

BACHELOR'S THESIS

Bachelor's Degree in Biomedical Engineering

GAIT TRAINING WITH HAPTIC FEEDBACK ASSISTANCE



Report

Author: Paula Simó Higuera
Advisor: Professor Voloshina
Co-Advisor: Professor Reinkensmeyer
Director: Raul Benítez Iglesias
Quarter: Summer 2022

ABSTRACT

This work serves as a proof of concept for developing a haptic feedback hip belt to encourage gait exploration. Current rehabilitation and assistive device training relies on repeating consistent movements. In contrast, past work has shown that training with increased variability can improve motor learning and performance of new tasks. Consequently, it is possible that training users to explore new movements would be beneficial to their adaptation to assistive technology. The wearable device presented in this thesis relies on an Inertial Measurement Unit to track the location of the user and direct user movements through vibrotactile feedback. This work serves as a basis for further development of a device that will guide users to explore a more diverse range of motion during gait training. Such a device could help people with disabilities explore their motor control pathways and more effectively adapt to novel tasks or wearable assistive devices (such as exoskeletons).

RESUM

Aquest treball serveix com a prova de concepte per desenvolupar un cinturó basat en retroalimentació hàptica per fomentar l'exploració de la marxa humana. La rehabilitació actual i l'entrenament de dispositius d'assistència depenen de la repetició de moviments invariables. Contràriament, es troben estudis que demostren que entrenar amb una major variabilitat pot millorar l'adquisició d'aprenentatge motor i el rendiment de l'individu al realitzar noves tasques. En conseqüència, és possible que entrenar als usuaris per explorar nous moviments sigui beneficiós per la seva adaptació a la tecnologia assistida. El dispositiu presentat en aquest treball de fi de grau utilitza una unitat de mesura inercial (IMU) per fer un seguiment de la localització de l'usuari i dirigir els seus moviments mitjançant retroalimentació vibrotàctil. Aquesta investigació serveix com a base per futurament crear un dispositiu que guiarà als usuaris a explorar un rang de moviment més divers durant l'entrenament de la marxa humana. La versió futura del dispositiu podria ajudar a persones amb discapacitat a explorar les seves vies de control motor, aconseguint així que es puguin adaptar més fàcilment a noves tasques o dispositius d'assistència com per exemple exoesquelets.

RESUMEN

Este trabajo sirve como prueba de concepto para desarrollar un cinturón basado en retroalimentación háptica para fomentar la exploración de la marcha humana. La rehabilitación actual y el entrenamiento de dispositivos de asistencia dependen de la repetición de movimientos invariables. Por el contrario, existen estudios que demuestran que entrenar con una mayor variabilidad puede mejorar la adquisición de aprendizaje motor y el rendimiento del individuo al realizar nuevas tareas. En consecuencia, es probable que entrenar a los usuarios para explorar nuevos movimientos sea beneficioso para su adaptación a la tecnología asistida. El dispositivo presentado en este trabajo de final de grado utiliza una unidad de medición inercial (IMU) para realizar un seguimiento de la localización del usuario y dirigir sus movimientos mediante retroalimentación vibrotáctil. Esta investigación sirve como base para futuramente crear un dispositivo que guiará a los usuarios a explorar un rango de movimiento más diverso durante el entrenamiento de la marcha humana. La versión futura del dispositivo podría ayudar a personas con discapacidad a explorar sus vías de control motor, consiguiendo así que se puedan adaptar más fácilmente a nuevas tareas o dispositivos de asistencia como por ejemplo exoesqueletos.

ACKNOWLEDGMENTS

I would like to thank my advisor, Professor Voloshina, for her constant support, her guidance, her excellent suggestions, her help, and her insights during the development of this thesis. I am delighted to have the opportunity to continue my academic journey in her Rehabilitation and Augmentation Lab. I am looking forward to learning as much as possible from her during the upcoming years.

I would like to express my gratitude to my co-advisor, Professor Reinkensmeyer, for the helpful and captivating discussions that served to shape this project and to give me the opportunity to join the meetings of the BioRobotics Lab. I am also thankful to all the students of his laboratory for their willingness to help me.

I would like to thank Ph.D. candidate Chi-Shih Jao and Ph.D. candidate Eudald Sangenís from the Microsystems Laboratory for their help in understanding the concept of Inertial Navigation Systems and debugging the code of the Inertial Measurement Unit. Their guidance was a key factor to develop this project.

I am also thankful to the Balsells Foundation for offering me the opportunity of studying and continue my academic path at the University of California Irvine thanks to a Balsells Fellowship.

I would like to also express my gratitude to my home university, Universitat Politècnica de Catalunya, for providing me with the necessary tools to become a biomedical engineer.

I am also grateful to all the scientific-technological community that works every day to try to leave the world better than how they found it. All that we know today is the result of the efforts of previous generations. Thank you for your work.

Last but not least, I would like to deeply thank my family and friends for their unconditional love and support. I would especially like to thank my parents for always prioritizing my education and my happiness, to my sister for guiding me in the hardest decisions, and to my partner for accompanying me every day.

TABLE OF CONTENTS

ABSTRACT	i
RESUM	ii
RESUMEN	iii
ACKNOWLEDGEMENTS	iv
LIST OF TABLES	vii
LIST OF FIGURES	vii
LIST OF ABBREVIATIONS	viii
1 INTRODUCTION	9
2 BACKGROUND	10
3 THESIS SCOPE	12
3.1 Project justification	12
3.2 Project scope	12
3.3 Gantt diagram	13
4 METHODS AND MATERIALS	14
4.1 Device description	14
4.1.1 List of materials and cost	14
4.2 Hardware	15
4.2.1 Arduino Nano 33 IoT	15
4.2.2 LSM6DS3 (6 Degree of Freedom Inertial Measurement Unit)	16
4.2.3 TCA9548A I2C Multiplexer	17
4.2.4 Adafruit DRV2605L Haptic Motor Controller	17
4.2.5 Motor disks	18
4.2.6 PowerBoost 1000 Charger - Rechargeable 5V Lipo USB Boost @ 1A - 1000C	18

4.2.7	Lithium-Ion Polymer Battery - 3.7v 2500mAh	18
4.3	Inertial Navigation Systems	19
4.3.1	IMU Data processing	19
4.4	Vicon cameras as ground-truth	22
4.5	Communication Protocols	23
4.6	Programming tools	24
4.6.1	Arduino Code	24
4.6.2	Matlab	25
4.7	Belt prototype	27
4.7.1	Schematics diagram	28
5	RESULTS	29
6	LIMITATIONS AND FUTURE DIRECTIONS	32
7	CONCLUSION	33

LIST OF TABLES

1	Materials and cost.	14
---	-----------------------------	----

LIST OF FIGURES

1	Tasks list and duration.	13
2	Gantt diagram.	13
3	Arduino Nano 33 Iot real picture of the board	15
4	Arduino Nano 33 IoT pinout scheme	16
5	Pin description of LSM6DS3	16
6	Picture of the TCA9548A I2C Multiplexer	17
7	Picture of the DRV2605L	17
8	Picture of the motor disks	18
9	Picture of PowerBoost 1000 Charger	18
10	Picture of lithium-ion battery	18
11	Inertial Frame, Earth-Centered Earth-Fixed Frame, and Local Navigation Frame	20
12	Body Frame with respect to fixed frame	20
13	Local Navigation Frame mechanisation	22
14	Setup of Vicon cameras	23
15	Electronic diagram.	27
16	CAD of the encapsulation of the motor disks and motor controllers.	28
17	Schematics of the design.	28
18	Arduino Nano 33 IoT attached to the foot.	29
19	Comparison between 3D position of three steps recorded by different systems.	29
20	Comparison between 2D position of three steps recorded by different systems.	30
21	Comparison between velocity recorded by different systems.	30
22	Trajectory in N-frame followed by a participant turning three times to the left.	31
23	Belt prototype.	31

LIST OF ABBREVIATIONS

OPTIMAL	Optimizing Performance through Intrinsic Motivation and Attentional Learning
UART	Universal Asynchronous Receiver/Transmitter
VR	Virtual Reality
GMP	Generalized Motor Program
3D	Three-dimensional
IMU	Inertial Measurement Unit
ZUPT	Zero Velocity Update
BLE	Bluetooth Low Energy
USB	Universal Serial Bus
I-frame	Inertial Frame
E-frame	Earth-Centered Earth-Fixed Frame
N-frame	Local Navigation Frame
INS	Inertial Measurement Unit
GPS	Global Positioning System
EKF	Extended Kalman Filter
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
IP	Internet Protocol address
SPI	Serial Peripheral Interface
I2C	Inter-Integrated Circuit
SPI	Serial Peripheral Interface
SSID	Service Set Identifier
SDA	Serial Data Line
SCL	Serial Clock Line
LiPo	Lithium-Ion Polymer
LED	Light Emitting Diode
ID	Identification
PC	Personal Computer
GND	Ground

1 INTRODUCTION

Adults with motor impairment often have difficulty learning new motor tasks, such as re-learning how to walk or adapting to an assistive device. Similarly, neurologically healthy individuals may need up to several days of training to learn more complicated tasks, such as learning how to use exoskeletons effectively (e.g. [1, 2, 3]). This is potentially a result of assistive device training being invariable. For example, training with assistive devices usually involves continuous walking on a treadmill with very limited gait variability. Similarly, traditional rehabilitation after motor impairment tends to involve sessions that rely on performing invariable and uniform tasks.

In contrast, past work has shown that training with variability can be beneficial to restoring, at least partially, the range motion of a person with motor impairment [4]. Moreover, toddlers learn how to walk by performing variable motions, such as taking more steps, falling less, or traveling further distances [5]. This exploration is then likely transferred to performing new tasks and building up new motor control frameworks. It is possible that adding gait variability training during rehabilitation or training with assistive technology would be beneficial for shortening recovery periods and creating more effective training protocols.

In addition to training with variability, providing biofeedback during gait training has also been shown to be beneficial to learning [6, 7]. For example, visual feedback provided with Virtual Reality (VR) was beneficial for interacting with robotic-assisted gait training devices such as Lokomat, a robotic treadmill system [6, 7, 8]. The main benefits were faster adaptation to the robot, training motivation increase, higher level of concentration, and reduction of the drop-out rate. Moreover, haptic feedback has successfully been used to enhance other aspects of gait such as balance, velocity, or step length (e.g. [9, 10]). For example, vibrotactile feedback applied to the trunk could increase confidence and the feeling of balance of participants [9].

To this end, we decided to develop a device to encourage gait exploration during training based on haptic feedback technology. This work is the first step for further developing a device that will provide assistance in gait variability and force people to make more variable movements. The device will be used to construct a novel training approach, that could supplement traditional gait training, or training with an assistive device.

2 BACKGROUND

Gait rehabilitation training, as well as training with assistive devices, traditionally relies on performing repetitive, invariable movements. However, learning motor tasks by performing invariable and uniform tasks often leads to slow and ineffective learning. Creating new training protocols to fasten the recovery periods on clinical rehabilitation and quicken the adaptation to assistive devices could lead to improved patient well-being, optimization of economic resources used for the treatment, and an increase of caregivers and hospital availability to attend to new patients. To improve motor learning, there are several aspects to consider.

For one, people tend to learn better when using variable movement patterns. For example, children perform variable motions while learning how to walk [5]. Knowledge acquired during exploration is then transferred to performing new tasks [5]. Specifically, gait development in children is characterized by six stages: infant stepping, inactive period, supported locomotion, unsupported locomotion, mature similar gait and mature gait [11]. During first periods the infant presents variability in all the gait parameters, imbalance and difficulty in maintaining posture [12]. Later, with the development of free walking, the activity pattern becomes more organized. There are several exercises that help with the learning process, such as practicing the stepping reflex (newborns try to take steps when someone holds them and let their soles touch a flat surface), walking while avoiding obstacles, cruising (placing objects in different parts of the room and moving towards them), and walking using a towel for support. Such exercises are highly variable and help children develop new motor pathways by exploring their gait. A similar gait exploration training approach could also be conducted with adults.

Furthermore, practicing diverse motions helps to learn new motor tasks. One motor learning theory suggests that a movement pattern is generated through a set of previously defined motor commands, or a generalized motor program (GMP). Our memory then recalls these predefined commands and adapts them to a particular situation [13, 14]. To produce a movement, humans need to know the initial conditions (where is the task performed) and the response specifications (what is expected). Once those are defined, the person can perform the action. Later, the human brain analyzes the sensory consequences (how did it feel) and the response outcome (analyze if the task was performed correctly). By evaluating the results the person will know how to move or adapt their motions in a similar situation in the future. Practicing diverse motions provides motor information that will help to refine the movement pattern that a task follows. Thus, it is reasonable to believe that training with variability while learning how to use assistive devices, would improve learning. Moreover, there are some studies that show that training with augmented variability in adults can partially restore upper-limb motor control pathways in stroke patients [4]. A similar approach could be helpful for lower-limb clinical rehabilitation.

Additionally, people learn motor tasks by correcting previous errors. Another motor learning theory suggests that to perform a task, humans execute smaller movements at first to achieve their goal. Aided by perceptual feedback (perspective the individual has about their motion), they can then refine their movement to produce more reliable and efficient motion [15, 16, 17]. Thus, motor learning is partially achieved through optimization of the correction of previous errors. Letting people explore new motions could help to make them aware of the errors they make while learning a new task. Consequently, they could detect their own errors and learn how to negate them.

Appropriate motivation also has a significant effect on the success of motor learning. Specifically, the OPTIMAL motor learning theory suggests that learning is highly dependent on motivation and

attention during task performance [17, 18]. It suggests that people learn more efficiently when their expectancies of success while performing a task are high, and when they have autonomy to control their movements. These factors increase their motivation and attention to perform the task. For example, one study showed that integrating an avatar game based on VR biofeedback could help to maintain motivation in gait therapy leading to considerable improvements in several aspects of gait in children with cerebral palsy [19].

Finally, motor-learning can be augmented through biofeedback. Haptic feedback, in particular, has been shown to be effective in medical care to replace (for people with total sensory impairment), augment (for rehabilitation), or train (for patients without impairment) sensory capabilities. Haptic systems are those which can impact the sense of touch, including temperature, vibration, texture, slip, pain, force, and proprioception sensations [20]. Usually, haptic devices are used to create a more realistic experience. They can be used to transmit missing information, transmit behavioral cues, or train by guiding or correcting movement patterns [20]. In the context of balance and motor movement, haptic feedback is especially effective for augmentation. For example, attaching tactors to the waist and using them for feedback on torso movement, can significantly improve postural stability [21].

It is likely that people learn as a combination of experiencing biofeedback, variability, motivation, previous errors correction, and the ability to explore movements independently. And, although biofeedback has been used in the past to improve learning, to our knowledge, it has not been used to motivate gait exploration and training with variability. In turn, haptic feedback could be useful in encouraging users to vary their gait during training. For this reason, we aim to design and construct a haptic hip belt to use it as a feedback mechanism to motivate gait variability. A device that provides motor suggestions while still allowing the user to explore and make errors, could decrease the time of motor learning while improving overall performance. Such a device could lead to assisting many people in adapting to their assistive devices, improving gait performance, and consequently, improving their quality of life.

3 THESIS SCOPE

3.1 Project justification

The main goal of this project is to show a proof of concept for a device that could improve gait rehabilitation training and training with wearable assistive devices. Potentially, the device will help to create a new training protocol that reduces the time required to learn or relearn motor tasks. The use of our device will be beneficial for patients who suffer from different mobility limitations. For example, people who need rehabilitation after surgery, people learning how to use an assistive device such as an exoskeleton, people relearning how to walk after a neurological impairment, etc. This project hypothesizes that encouraging patients to walk variably will allow participants to practice new movements and consequently transferring the knowledge acquired to face new tasks such as walking through uneven terrain, walking backwards or walking uphill. Thus, the main concern of this report is to answer the question: How can we most effectively enforce gait exploration during training?

Our approach consists of encouraging people to explore variability, particularly concerning movement direction and movement speed. This report aims to summarize the main steps needed to create a device with these characteristics.

3.2 Project scope

This project focuses on designing and programming a device that tracks the location of a person by recording data from an Inertial Measurement Unit placed on the heel. The device also contains two vibration motors located on each side of the belt also worn by the user. Every time the left motor induces a vibration, it indicates to the user to turn left. Every time the right motor vibrates it indicates the person to turn right. When a vibration starts, we define a vector that describes the initial direction of movement. A second vector, which is constantly updating throughout the task, defines the current direction of movement. By analyzing the angle between the first vector and the second vector, we can identify if the wearer has turned following the desired path. If the individual has achieved performing the task correctly, the device stops vibrating.

This report contains a preliminary design of the wearable device. The current device serves as a proof of concept for further developing a device that will encourage participants to perform several motions. The main topics the report covers include how to integrate inertial navigation systems into a product to achieve tracking a person and how to synchronize feedback with real-time movement.

3.3 Gantt diagram

We decided to organize our work defining tasks and deadlines (Fig. 1). We followed the timeline shown in the Gantt diagram (Fig. 2).

ACTIVITIES	Start date	Duration	End date
1. Read papers related to the topic and create a lab notebook	2/1/22	20	2/21/22
2. Design the device	2/20/22	10	3/2/22
3. Write a project draft	3/3/22	5	3/8/22
4. Choose and purchase materials	3/28/22	10	4/7/22
5. Learn about Inertial Navigation Systems (INS)	3/30/22	65	6/3/22
6. Create a code for processing Inertial Measurement Unit (IMU) data	4/25/22	50	6/14/22
7. Control haptic motor controllers DRV2605L with multiplexer TCA9548A	4/5/22	12	4/17/22
8. Create a closed-loop communication Arduino-Matlab via serial	5/14/22	7	5/21/22
9. Create a wirelessly closed-loop communication	5/26/22	7	6/2/22
10. Define automatic end and start point of vibration	5/20/22	15	6/4/22
11. Create CAD and 3D print components	5/30/22	4	6/3/22
12. Assemble the device	6/10/22	2	6/12/22
13. Write the report	5/15/22	40	6/24/22

Figure 1. Tasks list and duration.

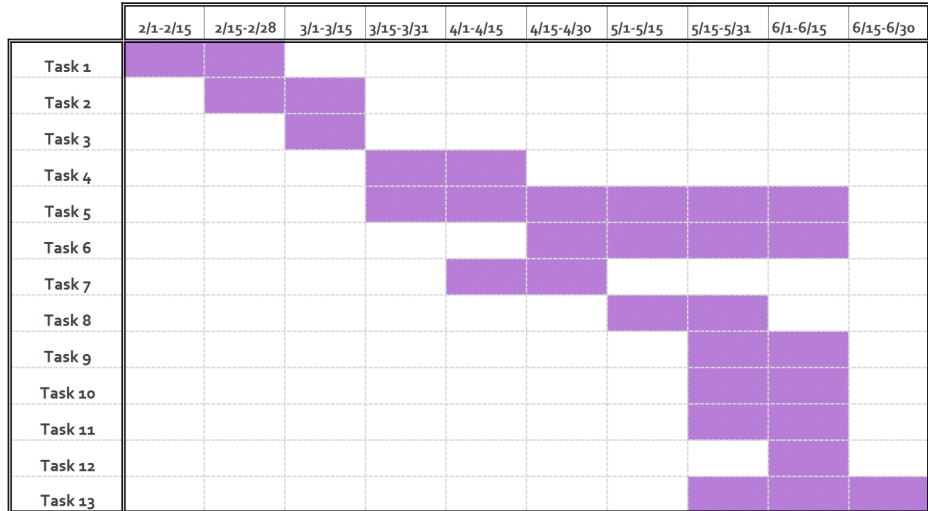


Figure 2. Gantt diagram.

4 METHODS AND MATERIALS

4.1 Device description

The device consists of a hip belt that gives instructions by vibration. The location of the actuators corresponds to the left and right sides. Every time the left component vibrates, it indicates to the person to turn left. Every time the right component vibrates, it guides the person to turn right. The vibration stops every time the task has been performed correctly. The device tracks patients performing tasks using an Inertial Measurement Unit placed on the heel of the participant. If the sensor detects that the patient has moved in the correct direction, the motor disk stops vibrating.

The electronics biofeedback components of the system consist of two Adafruit DRV2605L Haptic Motor Controllers that can direct vibrations at different speeds to the two vibration motors. Each motor controller controls one motor disk. An Arduino Nano 33 IoT board, which incorporates an Inertial Measurement Unit LSM6DS3, detects the accelerations and angular velocities of the participant during walking and dictates when vibration is to occur. To connect the motor controllers to the board using the same I2C connection, a TCA9548A I2C Multiplexer is used. A custom-made Matlab algorithm determines the position and orientation of the individual. By the implementation of an algorithm created in Matlab, we can obtain the position and orientation of the person in the space. A lithium-ion polymer battery - 3.7V 2500mAh powers the board due to the connection of a PowerBoost 1000 Charger - Rechargeable 5V that converts the 3.7V offered by the battery to 5V, which is the voltage required to power the Arduino Nano 33 IoT through the USB port.

A strap that acts as a hip belt incorporates the electronics. A 3D printed case packs each motor controller with its motor disk. The cases are sewn to the strap. The back of the belt contains the battery system and the multiplexer. A cable running from the back of the belt to the heel of the foot connects the microcontroller and the Inertial Measurement Unit to the whole system.

4.1.1 List of materials and cost

List of costs			
Product	Cost/Unit	Quantity	Cost
Arduino Nano 33 IoT	\$18.5	1	\$18.5
TCA9548A I2C Multiplexer	\$ 6.95	1	\$ 6.95
Adafruit DRV2605L Haptic Motor Controller	\$ 7.95	2	\$15.9
PowerBoost 1000 Charger-Rechargeable 5V Lipo USB	\$ 19.95	1	\$ 19.95
Lithium-Ion Polymer Battery - 3.7v 2500mAh	\$14.95	1	\$ 14.95
Breadboard	\$3.75	1	\$3.75
Motor Disk	\$1.95	2	\$1.95
Wires	\$5	1 pack	\$5
Total cost	-	-	\$88.9

Table 1. Materials and cost.

4.2 Hardware

4.2.1 *Arduino Nano 33 IoT*

Several factors dictated the selection of Arduino Nano 33 IoT (Fig. 3) as the microcontroller of our project. First of all, we looked for a device that could be programmed with Matlab and C++. Arduino is always coded in C++ and Matlab is the programming language used to control the exoskeletons of our laboratory. We wanted to code everything in the same language to be able to interact the systems in the future. Secondly, a wearable device needs to have a wireless connection to send the data. The Arduino Nano 33 IoT offers both BLE (Bluetooth Low Energy) and Wi-Fi connections. Lastly, the Arduino Nano 33 IoT has an Inertial Measurement Unit sensor (LSM6DS3) integrated (there is a detailed explanation of this component in the next section) [22].



Figure 3. Arduino Nano 33 Iot real picture of the board [22].

The main features that made Arduino Nano 33 IoT a perfect choice for this project are the following:

- Wi-Fi and BLE connectivity as a result of u-Blox NINA-W102 (low power chipset). Consequently, the device could be wearable.
- Connectivity to Arduino IoT Cloud, which allows to upload code over the air. Thus, code could be modified without the need of external cables.
- LSM6DS3 (6 axis Inertial Measurement Unit) to detect the three-axis of acceleration and three-axis of rotation. This sensor is the component that allows us to track participants.
- Input voltage limit of 21V through Vin pin and 5V through USB port. This allows to power the device in multiple ways.
- Variety of communication protocols: UART, SPI, I2C. Our project uses I2C to communicate to the motor controllers and SPI to communicate through Wi-Fi.
- Lightweight: 5gr (with headers). A light weight board will be more comfortable for participants.
- Small dimension (45mm x 18mm). A small dimension board allows to create a suitable design to achieve our purposes.

The drawbacks of the board are the following:

- The operating voltage of Arduino Nano 33 IoT is 3.3V. Consequently, none of the other components should need a higher input voltage.
- Nano 33 IoT has no battery connector or charger. To solve this problem, we decided to connect a battery with a PowerBoost 1000 Charger (a detailed explanation of the component is given below).

Our project uses Vin, GND, SDA and SCL. The pinout of the board is the following (Fig. 4):

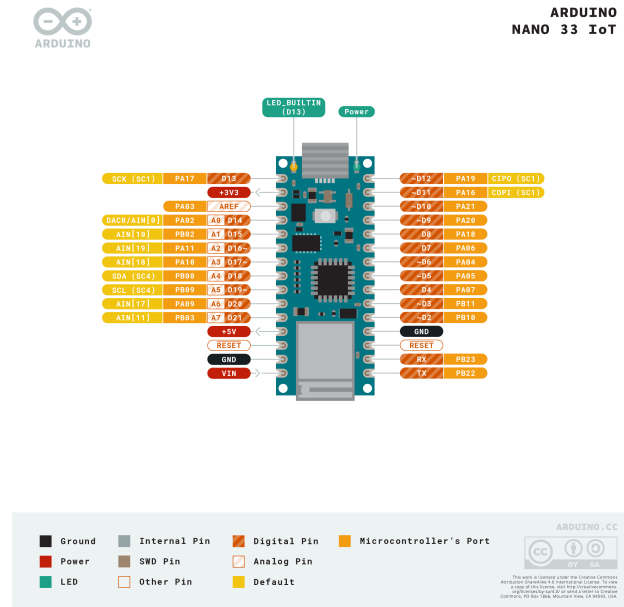


Figure 4. Arduino Nano 33 IoT pinout scheme [22].

4.2.2 LSM6DS3 (6 Degree of Freedom Inertial Measurement Unit)

The LSM6DS3 (Fig. 5) is a system-in-package that features a 3D digital accelerometer and gyroscope. It has a full-scale acceleration range of $\pm 2/\pm 4/\pm 8/\pm 16$ g and an angular rate range of $\pm 125/\pm 250/\pm 500/\pm 1000/\pm 2000$ dps [22]. The raw data to calculate the position comes from this component. IMUs are a convenient component to track a person because they are affordable, there is no need to install new components in the room, and they occupy a small space. Through the integration of both acceleration and gyroscope motion, it is possible to detect the position of the person in the space. However, noise and short term stability affect every sensor. This leads to errors in the results because the integration accumulates the drift from the raw data in a short period. Next sections will explain how to face this problem.

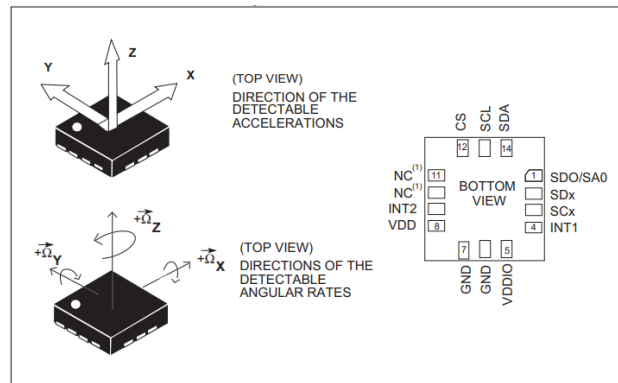


Figure 5. Pin description of LSM6DS3 [22].

4.2.3 TCA9548A I2C Multiplexer

The TCA9548A I2C (Fig. 6) multiplexer allows connecting more than one device with the same address to the Arduino. The Arduino Nano 33 IoT has one SDA (I2C data pin) and one SCL pin (I2C clock pin). These pins allow the user to connect one device with I2C connectivity to the Arduino. However, we required at least two connections for our project. In our approach, each haptic motor controller controls one motor disk. The motor controllers communicate with the Arduino with I2C connectivity, and each motor controller has the same address. Thus, we required a TCA9548A multiplexer to connect as many motor controllers as needed.

The TCA9548A can connect up to eight devices with the same address with one microcontroller. Moreover, it includes the following features [23]:

- Selectable I2C address 0x70-0x77.
- Weight: 1.8g.
- Product Dimensions: 30.6mm x 17.6mm x 2.7mm / 1.2" x 0.7" x 0.1".



Figure 6. Picture of the TCA9548A I2C Multiplexer [23].

4.2.4 Adafruit DRV2605L Haptic Motor Controller

The DRV2605L (Fig. 7) is a motor driver designed to control haptic motors, specifically buzzers and vibration motors. This device can produce 123 effects such as transition ramps, clicks, or alerts. The selection of this device allows us to give different instructions by providing vibrations at diverse speeds. Other crucial features of the motor drivers are [24]:

- Powered at 3.3V (the range is from 3 to 5V).
- Controlled with I2C connectivity.

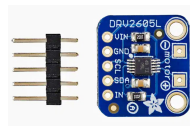


Figure 7. Picture of the DRV2605L [24].

4.2.5 Motor disks

The motor disks (Fig. 8) provide the vibrotactile feedback. They are small and light (0.9 grams, 10mm diameter, 2.7mm thick). They are controlled and powered with two wires, and require an input voltage between 2 and 5V. This makes it a convenient piece for the project because Arduino Nano 33 IoT provides a voltage of 3.3V [25].



Figure 8. Picture of the motor disk [25].

4.2.6 PowerBoost 1000 Charger - Rechargeable 5V Lipo USB Boost @ 1A - 1000C

The PowerBoost 1000 Charger (Fig. 9) is a power supply for the portable project. It has a built-in load-sharing battery charger circuit that makes it possible to charge and use the device at the same time. The main advantage of this board is that it converts the power of 3.7V (power provided by a LiPo battery) to an output of 5.2V for running 5V projects. The Arduino Nano 33 IoT is powered with 5V through the USB port or with 5-21V with the input and ground pins. Thus, this board is convenient to power our project with a rechargeable LiPoly battery [26].

The main component of the board is a TPS61090 boost converter from Texas Instruments. Apart from converting the voltage from 3.7 to 5V, its features include a low battery indicator (LED light turns red when the voltage is below 3.2V) and a high level of efficiency (90% operating efficiency in most cases).



Figure 9. Picture of PowerBoost 1000 Charger [26].

4.2.7 Lithium-Ion Polymer Battery - 3.7v 2500mAh

Lithium-Ion Polymer batteries (Fig. 10) are thin and light. The output can reach 4.2V when the device is completely charged, and its nominal output is 3.7V. The capacity of the battery is 2500mAh. Moreover, it is a rechargeable battery, which makes it perfect for a wearable device.



Figure 10. Picture of lithium-ion battery [27].

4.3 Inertial Navigation Systems

4.3.1 IMU Data processing

The haptic feedback hip belt provides vibrotactile feedback to guide participants to perform a task and tracks the position of a person with an Inertial Measurement Unit to verify that the exercise has been performed correctly. Obtaining the positions of the individual in real time using an IMU requires several manipulations, described below.

Navigation using an IMU involves estimating the path followed to travel from point A to point B, using the recorded accelerations and rotations from the device. For this, we first need to define a global reference frame. In other words, a coordinate frame that provides a reference relative to which the position of the device may be described [28]. For example, when people use a Global Positioning System to track location in a city, they use a map as a reference or a known grid system. In this situation, people determine via GPS their location in a Cartesian reference frame based on x and y coordinates. Position, orientation, and motion do not provide meaningful data by themselves. Thus, the solution to any navigation problem always implies using at least two coordinate frames. These involve the body frame, which describes the object that the navigation system tracks, and the reference frame, which describes a reference frame for calculating the orientation of the object.

There are four main reference frames commonly used in a navigation problem. They are Inertial Frame, Earth-Centered Earth-Fixed Frame, Local Navigation Frame, and Body Frame.

Inertial Frame

The Inertial Frame (denoted by the letter i) defines a coordinate system located at the Earth's center of mass. The spin axis of the planet defines its orientation. It is a reference frame that does not accelerate or rotate with respect to the Universe [29].

Earth-Centered Earth-Fixed Frame

The Earth-Centered Earth-Fixed Frame (denoted by the letter e) defines its axes fixed with respect to the Earth. The z -axis points from the center of the Earth to the north pole. The x -axis goes from the center to the intersection between the Conventional Zero Meridian and the equator. Finally, the y -axis is perpendicular and crosses the equator and the 90° east meridian [29].

Local Navigation Frame

The Local Navigation Frame (denoted by the letter n) defines its origin by the initial position of the object analyzed. Once the object starts moving, the N-Frame axis remain fixed while the object axis (known as Body Frame axis) move with the object. In the Local Navigation Frame the location of the axis correspond to north (x -axis), east (y -axis), and normal to the surface (z -axis) positions. This navigation frame is useful for large navigation [29].

The following figure shows the distribution of the axis of the I-Frame, the E-Frame and the N-Frame (Fig. 11):

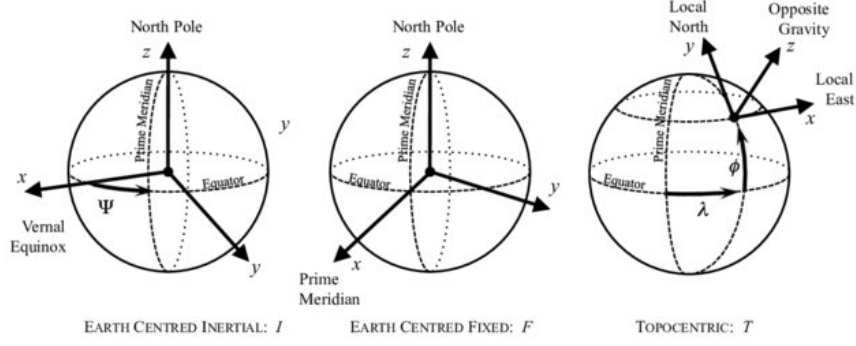


Figure 11. Inertial Frame, Earth-Centered Earth-Fixed Frame, and Local Navigation Frame [30].

Body Frame

Finally, the Body Frame defines the origin and orientation of the item analyzed. The origin of the Body Frame is equal to the origin of the Local Navigation Frame (Fig. 12). However, in the Body Frame, the axes remain invariable with respect to the object [29].

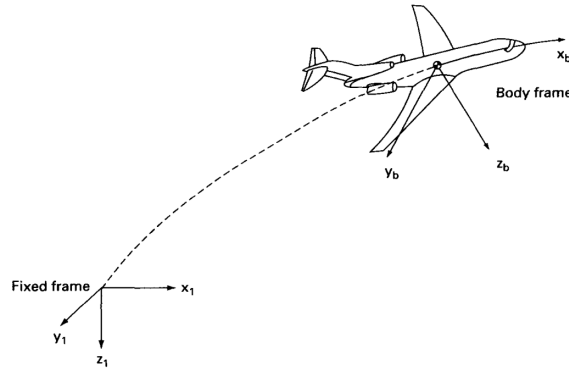


Figure 12. Body Frame with respect to fixed frame [31].

Our approach consists of recording the acceleration and angular velocities with an IMU in the Body Frame and converting the data to the Local Navigation Frame to obtain the accelerations in this reference frame. To convert the Body Frame to the Navigation Frame we apply the following formula [28]:

$$\dot{v}_e^n = C_b^n f_b - [2\omega_{ie}^n + \omega_{en}^n] \times v_e^n + g_l^n$$

where:

1. \dot{v}_e^n is the rate of change of v_e^n with respect to navigation axes. It is also known as acceleration with respect to N-frame.
2. C_b^n is the direction cosine matrix from Body Frame to Navigation Frame. It is a 3x3 matrix obtained from the angular velocities recorded by the gyroscopes. It represents unit vectors in the Body Frame and projects them in the N-frame.
3. f_b is the vector quantity that indicates the specific force measurements recorded by the accelerometers.

4. ω_{ie}^n is the Earth's rate with respect to the inertial frame.
5. ω_{en}^n is the turn rate of the navigation frame with respect to the Earth.
6. v_e^n is the velocity with respect to N-frame.
7. g_l^n is the local gravity vector. It combines the impact of the mass attraction of the Earth and the centripetal acceleration originated by the Earth's rotation.

This formula calculates the acceleration by taking into account four main terms:

1. The specific force acting in the object, $C_b^n f_b$, which calculation consists of multiplying the accelerations obtained by the IMU, f_b , by the values of the direction cosine matrix from the body to the navigation frame of the current sample, C_b^n . The multiplication of these two elements transform the measured specific force vector, f_b , into the N-frame. C_b^n values vary depending on the result the following equation gives:

$$\dot{C}_n^b = C_n^b \Omega_{nb}^b$$

where Ω_{nb}^b is the current skew-symmetric matrix, a square matrix whose transpose is equal to its negative, formed by the element of the vector ω_{nb}^b , which defines the turn rate of the body with respect to the N-Frame. The following equation provides the values of ω_{nb}^b :

$$\omega_{nb}^b = \omega_{ib}^b - C_n^b [\omega_{ie}^n + \omega_{en}^n]$$

where ω_{ib}^b represent the turn rate of the body with respect to I-Frame, C_n^b is the direction cosine matrix from the body to the N-Frame, ω_{ie}^n is the Earth's rate with respect to the I-Frame, and ω_{en}^n is the turn rate of the navigation frame with respect to the Earth. Thus, the turn rate of the body with respect to the N-frame is the difference between the measured body rates, ω_{ib}^b , and the estimates of the components of the N-frame.

2. A correction for the Coriolis acceleration which corresponds to the acceleration generated by the object's velocity on the rotating Earth's surface.
3. A correction for the centripetal acceleration of the device, as a result of its own motion on the outer layer of the Earth. For example, a person moving through the east would define a circular figure with respect to the I-Frame if the motion occurs over the Earth's surface. This phenomenon results from the effects that a force acting towards the center of the Earth of magnitude equal to the product of the object's mass, its turn rate with respect to the Earth, and its linear velocity causes to the device.
4. Compensation for the gravitational force acting on the object, which corresponds to the term g_l^n of the equation.

The process is summarized in (Fig. 13):

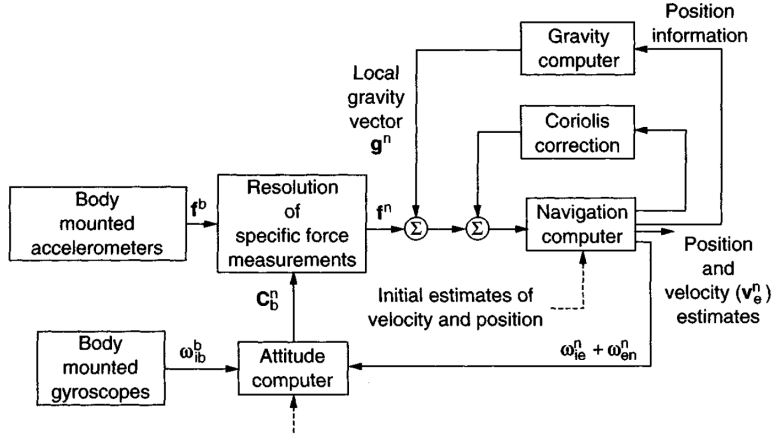


Figure 13. Local Navigation Frame mechanisation [28].

After calculating acceleration values in the new frame, we integrate them twice to provide estimates of velocity and position. Unfortunately, this leads to errors in the calculations, since raw data has drift as a result of integration. Thus, it is not possible to get an accurate estimation of the position of a person during more than a few seconds based just on IMU data [32]. One way to mitigate these errors would be to filter the data. Some studies demonstrate that without an algorithm that removes the error, low-cost IMUs accumulate an error in position equivalent to time cubed which means that the error exceeds a meter of deviation in a few seconds [33].

Human gait cycle consists of two phases defined by the trajectory that the foot follows while walking: stance phase and swing phase. The swing phase corresponds to the period where the foot analyzed is not in contact with the floor, whereas the stance phase corresponds to the part of the gait cycle where the foot is in contact with the ground. One way to tackle the drifting problem from the IMUs is to apply a Zero Velocity Update correction, ZUPT [33, 34, 35]. The idea comes from attaching the sensor to the foot and assuming that in the stance phase of every step, the foot is stationary and the velocity is zero.

4.4 Vicon cameras as ground-truth

In order to compare our IMU outputs to standard measurement approaches, we decided to collect data using a motion capture system. Specifically, we used cameras from Vicon (Fig. 14), which recognize reflective markers and collect their movement with a high precision level [36]. We collected several steps of walking with the IMU, while a reflective marker was attached on top of the device. We then compared position information obtained from IMU data to that obtained from the cameras.

For the purposes of this project, we decided to use IMUs instead of motion capture cameras because we ultimately want our devices to be more accessible and mobile than cameras allow. In addition, motion capture cameras are expensive and require an external setup. These properties would cause limitations for the use of the device because the product could only be used in clinics with enough space and resources to add this installation. Consequently, it would limit the possibility of creating a home-rehabilitation product. Due to the limitations caused by the cameras mentioned above, we concluded that IMUs was the perfect choice to satisfy our goals.

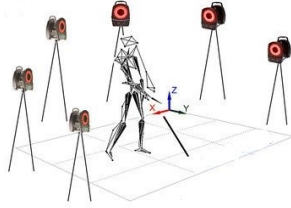


Figure 14. Setup of Vicon cameras [37].

4.5 Communication Protocols

Our device is a wirelessly wearable system coded in two programming languages: C++ and Matlab. Those two programming languages need to establish a communication loop. Arduino sends data to Matlab, Matlab processes the data to convert the raw data to position in the N-frame, and it sends data to Arduino to start and stop vibrating the motor disks. In first instance, we established this communication through the serial port.

We decided to communicate between the two programs through Wi-Fi, in order to be able to create a wearable device without cables between the microcontroller and the computer. The selection of the communication module followed a defined structure.

First, a comparison between BLE (Bluetooth Low Energy) and Wi-Fi helped us deciding the best communication module for our approach because our board Arduino Nano 33 IoT includes both methods to provide wireless communication. Consequently, both communications could suit this application. BLE is a wireless communication with limited ranges of transmission. It allows us to send data over short distances. Additionally, its design allows sending minimal amounts of data. The Inertial Measurement Unit sends data from six sensors every ≈ 0.01 seconds (100Hz). Thus, the use of this technology limits our application to short transmission ranges and small amounts of data. In contrast, Wi-Fi can transmit data at high velocities (1.3Gps), which makes it ideal for sending large amounts of data [38]. Moreover, some studies have shown that BLE do not necessarily use less power consumption than Wi-Fi for Internet of Things applications [39]. Due to all the reasons mentioned above, we decided to communicate with our device through Wi-Fi communication.

Once we decided that Wi-Fi was a better solution than BLE, we explored which types of communication protocols were suitable to communicate with Matlab and Arduino wirelessly. The two main existing communication protocols are TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). TCP involves a server and a client. It ensures that communication between the server and the client has been established before sending data packages. In contrast, UDP sends data to other network hosts without ensuring that communication between the server and the client has been established [40]. Although the connection in the UDP protocol is faster than in TCP, UDP does not ensure that the information has been received by the receiver.

For our current approach, we established a TCP/IP communication to ensure that the data sent was reliable. In this system, IP states for Internet Protocol and defines the address of the connected devices. TCP is responsible to send the information and guarantee that client and the server are communicating. In real-time environments, it will be helpful to implement UDP/IP communication to achieve a faster communication between Arduino and Matlab.

4.6 Programming tools

The two programming tools of this project are Arduino and Matlab, they communicate wirelessly through TCP/IP communication. Matlab processes all the data and gives instructions, whereas Arduino follows the directions that Matlab gives. The next sections describe the main parts of the code.

4.6.1 Arduino Code

Arduino code always includes four main parts: the selection of the libraries, the variables, the setup, and the loop.

1. Libraries

First, we include the libraries. They offer diverse functionalities to use in the sketches. Our code uses six libraries:

(a) SPI

SPI stands for Serial Peripheral Interface. Microcontrollers use this serial data protocol for communicating with peripherals at small distances.

(b) WifiNINA

This library allows for establishing a network connection (local and internet) with diverse boards. one of them is Arduino Nano 33 IoT. The most valuable feature that it offers for our project is the possibility to create servers and clients, and send or receive UDP packets through Wi-Fi. Moreover, it allows us to connect to the network and read the IP address of the board.

(c) Adafruit DRV2605

This library controls the Adafruit DRV2605L Haptic Driver.

(d) Adafruit Sensor

The library serves to unify the systems of sensor drivers and ensure that the code would work correctly if a sensor needs a replacement or an update to a new version.

(e) Arduino LSM6DS3

This library allows for reading of the Inertial Measurement Unit LSM6DS3 data from the accelerometers and gyroscopes on the Arduino Nano 33 IoT.

(f) elapsedMillis

This library serves to create a timer. The user can create a variable that increases as time elapses.

2. Variables

Since we are connecting our device through Wi-Fi, we first need to define the Wi-Fi credentials to establish the communication (SSID and password). Then, we define a server with a known port to receive the command from Matlab. Finally, the definition of the IP address of the PC and the port serve to send data to Matlab.

The next step is defining a variable to set a timer. It is important to define the I2C TCA9548A multiplexer and assign a unique ID address to the motor controllers.

3. Setup

After defining the variables, we implement the setup of the system. First, we select the baudrate and Arduino shows the connection between Wi-Fi and the IP address. Second, the program shows that the IMU has initialized successfully. Third, we begin the two motor controllers and select the library to control them. Moreover, we set the mode to trigger I2C by sending an initializer command. Finally, we initialize the SDA and SCL output pins of the multiplexer used for our application.

4. Loop

Several steps are required to send the data from Arduino to Matlab: 1) use a WifiClient class to create a TCP connection to send the data from the accelerometers and gyroscopes to Matlab, 2) start a timer and call the function *millis()* to register the time samples, 3) create six variables types to store three accelerations and three angular velocities in every iteration, 4) check if the IMU is ready to collect the data, 5) if the IMU is available, Arduino prints the data with the command *client.println* and Matlab receives the data, 6) the time counter resets in every iteration.

The receipt of the data from Matlab to Arduino follows a different structure. First, the program checks the server availability and the client connection. Second, Arduino defines an outcome for every command sent by Matlab. For example, we select that if Arduino receives an 'a', the motor disk located it the right will vibrate. The commands follow a similar structure to send vibration to the left and to stop the vibration. Thus, Arduino code is mainly used to provide the haptic feedback.

4.6.2 Matlab

The Matlab code consists of two main parts: the calculation of the position by taking the raw data and transforming it to the estimation of the position in the N-frame, and the control of the vibrations. The current application does not process the data in real-time. Thus, the TCP connection focuses on sending the data but not collecting it. In our current approach, Arduino sends the data through the serial and Matlab collects it and saves it in a .csv file. Later, we process the data to obtain the acceleration in the N-frame which serves to calculate the position in the N-frame.

The calculation of the position in the N-frame follows several steps. First, we define our current location with respect to the Earth's coordinates. In our case, the location is Irvine, California, with a latitude of 33.64° , and a longitude of -117.84° . The gravity in this location is 9.796 m/s^2 . Moreover, we provide the angular velocity of the rotation of the Earth, which corresponds to 7.29×10^{-5} radians/second, and the radius of the Earth, which is 6.37×10^6 meters.

Second, the program identifies the sample rate of the board, which is 101.4 Hz. Moreover, we convert the accelerations and the gyroscopes units to the international system. Thus, accelerations transform from g to m/s^2 and the gyroscope data transforms from degrees/second to radians/second.

Third, the sensors initialize. We define the initialization period, in our approach it corresponds to the first two seconds of data collection. This period serves to take the mean of the data recorded during this period from the gyroscopes. The mean from the accelerations recorded during these two seconds serve to later obtain the initial roll, pitch and yaw angles.

The following steps consist of obtaining all the components needed to calculate the acceleration in the N-frame: 1) calculate the initial direction cosine matrix from the Body Frame to the N-Frame, 2) calculate the turn rates of the Earth expressed on the local geographical frame and the turn rate of the local geographical frame with respect to the Earth frame, 3) calculate the angular rate measurements in the N-frame, 4) calculate the skew-symmetric matrix with the values of the angular rate measurements, 5) obtain the direction cosine matrix for every sample, 6) calculate the local gravity vector in the N-frame, 7) calculate the acceleration in the N-frame. These steps are explained in detail above in the Inertial Navigation Systems section.

After obtaining the result of the acceleration in the N- frame, we apply a ZUPT correction. The Inertial Navigation Systems (INS) section explains this concept. There are several methods to apply the ZUPT correction. The most intuitive manner is to plot the acceleration, localize where the accelerations are close to zero, and manually select the start and end points of the correction. However, this method is not effective for a real-time environment. One way to automate the process is to calculate the magnitude of the acceleration and the magnitude of the gyroscopes and set a threshold for at which we assume the device is stationary. If the sample value of one of the two signals is lower than the threshold, the ZUPT correction acts, and the velocity is set to zero. This approach works for some data collections, but it has some flaws. For example, if the signal is very noisy, the threshold stops functioning because the velocity is set to zero when it should not. Another problem could be a poor detection of the start and the end point of the correction.

The ZUPT correction is a strategy to set the velocity to zero when the foot is in stance phase. After selecting these points, we obtain the velocity of the steps with a discrete integration of the acceleration. Later, we calculate the position with a discrete integration of the velocity.

$$v(i) = v(i - 1) + a(i) * (1/f_s)$$

$$x(i) = x(i - 1) + v(i) * (1/f_s)$$

After that, we use the position samples in a real-time simulation to control the motor disks. To create a real-time environment it is necessary to define the IP address of the board, the port, and use the command *tcpclient*. Later, the haptic feedback starts after recording a selected number of samples. To start the vibration, Matlab sends a command to Arduino. Depending on the command sent by Matlab, the left motor disk or the right motor disk starts vibrating. Afterwards, we calculate the first reference vector with the following one hundred samples, this vector defines the initial direction of the motion. Later, we calculate a new current vector of one hundred samples which is constantly updating throughout the task, this second vector defines the current direction the movement. These two vector serve to calculate the angle formed between the first reference vector and the second current vector in every iteration. When the value of the angle is higher than 80° , the vibration stops (Matlab sends a command to stop vibrating to Arduino) because the person has performed the task correctly. The person continues moving and the devices continues tracking the position. Then, Matlab repeats the whole process. First, a vibration occurs to give a new indication. Second, the program updates the new reference vector and the current vector. Third, The haptic feedback stops when the person performs the task as indicated.

4.7 Belt prototype

The prototype of the product consists of a strap with sewed electronics. The prototype places the two motor controllers DRV2605L packed with the motor disks in the two sides of the hip. The TCA9548A I2C Multiplexer, the PowerBoost 1000 Charger, and the Lithium-Ion Polymer Battery are placed on the back of the belt. Finally, one cable goes from the back to the heel of the person to connect the system to the microcontroller Arduino Nano 33 IoT. The foot carries the Arduino Nano 33 IoT because it incorporates the IMU that collects the data to obtain an estimation of the position and to set the velocity to zero every time the foot is in stance phase.

The first step to design the prototype was deciding the components (explained in the hardware section) and defining how to connect the cables. We decided to make the following connections: 1) connect the Lithium-Ion Polymer Battery to the PowerBoost 1000 Charger that converts the voltage of 3.7V given by the battery to 5V, 2) connect the Arduino Nano 33 IoT board to the PowerBoost 1000 Charger through a micro USB to provide 5V the board, 3) connect Arduino Nano 33 IoT to Vin and GND of the multiplexer and the motor controllers to power the whole system, 4) connect A0, A1 and A2 from the TCA9548A multiplexer to GND, 5) connect the SDA and SCL pins from Arduino Nano 33 IoT to the SDA and SCL pins of the TCA9548A multiplexer to establish the I2C communication, 6) connect the pins SDA and SCL from the motor controllers to the output pins SC7, SD7, SC6 and SC6 of the multiplexer, 7) connect the motor disk to the motor controllers. The connections designed with Fritzing are below (Fig. 15).

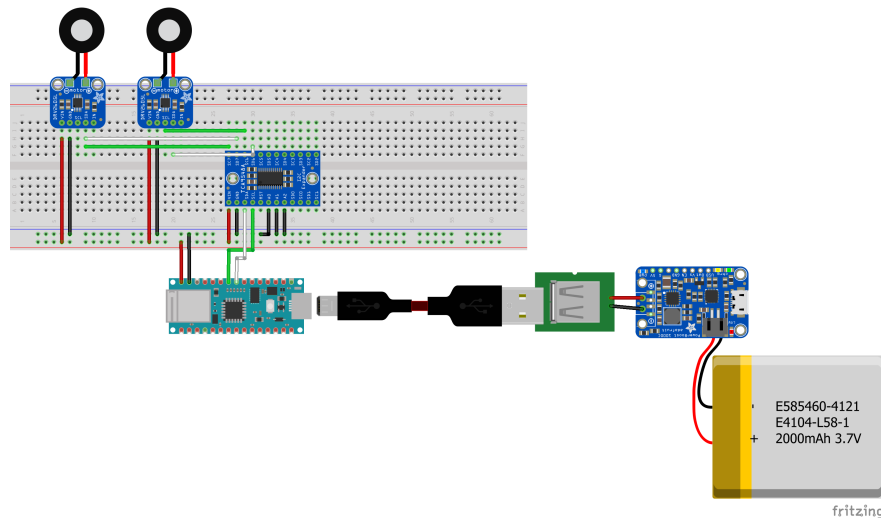


Figure 15. Electronic diagram.

The second step was creating an encapsulation for the motor disks and the motor controllers DRV2605L. We designed the piece with Solidworks (Fig. 16). We 3D printed the piece in the laboratory with Flashforge-ABS material. We converted the *sldprt* file generated by Solidworks to *stl* and we loaded it to Flashprint. The program converted the file to *x3g* to print it.

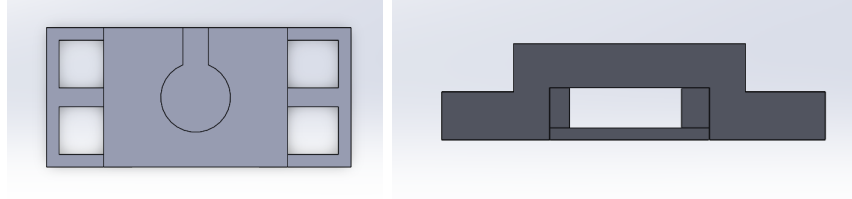


Figure 16. CAD of the encapsulation of the motor disks and motor controllers.

The third step was assembling the device. We soldered the electronic components that needed it and we packed the motor disks and the motor controllers together in the 3D printed encapsulation. Later, we connected everything following the previous mentioned connections. Finally, we sewed the electronics to the strap.

4.7.1 Schematics diagram

This section shows the electronics diagram of the core part of the system (Fig. 17).

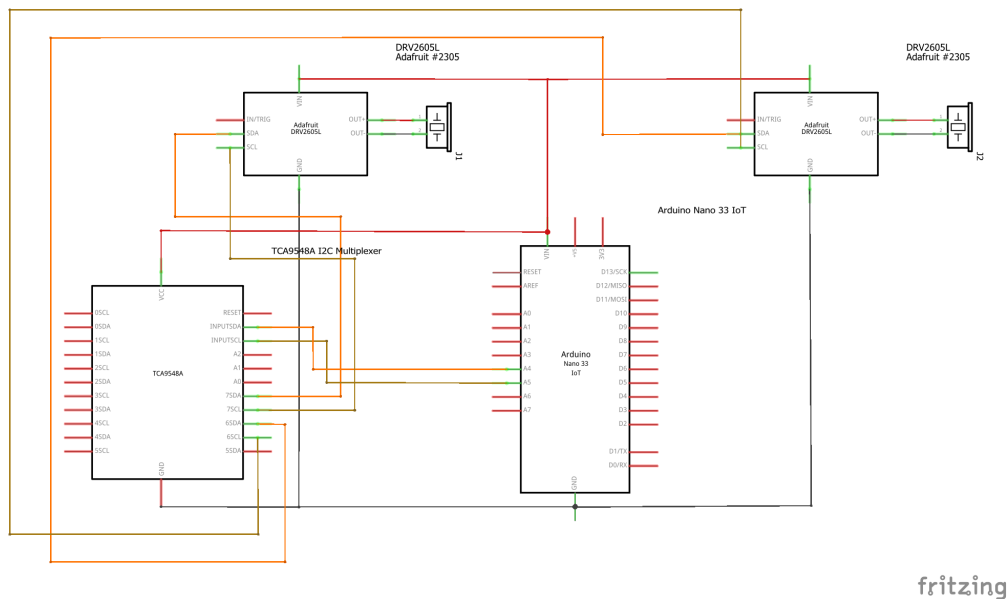


Figure 17. Schematics of the design.

5 RESULTS

The main challenge to create the device was achieving a clean position signal from the Inertial Measurement Unit. Every sensor from the 6 Degrees of Freedom LSM6DS3 (three accelerometers and three gyroscopes) produces its own noise. Calculating the position in the Local Navigation Frame implies integrating two times the acceleration obtained on this reference frame. Thus, every iteration doubles the noise and the error grows exponentially. The calculation of the acceleration in the N-frame takes into account the raw data from the gyroscopes and accelerometers (a force acting in the IMU), the Coriolis correction, the centripetal acceleration, and a compensation of the gravitational force. Any incorrect value from one of those components can produce more drift in the outcome. This condition caused a problem for tracking the IMU on the hip. Thus, we decided to place it over the foot, specifically on the heel, to be able to apply the ZUPT correction (Fig. 18).



Figure 18. Arduino Nano 33 IoT attached to the foot.

To study the amount of drift obtained post-processing the IMU data collections with ZUPT correction, we recorded data from motion capture cameras (Vicon) and data from the IMU simultaneously. We placed The IMU and a reflective marker on the heel to collect three steps. The motion mainly occurred in the y-axis. A comparison between the estimation of the position obtained with both data collections is shown below (Fig. 19).

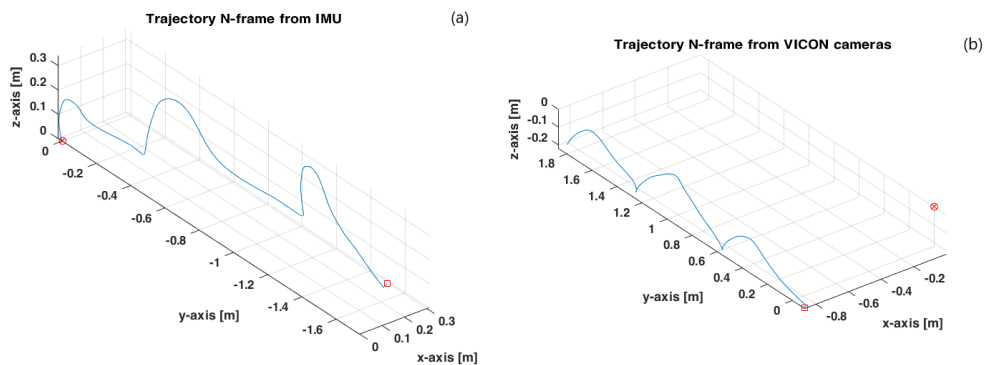


Figure 19. Comparison between 3D position of three steps recorded by different systems. (a) 3D position of three steps recorded with the IMU placed on the heel. (b) 3D position of three steps recorded with motion capture cameras with one marker placed on the heel.

Our study does not require analyzing the data from the z-axis coordinate because we are encouraging people to move left and right, but not up and down. Thus, we are studying the 2D plane of x and y coordinates (Fig. 20). Comparing the post-processing results of the data from the IMU and the data from motion capture cameras, we noted a loss of 0.2 meters in 1.8 meters of motion in the y-axis. Thus, data loss $\approx 11\%$ in ≈ 20 seconds in y-axis. In the x coordinate, the data from the cameras remains almost constant at one point. In contrast, the data from the IMU oscillates between 0 and 0.3 meters. Even though the trajectories from the data collection of the IMU and the motion capture cameras look similar, the Vicon cameras' data is still more precise and accurate.

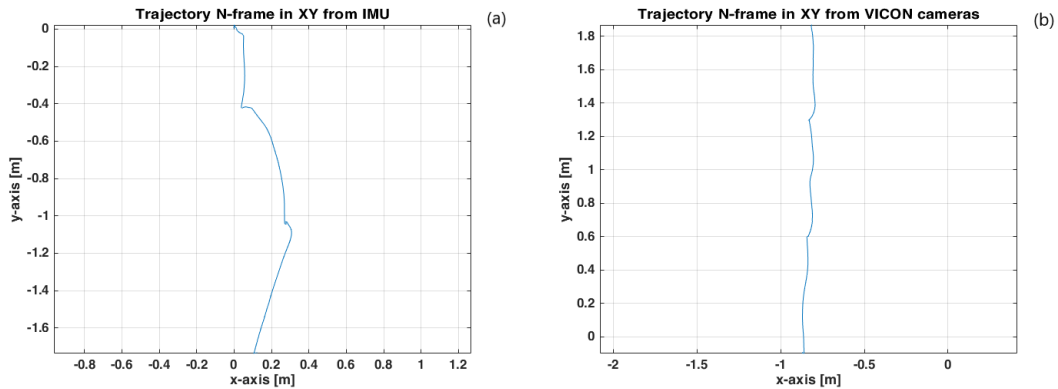


Figure 20. Comparison between 2D position of three steps recorded by different systems.(a) Position of three steps in X and Y coordinates recorded with the sensor placed on the heel.(b) Position of three steps in X and Y coordinates recorded with motion capture cameras with one marker placed on the heel.

The comparison between the velocities of the two data collections reveals that the data of the x-axis and z-axis remain similar in the two systems (Fig. 21). In contrast, the data from the y-axis varies notably. In the IMU data collection, velocity increases in every step, whereas the velocity of every step is similar in the motion capture cameras data collection. The main reason is that the motion mainly occurs in the y-axis, and our current solution sets the velocity to zero without correcting the biases from the IMU caused by the roll and pitch angles. A possible solution could be by applying an Extended Kalman Filter. We will provide an explanation of this method in the next section.

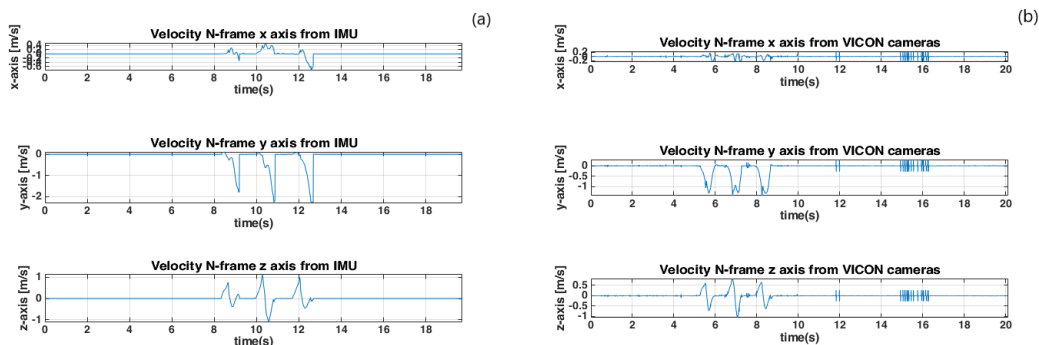


Figure 21. Comparison between velocity recorded by different systems. (a) Velocity of three axis when three steps were recorded with the IMU attached to the heel. (b) Velocity of three axis when three steps were recorded with motion capture cameras with one marker placed on the heel.

The real-time environment to control the vibrations gives satisfactory results (Fig. 22). We used motion capture cameras recordings to demonstrate that our method to stop the haptic feedback when a task has been performed worked correctly. Figure 22 shows the results obtained with a data collection where a person turned three times to the left. Green circles indicate the start of the vibration and red crosses mark the end of the haptic feedback. The motion started at (0.3, 2.25)m and stopped at (0.3, 1.95)m. The vibration started after a selected number of samples. It stopped after the reference vector and the current vector defined an angle higher than 80° . Thus, the vibration stopped correctly after every turn. In real-life, we could simulate a wearable device controlling the motor disks via Wi-fi without the need to connect a cable to the computer.

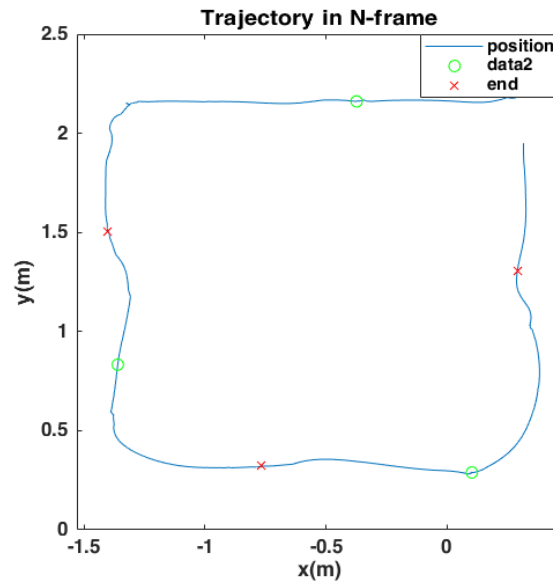


Figure 22. Trajectory in N-frame followed by a participant turning three times to the left. The green circles indicate the start points of vibration and the red crosses show the end points of the them.

We successfully created a prototype of the product (Fig. 23). It is an adjustable belt connected to the microcontroller placed on the foot. To remember the assembling process, refer to the last section.

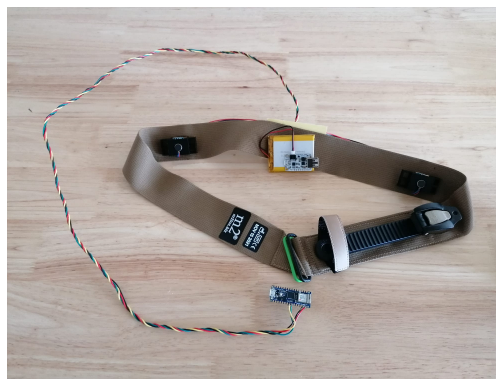


Figure 23. Belt prototype.

6 LIMITATIONS AND FUTURE DIRECTIONS

There are still several issues to address before testing this device with patients. One limitation for the IMU data processing was that we do not currently know which method is most optimal for applying the ZUPT correction when the foot is stationary. The current approach works correctly with some data samples, but it does not work satisfactorily for all data collections. In future studies, finding a more robust method to apply ZUPT correction will be necessary.

Position error from the IMU can also be reduced by applying an Extended Kalman Filter (EKF). Some studies have shown that introducing the ZUPT correction as measurements into an EKF corrects a considerable part of the position drift [33]. This filter is a linearization process of a nonlinear system [41]. The Kalman Filter estimates the value of the next sample based on the last value obtained. The filter constantly updates the variables of the system due to the analysis of the input and output values [41]. This allows the filter to obtain an optimal estimation. In Inertial Navigation Systems, the filter can be used to track the correlations between the errors from the integration and double integration of the acceleration. The filter also compensates pitch and roll deviations that affect the velocity detection [42]. Consequently, the filter reduces the cubic-in-time velocity error and replaces it by a linear error accumulation [42]. In future work, we will likely need to implement the filter for more accurate position tracking. This will lead to more robust detection of stance phase and will produce more reliable trajectories.

Another limitation that this study needs to face is related to real-time tracking. The real-time application depends on the data transmission frequency. When we established the TCP/IP communication, data transmission resulted to be slower than serial communication (there was a transmission frequency reduction from $\approx 100\text{Hz}$ to $\approx 10\text{Hz}$). This leads to a limitation to process the data in real-time. A possible solution to this problem is modifying the communication protocol (from TCP/IP to UDP/IP) to provide a faster data transmission. To temporarily avoid the limitation caused by this issue, we collected the data through the serial. However, the poor performance of the detection of the position with the ZUPT correction also limited the real-time application. In this study, we analyzed sensors outputs by recording data in a .csv file and processing off-line. This strategy helped to understand better the flaws of the current ZUPT correction. In future studies, a UDP communication and a robust ZUPT correction will be needed to be able to transform the data from the sensor to position in real-time.

Finally, the lack of a vigorous method to maintain the microcontroller that contains the IMU in a permanent position with respect to the heel could be causing errors. Currently, we have attached the device to the foot with tape. In future studies, we will design and 3D print a casing for the sensor. We will attach this piece to the shoe to ensure that the IMU remains in the same position during all the data collection. Consequently, we will remove a possible noise source.

7 CONCLUSION

Motor learning, during rehabilitation or when using an assistive device, generally involves conducting uniform movements. However, past research has shown that variability during training can be beneficial to learning. As a result, a device that can help people be more variable during gait training could improve not only clinical rehabilitation but also the efficacy of assistive robotic devices. In this thesis we present an initial design for a wearable device that can encourage gait exploration. We have shown that it is possible to track user positions by processing acceleration and gyroscope data obtained from a low-cost Inertial Measurement Unit. We have also shown that it is possible to control vibrotactile feedback based on the current position while simulating a real-time environment. Additionally, we have demonstrated that Arduino and Matlab can communicate through Wi-Fi, allowing for our device to be wearable. Furthermore, we have created a prototype of the future device. However, there are some limitations that limit the efficacy of the device. For example, we need to modify the zero velocity correction algorithm to be more robust and we need to establish a communication protocol that allows sending the data through Wi-Fi at the same frequency the IMU collects the data. Future work will address these limitations in order to make a reliable device that will encourage gait exploration during training and hopefully lead to more effective clinical rehabilitation and gait training protocols.

CONCLUSIONS

Generalment, l'aprenentatge motor durant la rehabilitació o quan s'utilitza un dispositiu d'assistència inclou la realització de moviments invariables. No obstant això, previs estudis han demostrat que l'aprenentatge motor pot veure's beneficiat a l'introduir variabilitat a l'entrenament. En conseqüència, crear un dispositiu que ajudi a la gent a fer moviments més variables durant l'entrenament de la marxa humana podria ser beneficiós tant per la rehabilitació clínica com per l'eficàcia dels dispositius d'assistència robòtica. Aquest treball de final de grau presenta un disseny inicial d'un dispositiu portable que podria fomentar l'exploració de la marxa humana. L'estudi demostra que, un cop processades les dades d'acceleració i velocitat angular obtingudes mitjançant una unitat de mesura inercial (IMU) de baix cost, és possible fer un seguiment de la posició d'un usuari. A més, s'ha creat una simulació en temps real per demostrar que és possible controlar realimentació vibrotàctil basant-se en la posició actual d'un individu. També s'ha demostrat que Arduino i Matlab poden comunicar-se via Wi-Fi, permetent així que el dispositiu sigui portable. L'estudi també inclou un prototip del dispositiu que es crearà en el futur. Malgrat tot, s'han detectat algunes limitacions que condicionen l'eficàcia del dispositiu. Per exemple, és necessari modificar l'algoritme que corregeix la velocitat posant-la a zero per tal que sigui més robust. A més, s'ha d'establir un protocol de comunicació que permeti enviar les dades via Wi-Fi a la mateixa freqüència que la unitat de mesura inercial registra les dades. En el futur es treballarà per corregir aquestes limitacions per tal de produir un dispositiu fiable que fomenti l'exploració de la marxa humana durant l'entrenament. En el millor dels casos, el dispositiu contribuirà a desenvolupar a una rehabilitació clínica més eficaç i protocols d'entrenament de la marxa humana més efectius.

CONCLUSIONES

Generalmente, el aprendizaje motor durante la rehabilitación o al utilizar dispositivos de asistencia incluye la realización de movimientos invariables. No obstante, previos estudios han demostrado que el aprendizaje motor puede beneficiarse de la introducción de variabilidad al entrenamiento. En consecuencia, crear un dispositivo que pueda ayudar a la gente a hacer movimientos más variables durante el entrenamiento de la marcha humana podría ser beneficioso tanto para la rehabilitación clínica como para la eficacia de dispositivos de asistencia robótica. Este trabajo de final de grado presenta un diseño inicial de un dispositivo portable que podría fomentar la exploración de la marcha humana. El estudio demuestra que, tras el procesamiento de datos de aceleración y velocidad angular obtenidos mediante una unidad de medición inercial (IMU) de bajo coste, es posible hacer un seguimiento de la posición del usuario. Además, se ha creado una simulación en tiempo real para demostrar que es posible controlar realimentación vibrotáctil basándose en la posición actual de un individuo. También se ha demostrado que Arduino y Matlab pueden comunicarse vía Wi-Fi, permitiendo así que el dispositivo sea portable. Adicionalmente, el estudio incluye un prototipo del dispositivo que se creará en el futuro. A pesar de todo, se han detectado algunas limitaciones que condicionan la eficacia del dispositivo. Por ejemplo, es necesario modificar el algoritmo que corrige la velocidad al ponerla a cero para que el sistema sea más robusto. Además, se debe establecer un protocolo de comunicación que permita enviar los datos vía Wi-Fi a la misma frecuencia que la unidad de medición inercial (IMU) registra los datos. En el futuro se trabajará para corregir estas limitaciones con la finalidad de establecer un dispositivo fiable que fomente la exploración de la marcha humana durante el entrenamiento. En el mejor de los casos, el dispositivo contribuirá a desarrollar una rehabilitación clínica más eficaz y protocolos de entrenamiento de la marcha humana más efectivos.

References

- [1] K. L. Poggensee, S. H. Collins, How adaptation, training, and customization contribute to benefits from exoskeleton assistance, *Science Robotics* 6 (58) (2021) eabf1078.
- [2] K. E. Gordon, D. P. Ferris, Learning to walk with a robotic ankle exoskeleton, *Journal of biomechanics* 40 (12) (2007) 2636–2644.
- [3] G. S. Sawicki, D. P. Ferris, Mechanics and energetics of level walking with powered ankle exoskeletons, *Journal of Experimental Biology* 211 (9) (2008) 1402–1413.
- [4] J. W. Krakauer, T. Kitago, J. Goldsmith, O. Ahmad, P. Roy, J. Stein, L. Bishop, K. Casey, B. Valladares, M. D. Harran, et al., Comparing a novel neuroanimation experience to conventional therapy for high-dose intensive upper-limb training in subacute stroke: The smarts2 randomized trial, *Neurorehabilitation and neural repair* 35 (5) (2021) 393–405.
- [5] K. E. Adolph, W. G. Cole, M. Komati, J. S. Garciaguirre, D. Badaly, J. M. Lingeman, G. L. Chan, R. B. Sotsky, How do you learn to walk? thousands of steps and dozens of falls per day, *Psychological science* 23 (11) (2012) 1387–1394.
- [6] L. Lunenburger, G. Colombo, R. Riener, V. Dietz, Biofeedback in gait training with the robotic orthosis lokomat, in: *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Vol. 2, IEEE, 2004, pp. 4888–4891.
- [7] J. Bergmann, C. Krewer, P. Bauer, A. Koenig, R. Riener, F. Müller, Virtual reality to augment robot-assisted gait training in non-ambulatory patients with a subacute stroke: a pilot randomized controlled trial., *European journal of physical and rehabilitation medicine* 54 (3) (2017) 397–407.
- [8] F. Tamburella, J. C. Moreno, D. S. Herrera Valenzuela, I. Pisotta, M. Iosa, F. Cincotti, D. Mattia, J. L. Pons, M. Molinari, Influences of the biofeedback content on robotic post-stroke gait rehabilitation: electromyographic vs joint torque biofeedback, *Journal of neuroengineering and rehabilitation* 16 (1) (2019) 1–17.
- [9] M. Janssen, R. Pas, J. Aarts, Y. Janssen-Potten, H. Vles, C. Nabuurs, R. van Lummel, R. Stokroos, H. Kingma, Clinical observational gait analysis to evaluate improvement of balance during gait with vibrotactile biofeedback, *Physiotherapy Research International* 17 (1) (2012) 4–11.
- [10] J. Jonsdottir, D. Cattaneo, A. Regola, A. Crippa, M. Recalcati, M. Rabuffetti, M. Ferrarin, A. Casiraghi, Concepts of motor learning applied to a rehabilitation protocol using biofeedback to improve gait in a chronic stroke patient: an ab system study with multiple gait analyses, *Neurorehabilitation and neural repair* 21 (2) (2007) 190–194.
- [11] J. Müller, S. Müller, H. Baur, F. Mayer, Intra-individual gait speed variability in healthy children aged 1–15 years, *Gait & Posture* 38 (4) (2013) 631–636.
- [12] C. K. Rhea, C. J. Wutzke, M. D. Lewek, Gait dynamics following variable and constant speed gait training in individuals with chronic stroke, *Gait & posture* 36 (2) (2012) 332–334.
- [13] R. A. Schmidt, A schema theory of discrete motor skill learning., *Psychological review* 82 (4) (1975) 225.

- [14] N. M. Seel, Encyclopedia of the Sciences of Learning, Springer Science & Business Media, 2011.
- [15] J. A. Adams, A closed-loop theory of motor learning, *Journal of motor behavior* 3 (2) (1971) 111–150.
- [16] D. A. Rosenbaum, *Human motor control*, Academic press, 2009.
- [17] C. K. Williams, An exploration of the utility of augmented haptic feedback for learning a curve-tracing task, Ph.D. thesis, University of Toronto (Canada) (2018).
- [18] G. Wulf, R. Lewthwaite, Optimizing performance through intrinsic motivation and attention for learning: The optimal theory of motor learning, *Psychonomic bulletin & review* 23 (5) (2016) 1382–1414.
- [19] A. Booth, Biofeedback to improve gait in children with cerebral palsy (2019).
- [20] P. B. Shull, D. D. Damian, Haptic wearables as sensory replacement, sensory augmentation and trainer—a review, *Journal of neuroengineering and rehabilitation* 12 (1) (2015) 1–13.
- [21] K. H. Sienko, M. D. Balkwill, L. Oddsson, C. Wall, Effects of multi-directional vibrotactile feedback on vestibular-deficient postural performance during continuous multi-directional support surface perturbations, *Journal of Vestibular Research* 18 (5-6) (2008) 273–285.
- [22] A. O. Shop, Arduino nano 33 iot.
URL <https://store-usa.arduino.cc/products/arduino-nano-33-iot>
- [23] TCA9548A, Adafruit.
URL <https://learn.adafruit.com/adafruit-tca9548a-1-to-8-i2c-multiplexer-breakout/overview>
- [24] DRV2605L, Adafruit.
URL <https://learn.adafruit.com/adafruit-drv2605-haptic-controller-breakout/pinouts>
- [25] MotorDisks, Adafruit.
URL <https://www.adafruit.com/product/1201>
- [26] PowerBoost1000Charger, Adafruit.
URL <https://www.adafruit.com/product/2465>
- [27] Battery-3.7v, Adafruit.
URL <https://www.adafruit.com/product/328>
- [28] D. Titterton, J. L. Weston, J. Weston, *Strapdown inertial navigation technology*, Vol. 17, IET, 2004.
- [29] N. Bonnor, Principles of gnss, inertial, and multisensor integrated navigation systems—second edition paul d. groves artech house, 2013, 776 pp isbn-13: 978-1-60807-005-3, *The Journal of Navigation* 67 (1) (2014) 191–192.
- [30] P. Furgale, J. Enright, T. Barfoot, Sun sensor navigation for planetary rovers: Theory and field testing, *IEEE Transactions on Aerospace and Electronic Systems* 47 (3) (2011) 1631–1647.

- [31] R. C. Nelson, et al., Flight stability and automatic control, Vol. 2, WCB/McGraw Hill New York, 1998.
- [32] E. Foxlin, et al., Motion tracking requirements and technologies (2002).
- [33] Y. Wang, A. Chernyshoff, A. M. Shkel, Error analysis of zupt-aided pedestrian inertial navigation, in: 2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN), IEEE, 2018, pp. 206–212.
- [34] I. Skog, P. Handel, J.-O. Nilsson, J. Rantakokko, Zero-velocity detection—an algorithm evaluation, IEEE transactions on biomedical engineering 57 (11) (2010) 2657–2666.
- [35] M. Ma, Q. Song, Y. Li, Z. Zhou, A zero velocity intervals detection algorithm based on sensor fusion for indoor pedestrian navigation, in: 2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), IEEE, 2017, pp. 418–423.
- [36] A. Pfister, A. M. West, S. Bronner, J. A. Noah, Comparative abilities of microsoft kinect and vicon 3d motion capture for gait analysis, Journal of medical engineering & technology 38 (5) (2014) 274–280.
- [37] F. Mun, S. W. Suh, H.-J. Park, A. Choi, Kinematic relationship between rotation of lumbar spine and hip joints during golf swing in professional golfers, Biomedical engineering online 14 (1) (2015) 1–10.
- [38] D. Geevarghese, Ble vs wi-fi: Which is better for iot product development?
URL <https://www.cabotsolutions.com/ble-vs-wi-fi-which-is-better-for-iot-development>
- [39] D. Thomas, E. Wilkie, J. Irvine, Comparison of power consumption of wifi inbuilt internet of things device with bluetooth low energy, International Journal of Computer and Information Engineering 10 (10) (2016) 1856–1859.
- [40] S. Kumar, S. Rai, Survey on transport layer protocols: Tcp & udp, International Journal of Computer Applications 46 (7) (2012) 20–25.
- [41] S. Wang, C. Fernandez, Y. Chunmei, F. Yongcun, C. Wen, D.-I. Stroe, Z. Chen, Battery System Modeling, Elsevier, 2021.
- [42] E. Foxlin, Pedestrian tracking with shoe-mounted inertial sensors, IEEE Computer graphics and applications 25 (6) (2005) 38–46.