University of Applied Sciences
HOCHSCHULE EMDEN·LEER

Project Report - Analytics and Mathematics

Gait Pattern Recognition with an Arduino Portenta H7 IMU

First Author: Peeranut Noonurak
Matriculation No.: 7023582
Second Author: Romin Kantibhai Mangroliya
Matriculation No.: 7023541
Third Author: Francisco David Vilaplana de Trías
Matriculation No.: 7023638
Course of Studies: Master's in Industrial Informatics

First examiner: Prof. Dr. Elmar Wings
Second examiner: Prof. Dr.-Ing. Armando Walter Colombo

Submission date: 4 January 2023

University of Applied Sciences Emden/Leer · Faculty of Technology ·
Department of Electrical Engineering and Computer Science
Master Degree in Industrial Informatics
Constantiaplatz 4 · 26723 Emden · http://www.hs-emden-leer.de

Contents

List of Figures

List of Tables

Acronyms

ACSM American College of Sports Medicine

DCNN Deep Convolutional Neural Network

EMG Electromyography

HO Heel-Off

HS Heel-Strike

IMU Inertial Measurement Unit

KDD Knowledge Discovery in Databases

Li-Po Lithium Polymer Battery

LSTM Long Short-Term Memory

MEMS Micro-Electromechanical Systems

RAMI 4.0 Reference Architecture Model Industrie 4.0

RNN Recurrent Neural Network

SGD Stochastic Gradient Descent

SRA Standard Reference Architecture

TF Time-Frequency

TO Toe-Off

TS Toe-Strike

1 Introduction

1.1 Introduction into your case for demonstrating of your analytics, if not given define one use case

1.1.1 Describe your case

The systematic examination of a person's walking gait is known as gait analysis. Examining a person's gait is crucial for spotting anomalies in their walking style [HKP+16]. Changes in gait patterns provide vital fitness information that might be utilized to evaluate or analyze people suffering from pathological conditions that affect their capacity to walk and their entire bio-mechanics system [ADU+16]. Gait analysis was once performed by a human observer, however and video recording has made the process more reliable and precise. This qualitative analysis method is still widely known. But, this method is labor-intensive, requires an expert therapist, and requires more precision than motion analysis.

Hence, a more objective gait technique is required, and an Inertial Measurement Unit (IMU) is another viable solution for the previously mentioned study. The use of IMU sensors with accelerometers has already been employed as an ambulatory monitoring solution for gait analysis, e.g.,[GCMÓ08],[SFB+05],[SMSC05]. In addition, medical professionals can use this instrument to evaluate kinematic gait parameters throughout several steps and for a predetermined amount of time. There are a wide variety of clinical uses for the gait analysis system. Tools are used to rehabilitate and diagnose medical conditions and sports activities.

Therefore, we conducted the research and looked into the usefulness of a gait pattern using an Arduino Portenta H7 IMU, specifically an accelerometer. Furthermore, we will look into how a Recurrent Neural Network (RNN) can be used to predict gait injury from the data provided by the Arduino Portenta H7 IMU. The following sections will provide more in-depth coverage of the topics introduced earlier.

1.1.2 What are the challenge and which results are reached?

The most challenging aspects of this research are including describing the recording method, determining how speed should be represented based on gait terminology, and analyzing how the feet move while walking. All three of these tasks are interconnected.

A. It is challenging to employ an Arduino Portenta H7 IMU sensor equipped with an accelerometer to address the recolonization of the gait pattern due to the abundance of potential attachment spots. The body plane is another crucial feature that must be considered. It is not just the microcontroller itself, but the whole system, including the battery and how it communicates with the compiler (Cable, Infrared, Bluetooth, etc.)

B. Speed is the rate of change in any object's position. It is also a ratio between distance covered and time. In gates terminology, speed is the time needed to cover a short distance on flat ground. Consequently, it is difficult for us to determine the required distance and time. In addition, one must comprehend how accelerometers function because they measure in either meter per second squared (m/s^2) or G-forces (g). Consideration of their mathematical relationship presents us with a further obstacle.

C. Before working with sensor data, we must first analyze gait patterns, or feet's movement, to know which data and format to use. The goal is to use wavelet-based approaches to analyze accelerometer data and define the gait cycle stages. After that, we must examine speed, periodicity, and strength. Also, consider the body plane.

1.2 Positioning your Analytic in front of existing solution

Machine learning techniques typically consist of two main components after pre-processing the motion data include Feature extraction from the input signal in brief windows of the streaming data and Model training to construct a predictive model fed by the data at the feature space.

Gait recognition applications have made use of a wide range of modeling algorithms, including Bayesian network classifier [KWM10], hidden Markov model classifier [RZJM07], support vector machines [KWM10], and decision trees [KWM10]. The quality of these systems' predictions depends on the features used to build the hypothesis class. Noise and motion artifacts make it hard to extract important features from complex sensor data.

Due to the complexity of IMU sensor data, collecting manual features for machine learning-based systems is time-consuming, subjective, and prone to biases. Manual feature extraction can lead to poor feature set expressivity (i.e., the possible predictors with a fixed set of features may need to be better). Given the manual characteristics, the best model may have lower accuracy than the ideal performance in the representative feature subspace. Separating feature extraction and predictive model training also reduces machine learning-based methods' expressivity. In this approach, critical information for high-performance predictive modeling should be extracted.

Nonetheless, this motivates us to investigate other algorithms based on wearable equipment that supply more precision for gait recognition [DTC17], including the prediction function. One study as shown in Figure 1.1 provides a novel approach to human gait identification by using Time-Frequency (TF) expansion of human gait cycles to capture joint 2-dimensional (2D) spectral and temporal characteristics of gait cycles. Construct a Deep Convolutional Neural Network (DCNN) model that is optimized for discriminative purposes by training for each of the sensor nodes (i.e., five inertial sensors) and the modalities (i.e., accelerometer and Gyroscope readings) to extract unique signature patterns from the 2D expanded gait cycles.

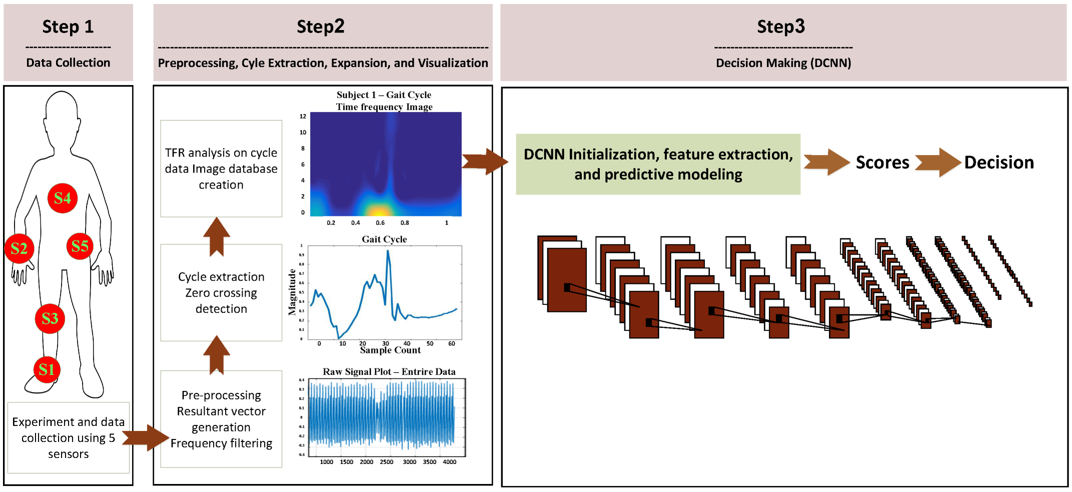


Figure 1.1: The overview of existing system for Human Gait Identification source: [DTC17]

When considering the state of the art, it could be used for the multiple application involved gait features when measured concurrently in both feet. On the other hand, Due to the fact that data from different sensors (i.e., sources) may have different characteristics, the DCNN training paradigm can be improved by sensor-dependent tuning for different sensor locations and modalities. Therefore, accuracy is not the only or primary concern for gait analysis, and other relevant shall be concerned include easily operational, less-space-consuming, non-expensive, and reliable.

For all of those reasons regard the disadvantage of the mentioned study, our primary objective was to identify a means of determining the gait phase that was not only more accurate but also required fewer resources and was less costly. Likewise, instead of using multiple sensors, Arduino Portenta H7 IMU shall be used to measure the acceleration of the foot. The use of a Neural Network as a model is going to be required in order to differentiate between gait phases and provide prediction in foot and ankle injuries.

2 Identify the Application Sector for your Analytic

Injuries prevention and return to play application

This section will highlight and investigate one relevant application fields for BTS Bioengineering's gait pattern recognition analysis. An Italian motion analysis company that creates turn-key 3D Motion Capture, tri-axial Force Plate, Video and wireless Electromyography (EMG), hardware and software systems for research, clinical and sports. They offer quantifiable, objective data and follow the precise progress of each movement with their fully integrated systems, which measure the mechanics and dynamics of human motion.

Figure 2.1: Injury prevention and return to play
source: [Mat22]

The primary goals of this application are injury prevention and the identification of optimal rehabilitation protocols in the event of injury [Mat22]. BTS Bioengineering's technologies and functional assessment tests are widely used by athletic trainers across several sports, for instance, MilanLab by AC Milan, PhysioeduLab, and Ripoll y de Prado Sport Clinic. Customers can keep track of the athlete's progress toward complete health by using the software's analysis techniques to monitor the athlete's progress and determine the optimal time for the athlete to return to playing competitive sports. As a result, the three primary features that stand out most about this application are the multiple-factor assessment service, the jump analysis, and the special protocols of particular sports movements.

3 What is your analytics doing

In accordance with what was discussed in chapter 1.2, we will advance the ongoing project by making it more financially viable while ensuring that it continues to operate effectively and dependably. However, our objective is to create a hardware system that is based on accelerometers. Four accelerometer sensors are attached to each shoe (both the left and the right) at the heel and the proximal region of the big toe. These sensors are also aligned to the mediolateral axis so that they may measure rotation in the sagittal plane. The hardware, as mentioned earlier, will be able to extract the four essential events of walking that occur in sequential order in the time format, which are Heel-Strike (HS), Toe-Strike (TS), Heel-Off (HO), and Toe-Off (TO), They will then be taken into account to diagnose and differentiate gait patterns. To this end, we'll present an explanation of our analytics' actions in a straightforward, two-part format:

3.1 Diagnosis of gait patterns

The recognition of human motion is the primary goal of the Diagnosis for gait pattern application, which makes use of the data provided by the accelerometer that is onboard with the Arduino Potenta H7 IMU. The step of diagnosing gait patterns can be broken down into two parts once we have collected the necessary data from the various sensors: the first part is called Data-Preparation, and the second part is called Gait Cycle Extraction.

Data-Preparation

The accelerometer readings include acceleration signals along all three axes as well as timing data. Nevertheless, the occurrence times of HS, TS, HO, and TO are plotted along the z-axis. Some unwanted information, like the gravitational component of vertical acceleration impulses, has yet to be processed and removed. Afterwards, the system records the elapsed time when the person's feet hit the ground. These intervals are linked to the heel and toe accelerations and are referred to as the "heel flat phase" and the "toe flat phase" in this work, , which will be discussed in greater detail in subsection 7.3.2.

Gait Cycle Extraction

Following the completion of the data preparation, we were able to acquire the time recorded data, which can be categorized as the "heel flat phase" and the "toe flat phase." Because of this, we are able to determine the gait cycle based on these data.

In addition to this, binary functions that represent the heel and toe are produced at this time scale. Consequently, the methodology that we have proposed makes use of the local information that is associated with the flat phase boundaries in order to extract the four gait events that are of interest from the time intervals in which the accelerometer moves by employing the Butterworth filter, it will be discussed in greater detail in subsection 7.4.1.

3.1.1 Differentiate of gait pattern

Differentiating the gait pattern is the next step after diagnosing the pattern, along with the data preparation and cycle extraction that came before it. Nevertheless, this analytic element seeks to provide the prediction of injury modelling. In order to offer the prediction, there are three primary processes in this analysis that need to be explained: Recurrent Neural Networks (RNN) initialization, Feature Extraction, and Predictive injury modelling. All of them will be discussed further below.

Recurrent Neural Networks (RNN) Initialization

RNNs are helpful for processing sequential data like natural language or time series. RNNs see patterns in sequential data well. Our gait parameters are time-based so that we can feed them into a recurrent neural network (RNN). We can examine gait trends and relationships over time. During pre-processing, data is segmented into more manageable portions. The RNN would then change its hidden internal state based on the current input and its previous state at each time step.

Feature extraction

In the context of modelling for predicting injuries, "feature extraction" refers to locating and extracting pertinent and important features from a dataset of gait patterns. The gait parameters (HS, TS, HO, and TO) are a group of the characteristics that shall be classified in this category. The purpose of feature extraction is to identify the most relevant and important features within a dataset to be used as inputs to a machine-learning model. Once the most relevant and important features have been identified, the feature extraction process is complete, it shall be be discussed further in subsection 7.5.

Predictive of injury modelling

An injury prediction model can be built using a machine learning model, such as a neural network, once the pertinent and important features have been identified and extracted from a person's gait. Gait patterns and injury outcomes are used to train a machine learning model. By precisely forecasting the probability of an injury, the system will be able to inform clinical decision-making and identify individuals who may be at an elevated risk of damage.

4 Where is your analytics in SRA?

Regarding on Standard Reference Architecture (SRA), The Reference Architecture Model Industrie 4.0 (RAMI 4.0) consists of a three-dimensional coordinate system containing the essential aspects of Industry 4.0. With this system the complexity between things can be reduced to more manageable units. The three axes represent all of the essential components of Industry 4.0, which consists of: Hierarchy Levels axis, Life Cycle Value Stream axis, and Layers axis [HR15].

In the beginning, we introduced RAMI 4.0, outlining its structure and the three axes for all the crucial parts of Industry 4.0. The RAMI 4.0 model employs these three dimensions as a framework for classifying serval objects, with all critical aspects of Industry 4.0 being mapped onto them. As shown in Figure 4.1, we can consider the following ways to map our gait injuries prediction system to the RAMI 4.0 framework as part of our analysis of Gait Pattern Recognition using an Arduino Portenta H7 IMU:

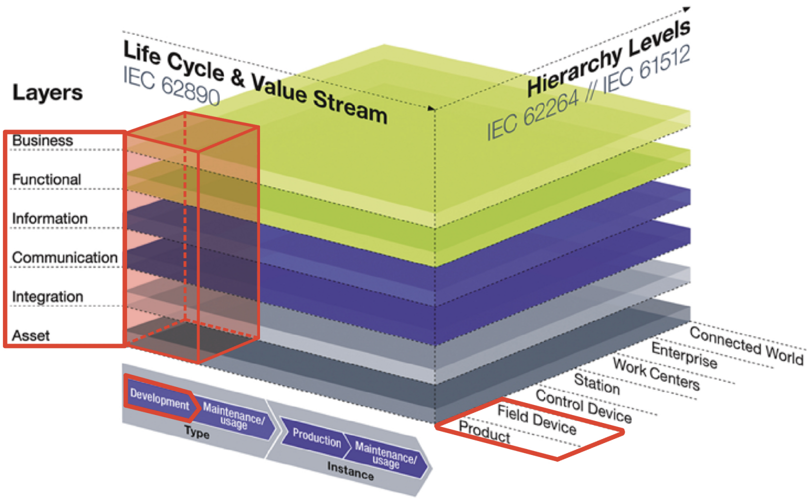


Figure 4.1: Position of Analytic Application in RAMI 4.0 Based on: [HR15]

Layers axis

- **Business layer**: This layer contains the system's business context, objectives, value proposition, and stakeholders. We sell a gait injury prediction system for hospitals, rehab centers, sports teams, and senior care institutions. We could also offer subscriptions or one-time payments.

- **Functional layer**: This layer contains the system's algorithms, procedures, and services. The gait injury prediction system's functional layer uses a Recurrent Neural Network (RNN) to detect and forecast gait-related injuries.
- **Information layer**: This layer stores and processes the system's data. The gait parameters dataset (HS, TS, HO, and TO) was obtained from the accelerometer and prepared by the Butterworth filter. The Recurrent Neural Network (RNN) will analyze this dataset to diagnose and predict injuries based on gait patterns.
- **Communication layer**: This layer describes the protocols and standards used by the system to communicate with other devices or systems. The communication layer of the gait injury prediction system consists of the various Bluetooth interfaces used by the Arduino Portenta H7 IMU.
- **Integration layer**: This layer shows how the system connects to other platforms and what standards and protocols are used. The integration layer connects the Arduino Portenta H7 IMU and the Recurrent Neural Network (RNN) training using TensorFlow lite in the gait injury prediction system (HS, TS, HO, TO).
- **Asset layer**: This layer contains the system's physical assets. The gait injury prediction system's asset layer includes the Arduino Portenta H7 microcontroller, the Accelerometer, the Bluetooth Module, the Lithium Polymer Battery (Li-Po), and any other hardware or software.

Life Cycle Value Stream axis

As Type data, the gait injury prediction system using the Arduino Portenta H7 IMU can be seen as mapping to the Life Cycle & Value Stream (IEC 62890) axis, which depicts the development and maintenance usage of the system.

- **Development**: At this point, the injury prediction system's idea and requirements are being defined. This includes the system's functions and capabilities. Moreover, the gait injury prediction system's hardware and software are also designed at this stage, including Arduino Portenta H7 microcontroller, accelerometer, algorithms, and software to diagnose and predict injuries based on gait patterns.

Hierarchy Levels axis

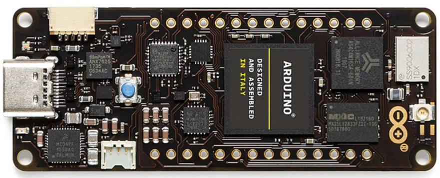
- **Product**: Gait injuries prediction system hardware and software components, including the Arduino Portenta H7 microcontroller, Accelerometer (Inertial Measurement Unit), Bluetooth Module, Li-Po battery, and any other hardware or software used by the system, are represented at the product level.
- **Field Product**: The field product level describes the implementation of a gait injury prediction system in the field, either as a self-contained unit or as a component of a larger, more comprehensive system or platform.

5 Which are the other components of the Infrastructure that will be connected to your Analytics

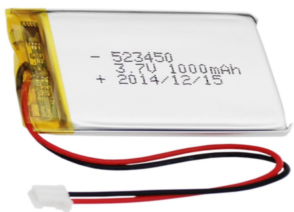
5.1 Arduino Portenta H7

Due to the presence of dual-core processors, the Arduino Portenta H7 is capable of doing high-level code execution in addition to real-time tasks simultaneously. For instance, you could simultaneously run code written in MicroPython and Arduino, and both cores could communicate to each other [Tea23].

There are two primary modes of operation for the Arduino Portenta H7: standalone and host. The Portenta H7 supports both of these modes of operation. It also offers several other features and capabilities that make it useful in various applications, including machine learning support, real-time processing, and low-power mode.



(a) Arduino Portenta H7
source: [Tea23]



(b) Li-Po Battery
source: [Wik23]

Figure 5.1: Arduino Portenta H7 and Li-Po Battery

5.2 Inertial Measurement Unit (On Board)

An inertial measurement unit (IMU) is an electronic device that measures and reports a body's specific force, angular rate, and sometimes the orientation of the body, using a combination of accelerometers, gyroscopes, and sometimes magnetometers.

IMUs are capable of measuring a wide range of parameters, including velocity, direction, acceleration, specific force, angular rate, and (if a magnetometer is present) the magnetic fields around the object. A variety of data kinds are captured with each tool in an IMU: Accelerometer, Gyroscope, and Magnetometer.

5.3 Bluetooth Module (On Board)

The Bluetooth modules allow for wireless data transmission and reception between two devices. Using the host controller interface, the Bluetooth module can receive and transmit data from a host system (HCI).

The Portenta H7's inbuilt Wi-Fi/Bluetooth module offers low energy Bluetooth functionality to give the board the flexibility to connect to gadgets that also support Bluetooth Low Energy, like the Arduino Nano 33 IoT or the majority of contemporary smartphones. Low Energy Bluetooth is designed to offer significantly lower power consumption and cost while keeping a comparable communication range when compared to conventional Bluetooth.

5.4 Lithium polymer battery 3.7V

One distinctive feature of 3.7V lipo batteries, a type of ternary lithium battery, over most lithium-ion batteries is their much lower weight [Wik23].

Furthermore, compared to its lithium-ion equivalents,

- It is much lighter and generally more powerful.
- Unusually high energy density.
- A broad variety of sizes and shapes are available in li-polymer batteries.

6 Identify the major infrastructure requirement for Analytics.

6.1 Structural requirements.

Environment.

The operation's primary goal is to record a participant's gait. Our study does not need to consider participant-impacting factors like temperature, wind, sound, etc. Given this and our desire to minimize process complexity, a Closed Environment is the best for our study.

Sensing device.

Gait identification using wearable motion sensors is a popular issue due to the widespread use of movement sensors. Most wearable motion sensors use Micro-Electromechanical Systems (MEMS) inertial sensors. These MEMS inertial sensors (accelerometers) are coupled to form IMUs. Electromechanical accelerometers measure acceleration forces along one, two, or three axes. IMUs are used for sophisticated motion analysis because they are mobile, compact, and powerful. We considered gait identification using wearable IMU because it's more effective.

Power source

The system's power supply is one of our study's key criteria. In our context, a wireless power supply to the research participant is required to keep the procedure simple. Batteries were the greatest choice as a result. Additionally, because the battery would be linked to the body, its size and weight might have an impact on how the person moves. Therefore, taking everything into account, a Li-Po battery with a 3.7V is best in our situation.

Processing device.

The central component of our structural need is the processing unit, or "The Brain," of our study. All types of tasks, including storing the acquired data, filtering the data, processing the data, and making decisions based on the processed data, will be carried out in this unit. Therefore, a computer with 8 Gb RAM, at least 500 Gb of storage, a monitor or display, and an 802.11ac 2.4/5 GHz wireless network adapter would be sufficient as a processing device.

6.2 Technological requirements.

As an ambulatory monitoring solution to handle the gait analysis, accelerometer-based devices have been suggested. Accelerometer-based methods to extract pertinent gait events and gait phases have been suggested in this situation.

- Basic technological requirements for data transmission include the transmitter and receiver modules that are built inside the Arduino Portenta H7 IMU in addition to the accelerometer..
- Software requirements are just as important to technological needs as hardware requirements are. Programming languages that can held TensorFlow libraries like Python, as well as other applications like the Arduino IDE and tools for graphical representation of the data that is provided.

6.3 Behavioural requirements.

Scenario.

The subject must be made system-ready as the first stage in the study. Two Arduino Portenta H7 IMU and a Li-po battery are firmly attached to each foot, or the right foot and left foot, one at the heel level and one at the forefoot. In order to prevent obstructing the subject's motions, the wires connecting the accelerometers and transmitter module (located at the waist level) were firmly secured around the legs. Once the subject is prepared, a walking surface—in this case, a treadmill—is set up in the room.

After the system has been established, the following phase is implementation. On the treadmill, the participant or subject is outfitted with hardware components for walking. In this context, steps per minute represent the walking speed. As soon as the subject begins to walk, the transmitter will transmit the data generated by the Arduino Portenta H7 IMU's sensors. Afterwards, the receiver will receive these data signals (Arduino on board). These acquired data are then stored in the processing unit with the aid of Arduino.

This data is processed using a variety of algorithms as we get closer and closer to the finish line, which ultimately helps in the decision-making process. The knowledge discovery procedure of the database serves as the foundation for the entire data processing system (KDD). Therefore, the entire analytics process begins with selecting the appropriate data, storing it, and finally preparing it. After the data have been adequately structured, they are sent on to the processing part of the Knowledge Discovery in Databases (KDD), which includes data transformation and data mining before the modelling and validation portions of the process. When sufficient information has been gathered, the user will be instructed to get off the treadmill and cut their connection to the foot sensors.

7 KDD Process

7.1 Topic's description

7.1.1 Problem's description

The objective of a gait pattern recognition system for injury prediction is to accurately identify patterns in an individual's gait (i.e., the way they walk) that may be indicative of an impending injury. This could involve analyzing various aspects of the gait cycle, such as stride length, cadence, and foot placement, and using various algorithm algorithms to identify patterns that are associated with increased risk of injury. The ultimate goal of such a system would be to enable early detection of injuries, potentially reducing the likelihood and severity of injuries for individuals who are at risk.

Inertial measurement units (IMUs) are used to measure motion and orientation. They often have accelerometers, gyroscopes, and magnetometers, but we only need the accelerometer data. In gait pattern recognition, IMUs measure the acceleration and angular velocity of body segments (e.g., lower leg, thigh) as a person walk. This data can extract gait cycle features, such as stride length, cadence, and foot placement, that can indicate an impending injury.

Therefore, using this gait analysis, the following fundamental injuries can be predicted: Balance disorders, Ankle Sprain, and Plantar Fasciitis. And, the target users for this gait pattern analysis are those working in healthcare, such as hospitals, rehabilitation centers, sports teams and athletic trainers, and nursing homes for the elderly. Nonetheless, it is critical to remember that gait pattern analysis is just one tool they can use to predict injuries, and it is not always full proof. In addition, this gait analysis system was designed with all requirements and considerations of RAMI4.0 in mind.

7.2 Database

7.2.1 What is the data

Acceleration signals along the three axes and timing data are measured in seconds and meters per second, respectively, for each gait phase. Three parameters can be used to evaluate the amount of data for a single research participant: the number of given steps, the number of Gait Phase Data sets, and the number of axes for each gait phase. These three parameters make up the total amount of data 7.1

$$Total\ Data = N_{Step} \times N_{Gait} \times N_{Axis} \tag{7.1}$$

where:
N_{Step} = Number of given step
N_{Gait} = Number of Gait Phase Data set (HS, TS, HO, TO), represented by 4.
N_{Axis} = Number of axis for each gait phase (X, Y, Z), represented by 6.

Number of given step

In a gait study, data volume and forecast accuracy must be balanced. Collect more data to learn stable, transferrable patterns for machine learning. Collecting enough data to make accurate projections is expensive and risky. Finding the optimal data volume-prediction accuracy balance may require experimentation.

Unfortunately, we're only starting analysis, not execution. An average-sized person will need 1,408 steps to walk one kilometer at a normal pace, according to a study in the Health & Fitness Journal of the American College of Sports Medicine (ACSM). Therefore, we decide to apply 1408 total steps to pur analysis. [HBR+08]

Number of Gait Phase Data sets

Gait phases will be analyzed as heel strike (HS), toe strike (TS), heel-off (HO), and toe-off (TO). By declaring 4 gait phase data sets, it is clear that data will be collected for each phase. This approach allows for a more detailed and comprehensive gait pattern analysis, which can help predict balance disorders or injuries like ankle sprains or plantar fasciitis.

Number of axes for each gait phase

By evaluating the x-axis acceleration data, the forward or backward body movement during gait can be determined. By evaluating acceleration data along the y-axis, one may identify body movement during walking. By evaluating z-axis acceleration data, one can tell whether the body moves uphill or downward during gait. Thus providing a more complete picture of the gait pattern that can be used to predict balance problems or injuries like ankle sprains or plantar fasciitis.

7.2.2 How to reach the data

The data is collected from the sensor using Bluetooth. Then in a Python environment we set the rate that the information will be taken out of the sensor, in our case 200Hz from the accelerometer and next we open a serial connection to the Portenta H7's serial port. Once this is done we gather the sensor data in an array in python with a time stamp so is possible to operate with the information as a function of the acceleration on each axis through time.

7.3 Data Selection

Data selection in the KDD process refers to the process of identifying and selecting the specific data that will be used in the analysis. This could include selecting specific records or subsets of the data based on certain criteria, such as the type of injury or the specific gait patterns being analyzed, Therefore in this section, we are describing the information regard two topic

7.3.1 Selection of injury's type

A person's walking pattern, or gait analysis, can be used to spot and treat a number of ailments and injuries. Three of the several instances include Balance disorders, Ankle Sprain, and Plantar Fasciitis

- **Balance disorders**:
  Several variations in walking style may result from a balance disorder. If a person has trouble keeping their balance while standing or walking, they may try to adjust their gait to feel more stable.

- **Ankle Sprain**:
  An ankle sprain is an injury that happens when the ligaments in the ankle are stretched or torn. This injury can occur when the ankle is suddenly turned or twisted or when the person lands awkwardly on the ankle. Sprained ankles can hurt, swell, and make it hard to walk or stand on the affected foot.

- **Plantar Fasciitis**
  Plantar fasciitis is a common cause of heel pain that is characterized by inflammation of the plantar fascia, a thick band of tissue that runs along the bottom of the foot from the heel to the toes. Plantar fasciitis can cause pain and difficulty walking, particularly first thing in the morning or after prolonged standing or walking periods.

7.3.2 Selection of Specific gait pattern

First of all, once we have the raw data, an analysis of it is not needed to be done directly on the first place, as it is possible to do a division between the steps that are being done by the same foot (as we are talking about the analysis of the signals made by the sensors of one foot only) by looking at the flat phases of the representation of $\ddot{z}_h$ or $\ddot{z}_t$, or by checking the periods where the value of one of them is 0, meaning that the heel or the toe in that moment it is touching the ground, this defining a "heel flat phase" or "toe flat phase" [BSS+15].

Secondly, in order to do a more extensive analysis we separate each step in 4 phases, HS, TS, HO and TO. This separation can help differentiate gait patterns between people, but not enough to discriminate all the features that describe different type of

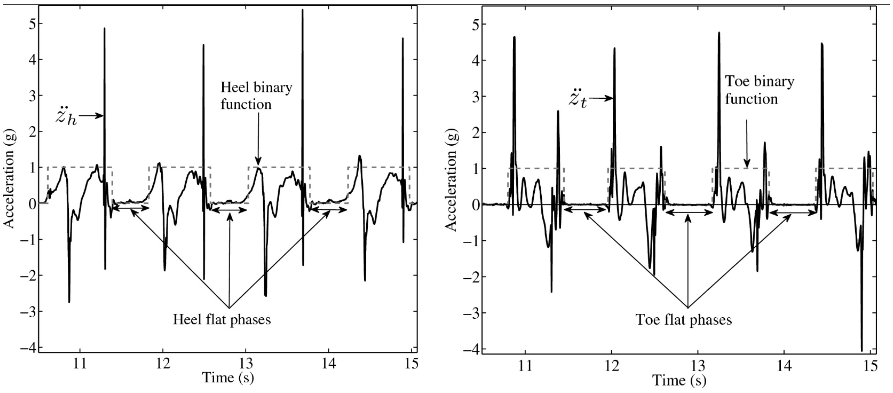


Figure 7.1: Flat phases represented by $\ddot{z}_h$ and $\ddot{z}_t$ respectively
source: [BSS+15]

injures or diseases, then we also add valuable data that can help distinguish them: The data measured by the accelerometer in all axis at the moment of each phase, thus helping to know the conditions of the foot at that specific instant, i.E: the straightness of the Heel Strike (HS).

7.4 Data Preparation / Outliers

7.4.1 Data Preparation

In the Knowledge Discovery in Databases (KDD) process, "data preparation" refers to the steps of cleaning, formatting, and organizing the data that will be used in the analysis. A few examples are eliminating duplicates, dealing with missing values, and normalizing the data range. In contrast, data preparation in the gait injury prediction system is the process that follows the collection of time-recorded data from research participants, which can be broken down into a "heel flat phase" and a "toe flat phase".

Consequently, gait cycle extraction will be performed to get the data ready for the Recurrent Neural Network (RNN). To do so, we must understand that the binary functions representing the heel and toe are produced at this time scale. Therefore, our proposed method employs the Butterworth filter to extract the four gait events of interest from the time intervals during which the accelerometer is in motion by making use of the local information associated with the flat phase boundaries. Each of the four gait events—heel strike (HS), toe strike (TS), heel-off (HO), and toe-off (TO)—can be defined in terms of their corresponding timing in the following way.

- **Heel Strike (HS)**
  HS is determined by filtering $\ddot{z}_h$[1] with a fourth-order zero-lag Butterworth high-pass filter (10 Hz cutoff). HS is the time of the maximum magnitude of

[1]The acceleration data in axis Z captured by the IMU sensor on the heel

the filtered $\ddot{z}_h{}^1$ in the heel non-flat phase. HS occurs quickly with a frequency greater than 10 Hz, so this filtering step was unnecessary.

- **Toe Strike (TS)**
  TS is found by observing the time at which the raw $\ddot{z}_t{}^2$ peaks at its highest and lowest points within the interval bounded by the HS and the lower boundary of the toe flat phase.

- **Heel-Off (HO)**
  Before defining the signal in a sub-interval, the vertical heel acceleration is divided using the toe binary function. The segmented heel signal is filtered using a fourth-order zero-lag Butterworth low-pass filter (20 Hz). The segmented acceleration signal is added twice to get its position signal in the sub-interval. This double integration's drift is limited by its short duration. To estimate the convex curvature in the position signal (the transition regions), we use a piecewise linear fitting method with two linear segments that best fit the signal in the least-squares sense. This procedure is done twice to improve the accuracy of this sub-interval. Lastly, it is assumed that the estimated HO will be at the time location of the convex curvature.

- **Toe-Off (TO)**
  When calculating TO, we take the midpoint between the upper boundary of the toe flat phase and the highest peak of the raw $\ddot{z}_t{}^2$ during the first half of the toe non-flat phase.

7.4.2 Outliers

When analyzing gait patterns using an accelerometer, outliers can happen for several reasons. Outliers may have a variety of causes, such as abrupt movements, improper accelerometer placement, and data processing errors.

A Long Short-Term Memory (LSTM) algorithm is one option for dealing with outliers in the data collected by the accelerometer in the Arduino Portenta H7 IMU for gait pattern analysis (Long Short-Term Memory). LSTM is a type of recurrent neural network that works well with time series data, such as the accelerometer's collected acceleration data.

In order to use LSTM to deal with outliers in acceleration data, one must first pre-process the data to eliminate any extreme values that could be considered outliers. This can be accomplished through the moving median technique.

[2] The acceleration data in axis Z captured by the IMU sensor on the toe

After pre-processing data, it can be fed into the LSTM model. The LSTM model will analyze the data and identify patterns or trends, which can then be used to identify and filter out remaining outliers. Hence, Using LSTM in conjunction with the moving median technique can efficiently handle outliers in the acceleration data collected by the Arduino Portenta H7 IMU for gait pattern analysis.

7.5 Data Transformation

In the KDD process, data transformation is the process of transforming the data into a format that is more suitable for further analysis, which in our case is a Recurrent neural network. Therefore, the gait injury prediction system's data transformation will include aggregation and feature extraction, which can help extract more meaningful patterns and trends from the data.

Regarding subsection: 7.2.1, we mentioned that the amount of data we are evaluating for a single research participant should be evaluated based on three parameters. Nonetheless, there are a total of 33,738 data items that must be reduced at some point. Consequently, we will use feature extraction to convert unprocessed data into numerical features that can be fed into a Recurrent Neural Network (RNN) while preserving the information in the original data set. In order to accomplish this, we suggest performing two calculations on the data before providing the data mapping. After performing two mentioned calculations, we therefore able to form the data set for individual research participant as follow table 7.1

- Average time between phases of the same gait phase: For calculate this, we take the information for the gait phases of one of the foot for then calculate the time that has passed between that phase and the next time the same phase take place in a iterative way till we have the sum of all of them, after this we need to divide into the total amount of steps (1408) to get the mean time. We repeat these process with both foot and all gait phases.

- Mean of the accelerometer info recorded when each stage of the same gait phase took place: Once again we take the information of the timings where one of the gait phases took place on one foot to then see the value of one axis in all of them and calculate the mean by dividing the sum into the number of recorder steps. We repeat these process with both foot calculating the mean velocity of the 4 phases for each axis.

7.6 Data Mining

Data mining refers to applying machine learning algorithms and techniques to the data to extract insights and identify patterns and trends. In the context of our injury

Data	
Parameter	*Unit*
PersonID	
Weight Height Age	kg m years
Gait Phase Mean	
HS_{mean} TS_{mean} HO_{mean} TO_{mean}	{s, s}
Velocity Xaxis Mean	
$V_{HS_{Xmean}}$ $V_{TS_{Xmean}}$ $V_{HO_{Xmean}}$ $V_{TO_{Xmean}}$	{m/s, m/s}
Velocity Yaxis Mean	
$V_{HS_{Ymean}}$ $V_{TS_{Ymean}}$ $V_{HO_{Ymean}}$ $V_{TO_{Ymean}}$	{m/s, m/s}
Velocity Zaxis Mean	
$V_{HS_{Zmean}}$ $V_{TS_{Zmean}}$ $V_{HO_{Zmean}}$ $V_{TO_{Zmean}}$	{m/s, m/s}

Table 7.1: Format of dataset for one research participant

detection system, we consider a Recurrent Neural Network (RNN) is the most suitable for the acceleration data to diagnose injuries based on gait patterns.

For this task, we are training the Recurrent Neural Network using the TensorFlow library in a Python environment. As input, we will provide the data formatted as shown in Table 7.1. About 100 individuals with each of the four output types (balance disorders, ankle sprain, plantar fasciitis, and normal gait) could be sufficient for a reliable prediction. For training the algorithm, we divided the data set into two batches: one containing 80% of the data and the other containing 20%. Now, to determine the number of hidden layers in our algorithm, we must consider the size and complexity of the input data, which in our case is extremely large and complex. However, excessive additions can lead to over fitness and slower training. Additionally, we must consider the available size in Portenta H7. Therefore, testing with varying quantities and sizes of layers will be required to determine the optimal combination.

To train the RNN we have to set the neuron data such as the activation function (i.E: for the output, a Sigmoid function activation will be the best for a Multi-class Clasification problem), then for the optimization a good optimizer for multi-class classification problems could be the Stochastic Gradient Descent (SGD) together with a Categorical cross-entropy loss function. We will require some experimentation to find the best configuration for the batch size and the number of epochs. Once the RNN is trained we test it using the test data set for then calculate the accuracy of the algorithm.

7.7 Model

The model in the KDD process refers to the machine learning model that has been trained on the data and is used to make predictions or identify patterns.

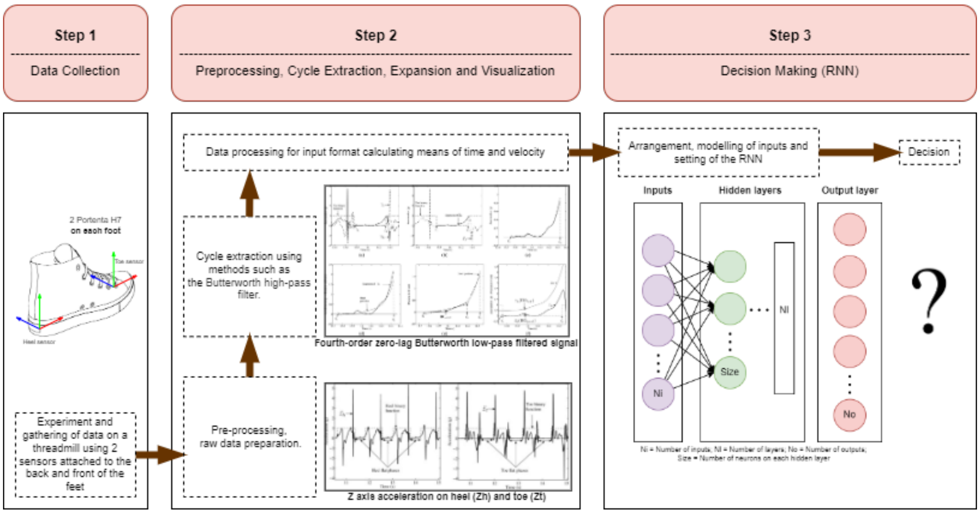


Figure 7.2: Summary Diagram

7.8 Validation/Verification

This would involve evaluating the performance of RNN with the test data, in our case a 20% of the data set (labeled data), using the sklearn library to show the accuracy of the algorithm predictions. It is also possible to validate it using additional data or methods, such as cross-validation, by splitting the data set into several training and testing sets, so the algorithm's accuracy is not biased by only one training data set, ensuring that it is reliable and accurate. If the accuracy is still low, another kind of readjustment must be done, like normalizing the data.

7.9 Model in Production

Model in production refers to the process of deploying the model in a real-world setting, in our system it involves using the RNN to predict injuries by taking measurements of people walking and then process it to get a diagnosis. For this task it is remarkable the scalability of the method, as we are operating with means it is possible to give more or less steps to get a more reliable output if needed. In terms of security, the information is mostly held in a close environment making it difficult to trespass it to get patient information. Finally, as only 4 sensors are being used, the maintenance part is quite simple, a malfunction of one of them can be easily perceived if too many outliers are being detected coming out of that output on the gait analysis phase.

8 Conclusion and Outlook

There are several ways to implement the gait parameter using sensors. Most consume sensors such as a gyroscope, accelerometer, pressure sensor pad, etc. Moreover, the research participant shall also need to attach the mentioned sensor all over the body, which can reduce the movement ability. Consequently, this can affect the quality of collected data, which can significantly affect the further analysis stage. Our approach aims to be more relaxed with the patient and use fewer resources, not affecting the outcome of the data.

We introduced the gait pattern analysis, consuming less investment in sensor usage but coming with the high ability of gait analysis function. As we launched the system that required only four Arduino Portenta H7, the research participant will no longer face the problem of unwieldiness. On the other hand, this can benefit our analysis, as we will be able to collect more precise data.

On the other hand, we need to execute the analysis physically. Therefore, we needed help accessing the actual world data leading us to face several obstacles, including defining the optimal amount of data for the neural network. Regarding the subsection 7.2.1, in order to be able to define the number of steps, we may require experimenting with different data volumes. Consequently, some neural network-related parameters could not be assigned because of this, making it also difficult to verify which setup will fit into the Arduino Portenta H7.

In conclusion, an additional neural network for fall detection that could be developed, trained, and incorporated into the Arduino devices in order to halt data collection and the treadmill in the event that a participant is about to fall would be an exciting prospect for the long-term development of our project.

Bibliography

[ADU+16] A. Ahmadi *et al.*, "3d human gait reconstruction and monitoring using body-worn inertial sensors and kinematic modeling," *IEEE Sensors Journal*, vol. 16, no. 24, pp. 8823–8831, 2016.

[BSS+15] M. Boutaayamou *et al.*, "Development and validation of an accelerometer-based method for quantifying gait events," *Medical Engineering & Physics*, vol. 37, no. 2, pp. 226–232, 2015, ISSN: 1350-4533. DOI: `https://doi.org/10.1016/j.medengphy.2015.01.001`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S135045331500003X`.

[DTC17] O. Dehzangi, M. Taherisadr, and R. ChangalVala, "Imu-based gait recognition using convolutional neural networks and multi-sensor fusion," *Sensors*, vol. 17, no. 12, p. 2735, 2017.

[GCMÓ08] A. Godfrey, R. Conway, D. Meagher, and G. ÓLaighin, "Direct measurement of human movement by accelerometry," *Medical engineering & physics*, vol. 30, no. 10, pp. 1364–1386, 2008.

[HBR+08] W. W. Hoeger, L. Bond, L. Ransdell, J. M. Shimon, and S. Merugu, "One-mile step count at walking and running speeds," *ACSM's Health & Fitness Journal*, vol. 12, no. 1, pp. 14–19, 2008.

[HKP+16] J. Hannink, T. Kautz, C. F. Pasluosta, K.-G. Gaßmann, J. Klucken, and B. M. Eskofier, "Sensor-based gait parameter extraction with deep convolutional neural networks," *IEEE journal of biomedical and health informatics*, vol. 21, no. 1, pp. 85–93, 2016.

[HR15] M. Hankel and B. Rexroth, "The reference architectural model industrie 4.0 (rami 4.0)," *Zvei*, vol. 2, no. 2, pp. 4–9, 2015.

[KWM10] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Cell phone-based biometric identification," in *2010 Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, IEEE, 2010, pp. 1–7.

[Mat22] S. I. Materia, *Injury prevention and return to play: Bts bioengineering*, Nov. 2022. [Online]. Available: `https://www.btsbioengineering.com/applications/injury-prevention-and-return-to-play/`.

[RZJM07] L. Rong, D. Zhiguo, Z. Jianzhong, and L. Ming, "Identification of individual walking patterns using gait acceleration," in *2007 1st international conference on bioinformatics and biomedical engineering*, IEEE, 2007, pp. 543–546.

[SFB+05] R. W. Selles, M. A. Formanoy, J. B. Bussmann, P. J. Janssens, and H. J. Stam, "Automated estimation of initial and terminal contact timing using accelerometers; development and validation in transtibial amputees and controls," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 13, no. 1, pp. 81–88, 2005.

[SMSC05] A. Sabatini, C. Martelloni, S. Scapellato, and F. Cavallo, "Assessment of walking features from foot inertial sensing," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 3, pp. 486–494, 2005. DOI: `10.1109/TBME.2004.840727`.

[Tea23] A. Team, *Portenta h7: Arduino documentation*, [Online; accessed Jauary 1, 2023], 2023. [Online]. Available: `https://docs.arduino.cc/hardware/portenta-h7`.

[Wik23] Wikipedia, *Lithium polymer battery — Wikipedia, the free encyclopedia*, `http://en.wikipedia.org/w/index.php?title=Lithium%20polymer%20battery&oldid=1123189045`, [Online; accessed 04-January-2023], 2023.