Packet Count</th>
<th>Gyro X</th>
<th>Gyro Y</th>
<th>Gyro Z</th>
<th>Acc X</th>
<th>Acc Y</th>
<th>Acc Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not Packed</td>
<td>Byte 1</td>
<td>2/3</td>
<td>4/5</td>
<td>6/7</td>
<td>8/9</td>
<td>10/11</td>
</tr>
<tr>
<td>Bit Packed</td>
<td>Byte 1</td>
<td>2/3</td>
<td>4/5</td>
<td>6/7</td>
<td>8/9</td>
<td>9/10</td>
</tr>
</tbody>
</table>

Table 2: Motion Packet Packing

<table>
<thead>
<tr>
<th>Packet Count</th>
<th>Potentiometer Reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte 1</td>
<td>Bytes 2/3</td>
</tr>
</tbody>
</table>

Table 3: Potentiometer Packet

sampling rates. A completed shot of the MAPR is included in figure 10.

3.1.3 Software Platform

A software platform was created to transform the MAPRs into MIDI devices and conduct the auditory Fitts testing. The software’s GUI is shown in figure 11. The software platform must 1) communicate with the MAPR, 2) fuse the sensor data using the Mahony filter, 3) map the fused data, 4) create and
deliver MIDI packets, and 5) conduct the Fitts law test. The platform was designed to allow users to control the testing without need for the proctor to be present during the test to prevent any experimental bias. The software platform acted as a liaison between the MAPR and Ableton Live. The software utilized CoreBluetooth to correspond with the MAPR, and CoreMIDI to correspond with Ableton Live. These APIs were provided by Apple with their Xcode platform.

![OS X Application GUI](image)

**Figure 11**: OS X Application GUI. The subject started the test by using the 'Begin Test' button. The 'Tests Completed' field was updated as mappings trials were completed. The remaining controls were for setting up the test.

**Hardware Integration**

CoreBluetooth, a Bluetooth API for iOS and OS X allows for communication between the MAPR and OS X. The incorporation of
Figure 12: Signal flow of the entire system. This figure shows the process of the data getting from the MAPR to Ableton. The signal state side shows the mapping of the data.
CoreBluetooth was first developed for iOS by developers at the FORE Center, and then moved over to OS X upon completion of the code. The CoreBluetooth development began several years prior to this research and provided a near perfect communication system. The iOS development featured rigorous data throughput testing to determine the quickest available sampling rate. Apple’s Bluetooth Developer’s Guide explains that the quickest connection interval allowable is 20 ms. Apple’s PacketLogger hardware development tool was used to show that the Bluetooth daemon will set the connection interval to 18.5 ms when given a desired connection interval of 20 ms. This discovery allowed for a sampling rate of 50 Hz on the MAPR. When porting the iOS code to OS X some discrepancies in the BLE Daemons of iOS and OS X were discovered. Testing software was ported to track the reception of BLE packets. It was found that the system was dropping a significant portion of the BLE packets. Debugging with PacketLogger showed the BLE Daemon was receiving the connection interval update, setting the interval appropriately, and then reverting to the default value seconds later. It was assumed that the update was happening too quickly and the MAPRs update call was being processed before the BLE Daemon had finished initializing the sensor to default values. This assumption was tested by delaying the connection interval update call, which fixed the dropped packet issue.

The CoreBluetooth implementation connected only to MAPR devices by reading advertised device names. Once a MAPR had been discovered the services of the MAPR are requested to determine MAPR type. The appropriate notify
characteristic was then subscribed to and data transmission begins. Upon receiving the packet it must be bit unrolled to create data variables. The unrolled values were raw voltage values that must be converted to appropriate units. The potentiometer value did not need to be unitized. The accelerometer and gyroscope values were found using the generic equation 5 where $U$ is a united value, $V$ is a raw voltage, $a$ is a scalar from the sensor’s datasheet, and $b$ is a calculated offset bias. The values for $a$ are shown in table 3. The scalar $a$ was dependent on the sensor’s sensitivity setting.

$$U = a \cdot V + b \quad (4)$$

<table>
<thead>
<tr>
<th>Accelerometer a</th>
<th>Gyroscope a</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{1}{256}$</td>
<td>$0.07\pi$</td>
</tr>
<tr>
<td>$\frac{1}{180}$</td>
<td></td>
</tr>
</tbody>
</table>

After applying equation 2, the accelerometer data was in a normalized non-united value and the gyroscope data was in units of $\frac{rad}{s^2}$. These data units were appropriate to be used by the Mahony algorithm. The Mahony algorithm accepted any normalized accelerometer value. The Mahony algorithm can be summarized by the following steps extracted from Madgwick’s implementation:

- Normalise accelerometer measurement
- Estimate direction of gravity and vector perpendicular to magnetic flux
- Compute error as sum of cross product of estimated and measured gravity direction of gravity
• Compute and apply integral feedback

• Apply proportional feedback

• Integrate rate of change of quaternion

• Normalise quaternion

Mapping

Mahony’s algorithm returned a rotation quaternion given by the form 
\( q_r = [q_0, q_1, q_2, q_3] \). Where \( q_r \) is a 1 by 4 vector describing the rotation quaternion.

This representation was advantageous as it allows for a single orientation value to be found from accelerometer and gyroscope measurements [Madgwick et al., 2011]. Once the quaternion has been acquired it was converted to Euler angles to describe the MAPR’s orientation in terms of roll, pitch, and yaw. In this research only the pitch value was used as only one degree of freedom was required to complete the test. Roll and pitch were valid values, but yaw read incorrectly due to not using a magnetometer. The yaw calculation in Madgwick’s analysis has also been shown to be the least accurate data point, while roll and pitch are near equally accurate [Madgwick et al., 2011]. The conversions from quaternion to Euler angles are shown in equations 5, 6, and 7. The Objective-C functions for \( \text{atan2} \) and \( \text{asin} \) return in units of radians within the range \(-\pi\) to \(\pi\). This range can be used to represent \(360^\circ\) of rotation about an axis. To allow left and right handed users ease of use with the motion devices, the absolute value of the Euler angle was taken to make movements in the left or
right direction produce an equal outcome.

\[
\psi = \text{atan}2(2q_2q_0 - 2q_1q_3, 1 - 2q_2q_2 - 2 * q_3 * q_3) \tag{5}
\]

\[
\theta = \text{atan}2(2q_1q_0 - 2q_2q_3, 1 - 2q_1q_1 - 2q_3q_3) \tag{6}
\]

\[
\phi = \text{asin}(2q_1q_2 + 2q_3q_0) \tag{7}
\]

Motion and potentiometer data was rescaled from their respective data ranges of \(-\pi\) to \(\pi\) and 0 to 2047 to the data range of 0 to 1. This allowed sensor data to be rescaled using the same generic equations and parameters. Once the data values had been conditioned they were scaled using a generic scaler to the range 0 to 127. The sensitivity of the MAPRs was held constant for all users to avoid any discrepancies in the tests.

**CoreMIDI**

CoreMIDI is OS X’s MIDI systems services API. The API allowed for a virtual MIDI source to be created to communicate with OS X’s MIDIServer process. The MIDIServer was used to receive messages from the virtual MIDI source and delegate the messages to all responding MIDI receivers connected to the OS X device. To create the virtual MIDI source, a virtual MIDI client must first be created. The MIDI client represents a MIDI session object to declare the application’s participation in OS X’s MIDI services. The application required the ability to send MIDI data, so a MIDI source object was created to transmit necessary MIDI values. MIDI Destinations connected to the OS X device can
request a list of MIDI sources from the MIDI server. The server returned this MIDI source and its given name. The MIDI source used was named “Thesis” and appeared as “Thesis” to a MIDI destination. The MIDI data transmitted by the application was of the type Continuous Control or CC. Continuous control messages are typically used for analog style inputs such as potentiometers. CC allows for the parameter’s value to be updated by sending a new value to the appropriate CC channel and controller. Note on messages are not appropriate as they do not allow for the parameter’s value to be changed. Note on messages simply transmit a binary on/off message as opposed to a value message. A CC packet requires the information channel, controller number, and controller value to be transmitted over the server [MMA, 2014]. The MIDI channel information is a number over the range 1-16 that is used by the MIDI Destination to determine if it is interested in the MIDI CC signal. The MIDI destination can subscribe to a particular channel of interest, and ignore messages on other channels. The MIDI controller number is a number from 0-119 that tells the MIDI destination which parameter the value is intended for. The MIDI destination assigns a UI or hardware parameter to a controller value. The parameter is only updated when a MIDI message with the correct controller value is received. The controller value is a 7-bit number over the range 0-127 with the new value for the provided controller number. In this application the channel was the same for each CC message as only 3 CC control numbers were required. MIDI channels can be beneficial when a MIDI device desires to
transmit to more controls than the allotted number of control numbers.

Figure 13: MIDI signal flow within Ableton. This shows all MIDI messages used within the experiment.

Fitts Test Implementation

The auditory Fitts Test resided within the OS X application. The program must be capable of 1) being completed without the presence of the test proctor, 2) generating and randomizing test conditions for a double blind test, 3) tracking the number of completed tapping events, and 4) recording the test results for further analysis.

The test can be fully completed by the test taker by setting up MIDI messages with the DAW to turn audio on and off at appropriate times. The audio play and audio stop buttons within Ableton Live were triggered by the OS X application by MIDI messages. When the test taker pressed the Begin Test button, a MIDI message was sent to Ableton instructing it to commence audio playback. Upon completing the test another message was sent to Ableton
instructing it to turn audio playback off. These audio queues alert the subject that a trial has started and ended.

The test timer did not begin immediately upon pressing the *Begin Test* button. *Begin Test* simply started the audio and alerted the software to wait on the first two correct mapping. This ensured that the subject had placed their attention onto the MAPR, and allowed both destination tones to be presented to the user before the test timer starts. The test timed itself for thirty seconds after receiving the first correct mapping, and then turned the audio off. At this time the test counter was incremented.

A test count was shown on the OS X application to let a user keep track of how many trials they had completed with a particular controller. The user could then begin the next test at their convenience by pressing the *Begin Test* button. These test features allowed the proctor to only be present at the beginning of the testing process to configure the first MAPR, and in the middle of the testing process to configure the second MAPR. The test taker was alerted when they had completed the testing for the first device and to locate the proctor for further instruction. At this time the proctor configured the second device and Ableton for the second round of testing and then left the room.

The test generated and randomized test conditions so that the proctor was unaware of the test condition. This was accomplished by randomizing the order in which the trials were administered. The test conditions for MIDI width and MIDI distance were placed into an array in order, and the array was shuffled.
These variables affected the tests’ index of difficulty. An alternative method to randomizing the test is to generate random variables for MIDI width and MIDI distance for each trial. Generating width and distance using uniform random variables results in an uneven distribution of ID. This causes some users to not take tests of some difficulty due to low trial count per subject. This issue was circumvented by standardizing width and distance across the population. This method has previously been used by Fitts[Fitts, 1954], Mackenzie[MacKenzie et al., 1991], Epps[Epps, 1986], and Ware and Mikaelian[Ware and Mikaelian, 1986]. This method provided the additional advantage of allowing movement times to be directly compared.

The distance was used to calculate MIDI low and MIDI high variables. MIDI low and high were centered around MIDI CC value 49. This value was selected to center the test around pitches that are more recognizable. Extreme low and high MIDI CC values become harder to differentiate. These extremes are caused by MIDI CC values close to the bounds of 0 and 127. The low and high variables were the two tracking variables that the test taker must acquire with the MIDI devices. MIDI width variables used are [2 4 8 16] and MIDI distances used are [16 32 64 90]. Researchers calculated these numbers by first determining the lowest width, and then multiplying by 2 to create the next member of the width set. This general case produces a width set of \([min \times 2 * min \times 4 * min \times 8 * min]\), where \(min\) is the minimum value of the width data set. The largest number of the width set is then used as the minimum value of the distance set. This
condition guarantees a minimum ID of 1. The general case for the distance set becomes \([8 \times \text{min } 16 \times \text{min } 32 \times \text{min } 64 \times \text{min}].\) Creating the test variables in this fashion results in a maximum test difficulty of 7. The variables used in this research varied slightly from the general case as the final value for MIDI distance was not 128. 128 was not an acceptable distance as this is the maximum range of 7-bit MIDI CC values. A distance of 128 did not allow width to be considered properly. An alternative data was reached by setting the minimum width to 1. The variables generated were width = \([1 \ 2 \ 4 \ 8]\) and distance = \([8 \ 16 \ 32 \ 64]\). These test conditions were considered, but found to be inadequate in the test environment. Due to the nature of Simpler’s Transpose parameter, 2 adjacent MIDI CC values could return the same pitch. This is confusing when testing as the tone generator may be properly tuned, but the test does not proceed to the next target tone. The test set described with 90 as the maximum test distance resulted in a maximum ID of 6.5 when paired with the minimum width of 2.

In a typical visual Fitts Test, the tapping events are completed by the user clicking a button. This method for event acquisition was unsuitable for the audible test as MIDI CC messages are sent without ever using a button. If the acquisition was counted as soon as the user enters the MIDI range for acquisition, this would allow users to complete the test too easily. They would simply have to move the device back and forth without using their ears. Ware and Mikaelian considered fixation time as a proper alternative to hardware tapping\[Ware and Mikaelian, 1986\]. To create an auditory tapping event, the test
taker must hover the data point they are acquiring for a period of time. This ensures that the user is attempting to input their MIDI value as the target, and the target value was not happened upon by chance. Ware and Mikaelian asserted that a fixation time of 0.4 s is an appropriate time interval [Ware and Mikaelian, 1986]. To measure fixation time, BLE packets were counted over a span of time as the correct data value was hovered. If the user exits the boundaries of an acquisition, the packet count was reverted to 0 and the test taker must reacquire the target for the correct amount of BLE packets. At a packet rate of 50 Hz, 20 packets equated to 0.4 seconds. Once 20 packets had been acquired within the acquisition range, the targets acquired was incremented and the new target tone was played. This method resulted in the fixation period being measured from the device side.

The OS X application recorded test results to a single comma separated values (CSV) file for later analysis. The data points recorded included test type 'Slider' or 'Motion', MIDI low destination, MIDI high destination, MIDI width, and targets acquired. The evaluation was completed using Matlab.
Testing and Evaluation

The testing was conducted to determine if the potentiometer and motion devices were significantly different in terms of efficiency by evaluating IP scores of the two devices. If one device has a significantly higher IP over another device, it suggests that the device is more efficient. In developing the test, visual Fitts test procedures were examined to determine a procedure. These tests are outlined in chapter 2. The procedure in this research most closely resembled Fitts’ original tapping tests where the test time is the independent variable and completed taps is dependent.

The subjects were present for the test set-up and configuration of the first MIDI device. The test was prepared by launching the OS X application, connecting the devices over BLE, selecting the device to be used first, ensuring the correct Ableton scene was selected, and providing the test instructions to the subject. The test instructions are shown in figure 14. The first device for each successive subject was alternated to eliminate test learning bias. The subjects were verbally instructed to adjust the volume as necessary using the keyboard volume inputs, and to keep from jarring the motion device as much as possible. The motion device was configured to be highly sensitive, and quick jolts such as setting the device down can disrupt the orientation algorithm. The algorithm would realign after the jolt had concluded, but the quick jump in MIDI value
may be confusing to the user and perceived as a malfunction. The test instructions suggest that the motion device would be demonstrated by the instructor. This was completed by manually starting the audio within Ableton and demonstrating the axis of rotation to the subject. Finally the subjects were asked if they had any questions and then allowed to proceed with the testing.

When 16 trials had been completed, the *Begin Test* button was disabled and an alert message appeared display the text, "You’re through with this device. Come find me!". A practice block was not incorporated to reduce testing procedure length, and to reduce familiarity with the test procedure. Fitts also suggests that practice trials will hold no effect on the outcome of the testing[Fitts, 1954].

Figure 15 shows the MIDI devices presented to the subjects. Each subject used the same devices. The potentiometer used was not encased like a typical potentiometer MIDI controller, and the impact of this discrepancy was not accounted for. The rotary potentiometer also did not feature a rubber cap on top of the rotating portion. This dramatically decreases diameter of the knob. This may cause finer tunings to be harder to complete. Varying the diameter of the potentiometer may reveal some efficiency gains with either a smaller or larger diameter. This also shows a key benefit of rotating the motion controller as the diameter of the axis of rotation can be changed by the user simply by moving it differently. A rotary potentiometer was the best fit for this experiment as the motion exerted on the device is rotational and very similar to the motion used for the motion controller. This makes the comparison more compelling as the
MIDI Device Testing

Today we will be testing two MIDI input devices. One requires you to turn a potentiometer to send MIDI messages, and the other is a motion controlled MIDI device. The motion device is controlled by rotating it about an axis. The correct axis of rotation will be demonstrated prior to using the motion MIDI device.

Press the start button to begin a test. Pressing this button will turn on the audio for the test. You will hear a tone generator in your left ear that is tunable by the MIDI device provided. In the right ear you will hear a stationary tone generator. The objective is to tune the controlled tone to the stationary tone. When you have correctly tuned the tone generator, the stationary tone will change to a new pitch. The first two times this is accomplished are not counted as part of the test, as they are to let you become familiar with the two destination tones. A diagram below demonstrates the test.

After the second tuning has been completed a 30 second timer will begin. After the timer begins your objective is to correctly tune the tone generator as many times as possible in the time allotted. When the 30 seconds is up the audio will turn off and you have completed a trial. The screen will increment the “Tests Completed” field to show that you have completed a trial. You must complete 16 trials with each MIDI device. Some tests will feel very easy and some will be quite difficult.

Upon completing 16 tests come and find me and I will set up the other device for you.

Figure 14: Test instructions that the subjects were presented with before taking the test.
exerted motion was similar for both devices.

![Motion and potentiometer MAPRs cased for testing. These MAPRs were used by all subjects.](image)

Figure 15: Motion and potentiometer MAPRs cased for testing. These MAPRs were used by all subjects.

Eight subjects were tested over the course of two days. Each testing procedure required roughly 25 minutes. Subjects were students and faculty at the University of Miami. Subjects were allowed to perform the experiment with either hand, and were allowed to use the motion device in any way that they felt appropriate.
5

Results

The test results were first validated before further analysis. A t-test was conducted on the average IPs of each subject to check result certainty. A p-score of .05 was desired to achieve 95% confidence that the results were not due to chance. The subject data was loaded into Matlab after every test and the p-score was calculated. The progression of the p-score as subjects were added is shown in figure 19. The p-score decreased to .04 after testing 8 subjects. The p-score was monitored until this p-score was achieved, and the trend could be said with confidence. The inputs and results of the final t-test are shown in table 4 and 5. It can be show with 95% confidence that the IP of the motion device was higher than the potentiometer device. The average IP of the potentiometer device and motion device was 1.4094 and 1.8213 respectively. A higher IP equates to a better performance in terms of bits/sec. The confidence interval lied between the values -0.8079 and -0.0159 bits/sec.

<table>
<thead>
<tr>
<th></th>
<th>Potentiometer Device</th>
<th>Motion Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean IP</td>
<td>1.4094</td>
<td>1.8213</td>
</tr>
<tr>
<td>Std Dev</td>
<td>0.3308</td>
<td>0.5622</td>
</tr>
</tbody>
</table>

Table 4: Average Controller Performance

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>P score</td>
<td>0.0435</td>
</tr>
<tr>
<td>Confidence Interval</td>
<td>-0.8079 -0.0159</td>
</tr>
<tr>
<td>Sample Size</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 5: Statistical Testing Results
Potentiometer data for all subjects is shown in figure 17 and motion data is shown in figure 16. Individuals’ regressions are shown in gray and the average is displayed in red. The averages from both graphs are compared in figure 18. In figures 17 and 16 individual data points for each subject are shown where each unique color represents a different subject. Subjects indicated that it was not the device that made the test hard, but the task they must complete.

These plots show every subject had a positive regression slope for each device. This shows that the ID of the tests provided an increase in difficulty as described by Fitts’ laws, or in other words, that increasing distance or decreasing width created a more difficult test. This finding validates Fitts’ law as an appropriate evaluation tool for this research. Outlying data points at a MT of 30 s represent tests when the user was able to correctly identify the destination tone only once during a trial. This suggests that pitch may also have some bearing on the difficulty of a trial. This could be counteracted in future test implementations by being more selective in how the MIDI mapping is completed.

The regression comparison shows an intersection of the motion and potentiometer regressions around a difficulty of 2.5. This suggests that simple tasks are easier to complete with the potentiometer device, and more difficult tasks are more efficient on the motion controller. Subjects reported completing easier tasks without having to use their ears. This is due to the MIDI width variable being so large that the subject did not have to accurately tune the tone generator. Even though the subjects report not using their ears, they were still
using audio queues to attempt tracking to each destination tone. When the
destination tone switches, the user registers this change and begins tuning in the
opposite direction. These tests are almost binary in nature.

![Motion Fit and MT/ID for all Subjects](image)

**Figure 16:** Trial Data and Regressions for all Subjects Using the Motion Controller.

The progression of IP and p-score as subject count increases is shown in
figure 19. The p-score was observed after every subject, and testing concluded
when the 95% certainty threshold was reached. The IP of the potentiometer
device was trending negatively when testing concluded. Performance with the
motion controller was relatively stationary throughout the study.
Figure 17: Trial Data and Regressions for all Subjects Using the Potentiometer Controller.
Figure 18: Average Regressions for Potentiometer and Motion Controller.
Figure 19: Progression of IP and P-score as Subject Count Increases.
The average movement time for all subjects is shown in figure 20. The potentiometer device had a mean movement time of 4.4 for all trials, and the motion device has a mean movement time of 2.8. This shows that on average 1.6 seconds can be saved per mapping.

Figure 20: Average MT of all Subjects.
Discussion

6.0.4 Conclusion

The evaluation determined the alternative hypothesis valid; that there is an increase in performance when using the motion controller. The motion controller offers an efficiency advantage over the potentiometer controller. The motion controller offers other advantages to musicians that may not be apparent from the test. The motion controller can potentially free up the musician’s hands to perform other tasks. The motion device can work on the back of the hand or thumb to allow the musician to strum a guitar or play a keyboard in between mappings. A musician knowing they can spare seconds in a real time performance environment would persuade some to make the switch form a traditional, potentiometer MIDI controller to a motion MIDI controller. A more efficient controller promotes creativity as the technology is less in the way of the creative process. The motion controller also does not limit the degree of expression as the two controller types offer the same data ranges.

6.0.5 Variation from Visual Tests

Visual Fitts tests have some advantages over auditory Fitts tests. Before the visual test has begun the subject has already become familiar with the distance and width of the target platforms, and can reference these values as they deem appropriate. In the auditory Fitts test, only one destination is presented at a time. This presents the need to allow the subjects to hear the destinations
prior to beginning each trial.

In a visual test the distance metric is linear and is observed linearly. The spatial positioning of the destinations also does not affect the outcome of the test as a straight line in space can be represented with any offset without effecting the test procedure. The audio test’s distance possesses a frequency component where offset can change the test environment. The offset can be distributed to both destinations by spacing them around a center point as explained in Chapter 3. The alternative method is keep one destination stationary throughout the tests and place the entire offset onto a single destination. This method lends to allowing the subject muscle memory of the location of the destination without an offset. This could negatively impact the test as the subject’s will use less pitch recognition to tune the oscillator. Muscle memory comes into play in the suggested method, but is far less impactful as each trial requires new muscle memory. A few subjects realized they could use muscle memory to help complete the test and alerted me of this discovery.

Mean movement times in visual Fitts tests are often reported to be below 1 second [MacKenzie et al., 1991]. In this research the average movement times of 4.4 s and 2.8 s suggest that the auditory test is more difficult than the visual tests. Mackenzie’s reporting of below 1 second includes tests with a higher index of difficulty than used in this study. The Fitts model for human motor systems is still represented accurately despite this discrepancy. It may also be argued that the 0.4 second fixation time enlarges the movement times as it reduces the
potential to complete mappings within the trial time. This hinderance does not take away from findings that the motion controller is more efficient.

6.0.6 Future Work

The one dimensional auditory test was a good determinant for the ability of auditory feedback to serve as the basis for a Fitts test. This shows that Fitts model for human motor function is not dependent on task sensory input. This serves as a basis for determining the efficiency of the motion controller in a real time control environment.

This test does not fully demonstrate the efficiencies of a motion device as many more motions and gestures can be completed with the device. The next logical step is to extend the test to two or three dimensions to utilize more axes of rotations. The testing devices would include a potentiometer MIDI device that has three potentiometers connected, and the same motion device used in this research. A joystick could be used to complete a 2D test, considering it allows the subject to interact with a single interface. The magnetometer would need to be included to report an accurate yaw value if conducting a 3D test. The test would require two or three unique parameters to be tuned. This could include tuning a digital delay’s delay time, a resonant filter’s cut off frequency, and the tone generator used in this research. Tuning three parameters creates complications for each device. The potentiometer device requires the user to lift their hands from one potentiometer and onto another. It also requires them to remember which potentiometer is for which parameter. The motion device requires users to
find an exact orientation in 3D space to map all three parameters at once.

Tuning a second parameter requires the orientation of one axis to be maintained while the second axis’s placement is determined. Completing a 3D task with a joystick requires an additional joystick or a single potentiometer.
LIST OF REFERENCES


