Configurable Medical Cyber-Physical System Framework for Physical Activity Monitoring

by

Faisal Arafsha

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements
for the Ph.D. degree in
Electrical and Computer Engineering

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

A digital twin facilitates the means to monitor, understand, and optimize the functions of the physical entity and provides continuous feedback to improve quality of life, and Medical Cyber-Physical Systems (MCPS) is an integral part of this vision. Many studies focus on human motion to digitize data for further analysis. The literature review presented here emphasizes on gait analysis and gait events detection using wearable devices, which compare results by testing on different groups of individuals. Amongst those, there is a focus on digitizing activities for athletes and sports activities.

However, there is a lack of research that address configurability for this type of MCPS. Adding new physical devices to an established MCPS requires manual configuration. Recent studies either solve the issue of users' mobility by providing a wireless solution with local storage, or sacrifice mobility in order to provide real-time information through wired communication. However, group physical activity applications, such as sports coaching and group physiotherapy, use customized devices that need to be automatically configured in the system. In addition, these systems need to support mobility and real-time data presentation.

To solve this problem, a framework is proposed to design a wellbeing Cyber-Physical System (CPS) that focuses on system configurability, providing real-time data of body sensor networks while supporting wireless and mobile communication. A communication protocol is proposed to allow seamless integration and communication of system components, and to enable bandwidth-conscious data transmission. As a proof of concept, a configurable CPS for gait activities monitoring is designed to read, visualize, and backup spatiotemporal data from one or more multi-sensory physical devices over conventional Wi-Fi and in real-time.

Two experiments were performed using the implemented CPS. The first experiment was performed outdoors and tested if the CPS components would recognize each other and work seamlessly over foreign networks while providing usable information. The second experiment was performed in collaboration with the Health Sciences Department using our system and the Tekscan Strideway gait mat simultaneously to compare results and to ensure accuracy. In addition, this experiment tested configurability of the system by using different measurement devices for different users.

Acknowledgement

I would like to express my sincere gratitude and appreciation to Dr. Abdulmotaleb El Saddik, for his patience and for his guidance throughout my PhD. Without your encouragement and continuous motivation and support, I would not have been able to complete this work.

Special thanks go to my closest colleagues, Basim Hafidh and Fedwa Laamarti, for always answering my questions even when it means interrupting their work, for their heartfelt thoughts, and for always treating my research as if it was theirs.

I am thankful to Dr. Sarah Fraser and her team in the Faculty of Health Sciences for their valuable engagement during the evaluation of my research, and for the opportunity to perform multi-disciplinary research collaboration.

I am grateful to Dr. Natalie Baddour and the NSERC CREATE-BEST team for being a part of my experience, and for providing me with the opportunity take my mind off of PhD when I needed. I am confident that this experience will open many doors for me in the future.

Finally, I would like to thank all my colleagues at the MCRLab and the DISCOVER lab, to my friends, to my family. Thank you for the hours and hours of constructive discussions and brainstorming, and also for those hours of procrastination and stress relief. We have created many memories during these years which I will cherish forever.

To my parents

Haifaa and Hani

for making every possible effort to get me where I am today

To my wife

Tarneem

for helping me during my research by taking care of everything else

This would not have been possible without you.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

2D	2-dimensional
3D	3-dimensional
AES	Advanced Encryption Standard
ADC	Analog to Digital Converter
AI	Artificial Intelligence
AT	Attention (Hayes AT commands)
AR	Augmented Reality
BSN	Body Sensor Network
BYOD	Bring your own device
CoM	Center of Mass
CoP	Center of Pressure
CSC	Cyber System Communication
CPS	Cyber-Physical System
DH	Data Handler
DB	Database
DBMS	Database Management System
DOF	Degrees of Freedom
ECC	Elliptic Curve Cryptography
XML	Extensible Markup Language
FF	Foot-flat
FSR	Force Sensitive Resistor
FTDI	Future Technology Devices International (serial interface)
GPIO	General Purpose Input/Output
GPS	Global Positioning System
HS	Heel-strike
HTTP	Hyper-text Transmission Protocol
IMU	Inertial Measurement Unit
IR	Infrared
IN	Insole, or SmartInsole©
IEEE	Institute of Electrical and Electronic Engineers
IDC	Interdigitated Capacitor
I2C	Inter-integrated Circuit
IoT	Internet of Things
IP	Internet Protocol
JSON	Java Script Object Notation
LED	Light Emitting Diode
LoA	Limit of Agreement

LiPoly	Lithium Polymer, also LiPo (battery)
MCPS	Medical Cyber-Physical System
MAL	Movement Analysis Lab
OGC	Open Geospatial Consortium
PDT	Packet Description Table
PAN	Personal Area Network
PC	Personal Computer
PSVP	Physical System Visualization Preset
PSC	Physical System Communication
PSDH	Physical System Data Holder
PSP	Physical System Profile
PWM	Pulse Width Modulation
RAM	Random Access Memory
SensorML	Sensor Model Language
SPI	Serial Peripheral Interface
TO	Toe-off
TCP	Transmission Control Protocol
UART	Universal Asynchronous Receiver-Transmitter
USB	Universal Serial Bus
UI	User Interface
Wi-Fi	Wireless Fidelity

Chapter 1 Introduction

The digital twin vision is introduced by Prof. El Saddik [1] as a digital replication of a living or non-living physical entity. By bridging the physical and the virtual worlds, data are transmitted seamlessly, allowing the virtual entity to exist simultaneously with the physical entity. A digital twin facilitates the means to monitor, understand, and optimize the functions of the physical entity and provides continuous feedback to improve quality of life and wellbeing. A digital twin is hence the convergence of several technologies such as AI, AR/VR and Haptics, IoT, Cybersecurity and communication networks.

A component of the digital twin vision is the ability to transform human physical movements and biological signals into digital information. This requires measurement of these activities through several wearable and embedded sensors around the body. Many wearable and non-wearable devices are already in the market, and are able to monitor detailed movement characteristics and read humans' biological signals. For example, fitness trackers can, in real-time, read the user's heart rate, GPS location, blood flow, oxygen levels, perspiration, and body temperature. In addition, they can accurately estimate step count, sleep time, sleep quality, calorie burn, and can extract many other useful information [2]–[4]. Several research studies are also developing wearable technologies for tracking the user's head and eye movements [5]–[7], as well as the upper and lower body movements [8]–[11] and gesture capturing [12], [13]. More recently, studies are moving towards designing devices aimed at measuring the pressure forces applied by the body limbs, such as smart gloves for hands [14], [15], and smart socks, shoes, and insoles for feet [16]–[19], in order to use the retrieved pressure information in medical and haptic applications.

1.1 Background

Cyber-Physical Systems (CPS) are systems engineered to seamlessly integrate computational components, networking, and physical processes in a well-defined context to serve a specific purpose. The motivation for the emergence of the concept of Cyber-Physical Systems is to enhance the overall system performance in comparison to the traditional integration of software systems and embedded computing systems or sensor networks [20], [21]. In an era of the largely trending research topic: "Smart Cities", the rapid advancement of sensor networks, information technology and high speed networking can only naturally evolve into highly sophisticated systems that attempt to overcome obstacles of assimilating them manually.

Cyber-Physical Systems can be considered as feedback systems that are interconnected in a well-defined network, with the possibility of having sensing or actuating devices at its endpoints [22]. Human interaction may be involved in the CPS design. However, adaptability, intelligence, and real-time data analysis and transmission are essential parts of any CPS. CPS designs also require implementations of solid cyber-security to protect privacy of data and to prevent malicious attacks. They require a design methodology that supports scalability and the ability to interface with legacy systems.

Cyber-Physical Systems have applications in communication, energy, health care, manufacturing, military, robotics, transportation, and many others [22]–[29]. The widespread employment of CPS concepts in different environments resulted in the emergence of specialized systems in industrial automation [30], manufacturing [31], vehicular systems [25], and medical cyber-physical systems (MCPS) which focus on many aspects of healthcare and wellbeing [28], [32]–[34].

1.2 Motivations

Some of the important qualities that distinguish Cyber-Physical Systems for different applications include mobility of its input and output entities, configurability of the different involved components, and its ability to provide useful data in real-time.

Mobility is the ability to move from one place to another at any time, using any means of transportation. In the context of digital technology, mobile users are able to move while staying connected and without an interruption of data transmission. It is imperative for the development of smart cities to be able to perform digital tasks anytime and anywhere. Furthermore, the concept of "Bring Your Own Device (BYOD)" is rapidly spreading, which means that users are not only able to move pre-configured devices, but can also use their own devices. In order to make this happen, a good mobility strategy is important in designing the smart system. However, there are involved costs and difficulties that need to be addressed. The design and deployment of mobile systems should take into consideration the network structures and protocols, access points, communication and presentation standards, privacy, security, and encryption, etc. Operating systems embedded in these end devices are also considered for encoding and presentation of data. Finally, mobile systems must consider the fact that sensitive information can be accessed, and possibly modified, from anywhere in the world. Therefore, secure data presentation and modification should be taken into account in the design of the mobility component of a CPS [35].

Configurability in a CPS includes two aspects: the ability for end-point devices to join the CPS automatically and stay connected, and their ability to transmit data seamlessly within the network. An example of configurability can be described using the logic in cellphone networks. When a cellphone is turned on, it is automatically registered to one of the nearby towers with the highest signal strength. After that, the user can move freely in between towers while the signal strengths of nearby towers are being monitored through trilateration algorithms. The cellular network uses this information to hand over the registered phone from one tower to the other based on the signal strength in order to maintain connectivity [36]. Furthermore, CPS end devices should be able to collect and transmit data seamlessly to one or more defined base stations. This process should be performed seamlessly and without interruption, especially for critical systems such as some medical CPS where lives depend on continuous, uninterrupted, data flow [37].

Finally, **real-time** presentation involves live data transmission, swift processing, and instantaneous visualization. Smart and lightweight network-wide programming must be used to achieve these tasks [38]. In addition to communication protocols, compression becomes very important such that transmitted packets become much lighter to allow for speedy delivery. Data processing, on both ends, should involve lightweight algorithms to allow for fast encryption and decryption of data. In addition, real-time visualization means fast calculation of required parameters must be performed. All of these points are vital for delivering information to the user interface in real-time.

Mobility, configurability, and real-time presentation characteristics in a CPS overlap in the communication requirement. All system end-point devices must follow a standard communication protocol to enable smooth mobility between access points, data transmission between system entities, and automatic recognition of newly added devices which enables system configurability (Figure 1). Therefore, a well-defined communication protocol, that considers all possible scenarios, is an important backbone in the design of a CPS and ensures flawless operation.

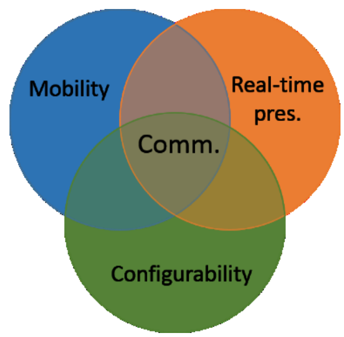

Figure 1: Mobility, configurability, and real-time presentation share the requirement of having a standard communication protocol for smooth system operation.

1.3 Research Objectives

Simultaneously monitoring multiple users' physical activity in sports and physiotherapy can be a challenging task, especially in fast-paced applications. Recording detailed foot kinetics and pressure points data can be beneficial in post-routine analysis such as sports-related performance enhancement or medical and rehabilitation diagnostics.

The objectives of this research is to provide a framework as a step towards standardizing the design of configurable Cyber-Physical Systems for physical activity monitoring. Having to monitor multiple mobile users in real-time means having to connect and read from several end-user health devices. For some applications, such as rigorous sports monitoring and devices affected by personal hygiene, these health devices may need to be replaced regularly. This research proposes a communication protocol that solves this problem and enables automatic configurability of new end-user health devices. The communication protocol also takes into consideration the mobility of these devices, and their ability to stream data in real-time.

1.4 Thesis Contributions

This research contributes to the literature through the following:

- A framework for designing configurable Medical Cyber-Physical systems that provides a solution for real-time visualization of sensory data from multiple sensor networks simultaneously. The proposed framework aims towards standardizing the design of cyber-physical systems for wellbeing applications and enabling auto-configuration of system components. Auto-configurability is tested and explained in chapters Chapter 3 and Chapter 5.The proposed framework follows a modular master-slave architecture, and includes a database management system that takes care of cloud or local storage of sensory data in a compatible manner.
- A communication protocol that includes a hand-shaking mechanism. The proposed communication protocol enables auto-configuration of new system devices added to the CPS. It also utilizes existing network protocols to create a seamless connection for data transmission between physical sensor networks, the main server, and the database. The proposed communication protocol automatically switches between TCP and UDP in order to minimize the stress on the network and to deliver sensory information to the main server in real-time for live data visualization.
- A taxonomy to categorize Medical Cyber-Physical Systems related to gait analysis and gait events detection for sports and wellbeing. The taxonomy helps researchers find a clear direction and purpose for the designed CPS and enables fair comparison with other systems. The proposed taxonomy also helps CPS designers understand the requirements for building such systems and the expectations that need to be accounted for.

1.5 Scholarly Achievements

Faisal Arafsha, Fedwa Laamarti, Abdulmotaleb El Saddik, "Cyber-Physical System Framework for Measurement and Analysis of Physical Activities", *Electronics*, 2019. (Submitted)

Fedwa Laamarti, **Faisal Arafsha**, Basim Hafidh, Abdulmotaleb El Saddik, "Automated Haptic Training System for Soccer Sprinting", *International Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2019. (Accepted)

Hawazin Badawi, Fedwa Laamarti, **Faisal Arafsha**, Abdulmotaleb El Saddik, "Standardizing a Shoe Insole as a Personal Health Device (PHD) Based on ISO/IEEE 11073 (X73) PHD Standards", *International Conference on Information Technology & Systems (ICITS)*, pp. 764-778, Springer, Cham, 2019.

Faisal Arafsha, Christina Hanna, Ahmed Aboualmagd, Sarah Fraser, Abdulmotaleb El Saddik, "Instrumented Wireless SmartInsole System for Mobile Gait Analysis: a validation pilot study with Tekscan Strideway", *Journal of Sensor and Actuator Networks (JSAN)* 7, no. 3, p. 36, 2018.

Faisal Arafsha, Fedwa Laamarti, Abdulmotaleb El Saddik, "Development of a Wireless CPS for Gait Parameters Measurement and Analysis", *International Instrumentation and Measurement Technology Conference (I2MTC)*, 2018.

Faisal Arafsha, Longyu Zhang, Haiwei Dong, and Abdulmotaleb El Saddik. "Contactless haptic feedback: state of the art" *IEEE International Symposium on Haptic, Audio and Visual Environments and Games (HAVE)*, 2015.

Faisal Arafsha, Kazi Masudul Alam, and Abdulmotaleb El Saddik. "Design and development of a user centric affective haptic jacket." *Multimedia Tools and Applications* 74, no. 9, pp. 3035-3052, 2015.

1.6 Thesis Organization

Chapter 2 shows a discussion of the current literature in gait measurement and analysis and provides a comparison of related studies. The proposed taxonomy is also explained in detail and an explanation of the technical features in the current literature is provided.

Chapter 3 introduces the proposed CPS framework explaining modules of the physical and the cyber parts. Then, the proposed communication protocol is described, and an operation scenario showing auto-configurability is explained.

Chapter 4 covers implementation details of the proposed framework showing circuit designs and implementation details of the multi-sensory health device, and the software design of the cyber system.

Chapter 5 shows two evaluation experiments to test mobility, configurability, data analysis, and data accuracy.

Chapter 6 concludes this thesis by providing a summary and a brief discussion and future work in the physical activity measurement research.

Chapter 2 Background and Related Works

2.1 Introduction

Medical Cyber-Physical Systems can be viewed as multilayer systems where the system user is the most important layer that defines which category is addressed by this CPS. The type of users this CPS is designed for defines which components are necessary in its design. Following that is the physical hardware which can be outlined to retrieve information from the system context. Sensing complexity varies from simple electronic sensors, to complex magnetic rendering machines. Once physical modules of the CPS are defined, system designers progressively increase the complexity of their design and add intelligence by defining calculations and soft-feedback in the system software. Communication also comes in as an integral part of the CPS, which depends on the setup requirements, range, mobility, etc. Finally, system feedback is added depending on the output requirements, and different feedback variations involve audio, visual, haptic, or a combination of more than one type.

There is a noticeable trend in recent studies, especially in the field of MCPS, showing a maneuver towards post-stroke and rehabilitation patients [15], [39]–[44]. Some of these systems also apply to wellbeing applications, such as training and feedback for sports [18], [45]–[48]. Many of these studies are interested in calculating several gait parameters wirelessly [42], [46], [49]–[51], and provide feedback in real-time [47], [50], [52]–[55]. In addition, logging and storage of historical data is a priority for these systems to enable post-routine analysis of acquired data [18], [51], [53], [56]–[59]. However, some studies reported adjusting or minimizing their sampling frequency because of issues with the amount of data being collected, which can affect the precision of the collected data, and to reduce power consumption, especially for wireless systems [28], [54], [60]–[63]. Therefore, more sophisticated and integrated systems are being developed to help solve some of these issues. These novel studies are gaining a lot of attention in the community of cyber-physical systems, and several transactions and conferences in engineering and in the medical/biomedical fields are paying close attention to them. For example, the IEEE

Instrumentation and Measurement Magazine recently published a review article by Postolache et al. that discusses balance sensing devices for physical activities [64]. Other transactions covering the topic include IEEE Journal of Biomedical and Health Informatics, Sensors Journals by MDPI and IEEE, IEEE Transactions in Biomedical Engineering, IEEE Transactions on Information Technology in Biomedicine, Journal of Biomechanics by Elsevier, and many others. From the medical side, journals including Foot, Gait and Posture, The Knee, and ports Orthopedics and Traumatology, also by Elsevier, are covering this topic in recent years.

In this literature review, we focus mostly on gait measurement and analysis related systems, which is a specialized category in the field of MCPS. The following sections start with a summarized literature review in plantar pressure sensing systems, systems with mixed sensing elements, fast human motion and sports related systems, and other systems with interesting or unique approaches. After that, a classification taxonomy is proposed to classify gait measurement and gait events detection systems as a specialized field in medical CPS. Then, a technical comparison of the covered literature is summarized. Finally, a discussion and some conclusions are derived from the related studies covered in this chapter.

2.2 Summary of Literature in Gait Measurement Systems

2.2.1 Plantar Pressure Sensing Systems

A number of studies have used accelerometer sensors to detect gait parameters. Godfrey et al. [51] attached an accelerometer sensor at the lower back of young and older adult subjects for purposes including counting steps and comparing gait characteristics between the two categories. This system was later validated by comparing gait of a Parkinson's disease patient and a healthy person [58]. Mota et. al also used a 3-axis accelerometer in addition to 3-axis gyroscope data to complement each other and send data to a computer over Wi-Fi [53]. They focus in their studies on angular movements by acquiring data from 5 embedded 6-DOF IMUs on the torso and the right thigh, leg, and foot. Their use of a Wi-Fi enabled microcontroller allows for data acquisition from users in an unrestricted setup, which shows potential for scalability and configurability and allows monitoring multiple users simultaneously over conventional wireless networks.

Howell et al. [65] developed two insoles to explore the optimum locations of FSR sensors; one including 32 FSR sensors, and another including 12 FSR sensors. The locations of the FSR sensors in the 12-sensor insole were driven by analyzing force data acquired by the 32-sensor insole. Optimum locations were determined from different tests, and the developed 12-sensor insole was then validated by comparing results with an established gait analysis system at the Physical Therapy Department in the University of Utah. Tests results on 6 controlled subjects and 4 stroke patients show high correlation between the data acquired from the developed insole system and the data acquired from the compared system, which validates the 12-sensor insole as a gait analysis system.

Tan et al., on the other hand developed an insole with piezoelectric transducers to measure and analyze gait and discussed the need for a low-cost wireless solution for that purpose [47], [49]. The focus here was on plantar pressure calculation using the designed system which contains a grid of 5x15 bespoke and low-cost piezoelectric sensors. They compare their results with the Kistler Force Plate which is used for similar purposes.

Other methods include a fabricated rubber insole developed by Motha et al. at Simon Fraser University in Canada [66] which focus on plantar pressure. They used interdigitated capacitors (IDC) as pressure sensors due to their high pressure sensitivity, amongst other reasons, as reported by [67]. The insole has embedded pressure sensors in 3 areas of the insole: forefoot, midfoot, and hindfoot. Their studies demonstrate how different postures present different responses from these three pressure areas. Crea et al. also designed an insole that embeds a grid of 64 pressure sensing elements that use LED and light sensor pairs to measure pressure distribution and center of pressure [52]. They also explain the possibility for estimating speed based on the distribution of detected pressure and the number of detected steps. The insole used in the research by Braun et al. uses 13 capacitive pressure sensors with a 3D accelerometer [57]. The system they used in their experiments records data internally in a flash storage. Data can, then, be loaded into a computer for analysis. Minto et al., on the other hand, created a body sensor network including 2 IMUs and ultrasonic sensors [50]. They report detailed spatiotemporal gait measurement such as ankle symmetry, stride length, ankle angle, foot trajectory, and more.

2.2.2 Systems with Mixed Sensing Elements

Hafidh et al. showcase the SmartInsole, a gait measurement device that embedded the complete circuit within the insole [46]. The embedded circuitry includes 5 FSR sensors, an accelerometer, a PICAXE microcontroller, a Bluetooth modem, and a 3.3V cell battery. Test results show clear patterns read by the accelerometer and pressure sensors to identify walking vs biking. In their study, they also explored a power-saving mechanism to improve battery life, which is an issue in such systems. The power saving module turns off all system components after 5 minutes of inactivity, and turns them back on when a WakeUp event is triggered. Zhuang et al. also designed a shoe insole with an embedded Bluetooth modem and an IMU with the goal of detecting and counting steps and step to step distance [68]. Since Bluetooth transmission consumes a large portion of the device's energy, they also implement a power-saving mechanism by reducing the transmission rate from the 96 Hz to 8 Hz sampling rate. They did a controlled experiment where 5 subjects were asked to walk at certain speeds and patterns. What's mostly interesting in their study is their use of the IMU to detect certain events (heel contact, foot flat, and toe off), and using the detection of these events to estimate step length. This is especially important in estimating gait velocity, which is still an ongoing research due to the difficulty of measuring walking distance with mobile devices in gait analysis. Their results were compared to Vicon motion analysis system [69] and show high accuracy with an average error of 2.89% for normal walking, and 3.90% for fast walking.

Jagos et al. developed the eSHOE which has all components embedded in the shoe. These components include a 3-axis accelerometer, and a 3-axis gyroscope, in addition to 4 FSR sensors [59]. Their main goal was to validate the eSHOE by comparing acquired gait parameters with the results obtained from the GAITRite walkway, which is a plantar pressure and gait analysis system developed by CIR Systems. The GAITRite is an electronic 6-meter mat with embedded pressure sensors capable of measuring spatiotemporal gait characteristics. Ten healthy patients, with an average age of 40.8 years, participated in their controlled experiment by wearing the eSHOE and walking over the electronic walkway. The researchers, including 3 physical medicine and rehabilitation specialists, analyzed parameters such as stride time, stance time, swing time, step time, and other parameters. Comparing results from both systems shows adequate similarity and high accuracy.

2.2.3 Fast Human Motion and Sports Related Studies

A group of researchers from RMIT University used piezoelectric transducers to measure the impact of kicking a soccer ball [70]. They placed the sensors on the expected impact on the upper surface area of the foot.

Other systems that do human movement analysis include the system developed by Buthe et al. [48], which focuses on timing analysis for tennis players using three IMUs; two attached to the right and left ankles of the player, and one handheld racket. Using these sensors, they were able to measure different activities related to the sport. Weizman et al. developed a system to measure the magnitude of kicking a soccer ball [70]. They used in house fabricated piezo resistive material to develop pressure sensors and measure kicking magnitude at a 10 kHz rate. The reason for the high sampling rate is that the duration a ball is in contact with the foot during a kicking action can be very brief (10 milliseconds [71]). In addition, they reported calculating the Centre of Pressure (CoP) using attached sensors.

2.2.4 Other Interesting and Unique Approaches

Rouhani et al. conducted a study on ankle kinetics to measure joints at multiple segments in the foot [56]. They used four IMU sensors to read 3D gyroscopic data and 3D acceleration data, and attached them on the toes, forefoot, hindfoot, and shank. In addition, they used pressure sensors embedded in an insole [72] to measure plantar pressure for a total of 22 subjects.

Edgar et al. developed a pair of shoes with 4 FSR sensors and a 3D accelerometer which transmits sensory data to a smartphone via Bluetooth [54]. The smartphone was programed to provide visualization that shows the intensity of the pressure on the 4 FSR sensors, as well as real-time generated graphs to show accelerometer data. They conducted a small study on 3 participants to classify postures and activities performed the system user: standing, walking, and sitting. The classification results were compared with previously developed systems by the same research group [43], [63] and showed >99% correlation. The system was later used by Lopez-Meyer et al. [42] to detect temporal gait parameters for analysis in post-stroke patients. They used the embedded FSR sensors to detect heel-strike and toe-off times to further investigate step characteristics such as step count and cadence.

2.3 Proposed Literature Taxonomy

A taxonomic classification is imperative in almost every research field. In the cyber-physical systems research area, different studies have different applications and provide details and specifications in an unorganized fashion which can be puzzling to the research community. In order to group, categorize, and organize systems based on similar characteristics, a well-defined taxonomic scheme is required. In this section, a classification taxonomy for related work in the field medical cyber-physical systems (MCPS) is proposed. The focus of this taxonomy is on distributed medical devices with embedded sensors and with the goal of monitoring users' physical and biological signals for health related applications. This classification is most beneficial to researchers and designers of new medical systems where it assists in finding a clear direction and purpose for their design. It helps provide a clear idea of the requirements and the objectives wished for by building such systems. In addition, many of the current research studies in the field of MCPS do not address all basic CPS requirements and specifications. The proposed taxonomy can solve this by helping future studies and researchers to properly and fairly compare, evaluate, and categorize systems. The taxonomy used here is influenced by several review articles and previous classifications [23], [24], [29]. However, this work builds on top of these studies and re-defines many of the described classification features. The new taxonomy is described here in detail, and was used to classify and review recent research related to CPS in healthcare and wellbeing.

Application: A healthcare CPS should define a specific context under which it operates to provide the greatest benefits. Some systems are specifically designed to operate under high-intensity conditions and require constant monitoring and immediate response. Examples of such systems are those designed for intensive care in hospitals. Other systems are designed for relatively relaxed situations such as retirement homes or sport activity monitoring, which may or may not require professional assistance depending on the defined context. The type of application can fall into two main categories: assisted living and controlled environment.

a. Assistive: Assisted living applications do not interfere with the user's independence, and enable access to health information to provide a better quality of life. This category covers systems that can be installed in houses or retirement homes, and aim to monitor health activities for the people living their normal lives and provide information when needed. Other systems, such as wearables, can be considered as portable and unrestricted systems that are designed for assisted living. They allow users to roam freely while collecting information that can be used to provide analytical data.

b. Controlled: Most healthcare systems are used in hospitals, while some can be found in schools, sports complexes, etc. These are considered controlled environments that need, and provide, immediate access to experts and professionals. Hospital systems can also be classified based on urgency. For example, alarms triggered by systems in an intensive care unit may require a faster response compared to alarms in normal hospital rooms.

Sensors: A CPS uses physical sensing devices, which can contain one or more sensors, to read and digitize biological and physiological data. In health-related applications, different sensing characteristics are used to classify the CPS.

a. Heterogeneity: Some applications require a single type of sensors to collect data from many users. Whether it is to make sense of a trend in that region, or to monitor simple physiological characteristics to provide feedback to each user, using a unified and focused sensing approach is classified here as homogeneous sensing. On the other hand, a CPS that collects various biological and/or physiological data is classified as a heterogeneous sensing CPS.

b. Method: Sensing devices can be active or passive. Active sensors require more power in order to produce signals to receive feedback from the body, while passive sensors can simply read biological signals from users without interference. Consequently, the sensing method in a CPS can be considered as a great factor in evaluating the efficiency and power consumption of that system.

c. Parameters: Each parameter monitored or analyzed by the system can use a single or multiple sensors. IMU's, for example, contain multiple sensors that can be used to measure one parameter: orientation. A combination of multiple sensors is used to provide a more accurate and precise result of a single type of feedback. Mastorakis et al. use a single infrared depth sensor in the Kinect camera for indoor fall detection [73], while Triantafyllou et al. use two (multiple) Kinects for the fall detection and other indoor activities [74].

Data Management: Different applications have different system requirements, thus the need for a cyber-physical system that provides a complete solution to fit that specific application. Depending on the type of application, there are multiple types of infrastructures, data requirements, and computation/communication compositions. The collected data need to be stored for further processing, and we explain several CPS approaches for data storing and processing.

a. Data requirement: Systems can be categorized based on the amount of data that the collect and process. Light data requirement systems collect simple health data such as heart rate, temperature, step-count, etc. while collecting and processing users' MRI or 3D data is considered heavy.

d. Data storage and data processing: Data can be stored and processed in distributed locations in the network, centrally in a local or a cloud server, or a hybrid infrastructure for storage and processing. Distributed infrastructures perform storage and/or processing all around the CPS network where each entity does its own processing and storage. Local infrastructures store and process all CPS data on a local server, while cloud infrastructures operates on an internet server. A hybrid data infrastructure performs some storage and processing within the network, and some in a central location.

Computation: Computation complexity and CPS feedback depend highly on the amount of data and the calculations involved in the system. Heavier and complex data require high performance hardware to be able to get the outputs and feedback intended by that system. In addition, heavier data could mean compromising high speed communications, which is a vital part in some healthcare and wellbeing systems.

a. Modeling: Modeling and visualization of CPS sensory data is an important part of the CPS. Healthcare providers need to be able to access data in an understandable and non-technical method. Even for activity monitoring in daily living applications, a user should be able to access his or her data in a timely fashion. Some systems perform modeling and visualization of sensory data automatically and in a real-time, while other systems require the user(s) to complete the monitored activity in order to make it possible to analyze recorded data by a professional.

b. Composition: The literature shows two types of system setups for performing computation and communication of collected data. Some require/allow data transmission and output computation to be done manually, while other systems perform the computation and communication automatically without user involvement.

Communication: Integrating a solid communication module within a system is one of the main elements that makes a system represent a CPS. The communication protocol(s) used to connect the CPS elements to each other can define the integrity and reliability of the whole system. Therefore, it is important to know and understand the method of communication and the security measures undertaken in the design of cyber-physical systems.

a. Medium: Some cyber-physical systems perform heavy data communication which could mean the need to compromise mobility by using wired communication mediums such as Ethernet or fiber optic (IEEE 802.3). Many systems utilize wireless communication mediums using low range PAN such as Bluetooth and ZigBee (IEEE 802.15), while others use higher range such as Wi-Fi (IEEE 802.11) in order to allow user to move more freely. However, wireless communication protocols and techniques differ in ranges, security, and robustness.

b. Encryption: Encryption of sensitive information is vital for cyber-physical systems, especially those related to private patient data. There are different approaches these systems implement to achieve high level security of private data. A network layer encryption at the physical level could produces higher levels of data security since raw traffic does not pass through any medium unencrypted, and data would only be decrypted where some processing is needed around the CPS. However, some systems perform data encryption at the cloud-level, which could rely on systems with higher encryption standards, but increase the risk of sniffing while data is transmitted.

System Feedback: The effectiveness of system feedback to caregivers is an important characteristic to assess the CPS. Many systems simply trigger an alarm when sensing parameter(s) reach a pre-defined threshold determined by the manufacturer. Other systems allow the medical staff to set these thresholds to one or more sensing parameters in the system. Feedback effectiveness can be assessed by looking into two main components: decision-making algorithm and alarm-triggering mechanism.

a. Decision-making algorithm: Healthcare systems mostly depend on a single parameter to trigger medical staff attention. Triggered alarms usually require an immediate response from the healthcare provider to save a patient's life or to provide assistance [29]. This could lead to worker's fatigue [75], increase in false alarms [76], or even mistakenly ignoring/disabling critical alarms because of the high number of false ones [77], [78]. Therefore, multiple-parameter feedback or multiple-level alarms could provide a better solution to assist the healthcare worker in assessing the situation and making a decision whether or not the situation requires an immediate response.

b. Alarm-triggering mechanism: A great number of research studies have been made to improve the alarm-triggering mechanisms hoping to solve the number of false alarms in healthcare systems [60], [79], [80]. Tang et al. proposed a mechanism that allows caregivers to assess the "battle ground situation" and manually define alarm triggering mechanisms tailored for specific situations [81].

Table 1: Proposed Taxonomy Mapping Description

Application	Assistive	In-home	Subject is independent. Application can be installed in-home
		Elderly living	Subject is independent. Application requires some professional assistance
		Unrestricted	Subject is independent. Application is unrestricted and operates portably
	Controlled	Hospital	Medical support must be available
		Intensive care	Intensive care unit in hospital
Sensors	Modality	Homogeneuous	Single sensor per person, covering a number of users (example: HR for all users)
		Heterogeneous	Multiple sensors per person (example: HR and glucose level for each user)
	Method	Active	Such as ECG or Ultrasound
		Passive	Such as temperature or FSR
	Paramters	Single	Each parameter requries a single sensor
		Multiple	Multiple sensors to calculate one or more parameters
Data Management	Data requirement	Light	Simple data such as temperature
		Heavy	Large data such as MRI or 3D data
	Data storage	Local	Data is stored on a centralized local server
		Cloud	Data is stored on a centralized cloud server
		Distributed	Data is stored in multple locations within the CPS
		Hybrid	Some data stored centrally, while some is distributed
	Data processing	Local	Data is processed on a centralized local server
		Cloud	Data is processed on a centralized cloud server
		Distributed	Data is processed in multple locations within the CPS
		Hybrid	Some data processed centrally, while some is distributed
Computation	Modeling	Real-time	Modeling or visualization of data happens dynamically and in real-time
		Analysis	Modeling or visualization is created by analysis of recorded data
	Composition	Manual	Computation and communication processed manually
		Automatic	Computation and communication processed Automatically
Communication	Medium	Wired	Wired communication medium (Ethernet, Fiber optic, coax, etc.)
		Wireless	Wireless communication medium (WiFi, Bluetooth, ZigBee, etc.)
	Encryption	User level	Encryption of sensitive data happens within the CPS
		Network level	Encryption happens through the network befoe sending data to the cloud storage
System Feedback	Decision making	Single paramater	Threshold alarm
		Multiple parameters	Multiple levels or multiple combined signals to trigger decision making
	Mechanism	Manual	Caregivers can assess and manipulate alarm-triggering mechanisms
		Automatic	Alarms thresholds and triggers are set automatically by the CPS

2.4 Literature Technical Comparison

Applications of different wellbeing systems highly depend on the technical design and the hardware and electronics used in the system. In order to develop new and improved systems in the future, it is useful to be able to compare the different implementations of the discussed studies from a technical point of view. In this section, a technical comparison of the literature is provided, along with the most important and comparable features disclosed by these systems. The features in this section's comparison is described in the below paragraphs.

2.4.1 Technical and Feedback Features

Sensing parameters: some systems use a single parameter and derive multiple feedback variables and outputs, while other systems use multiple parameters to provide more complicated results and analysis.

Number of sensing elements: this feature compares between the number of sensors or sensing elements included in the calculation of the feedback and the outputs provided by the system. The literature show that many systems use fewer sensors, while providing more accurate and sophisticated analysis. This shows that the higher number of sensors included in the design does not necessarily mean more accurate or better results.

Sensor types: most studies covered in the literature show that FSR sensors are popular amongst researchers. This is due to their low cost, high accuracy, and durability. This can be quite useful for applications involving rough handling such as sports related applications. However, this comparison shows that some studies use different types of sensors to extract the same feature or event. For example, it can be seen from the literature that, to detect the number of steps, some studies use FSR, while others use IMUs.

Microcontroller: depending on the type of application and the degree of tolerated bulkiness of the physical device, different microcontrollers and/or computers were used to extract sensory data and to store or visualize them.

Communication: this feature shows the different communication methods implemented in the studied systems, and how data is transferred to its final destination for analysis or visualization. Many systems use Bluetooth as the main transmission method. However, most of these systems have a 1-to-1 connection between the Bluetooth device and the sensor network's microcontroller. Very few systems use Wi-Fi as their main method of data transfer. This could be due to the high power consumption of Wi-Fi modules. The communication method can be an important factor in classifying the gait monitoring system, and can highly enhance or restrict data transfer and mobility of the system users.

Data Storage Location: many of the researched studies use flash storage or some other means of local storage for sensory data. In some systems, local storage is used as a backup in case wireless transmission is affected or interrupted, while other systems collect data on the receiving device, such as a smartphone or a laptop. On the other hand, some systems completely depend on cloud servers to store sensory data for later assessment and/or for live feedback.

System Feedback: the type of feedback varies between the implemented systems. Some systems provide live feedback by visualizing sensory data, while other systems store sensory data during the tests in a local storage or on the remote computer to be analyzed at a later time.

Reported Outcomes: many studies provide plantar pressure and spatiotemporal gait parameters outcomes of their systems However, some systems simply provide a comparison between two gait classes or speeds, while others report a single parameter as their main focus.

Sampling rate: higher sampling frequency means more accurate results. However, it also means higher power consumption and higher data transmission and storage requirement. This comparison shows the different sampling frequencies used by different systems to enable comparison of the accuracy of their outcomes, and if the accuracy can be directly associated with the sampling rate.

Gait analysis: while most systems report doing gait analysis, or at least the possibility of doing gait analysis through acquired sensory data, some systems focus on other results. This feature, and the remaining features, are described with either "yes" or "no".

Center of Pressure: since most systems report plantar pressure as an outcome, it would be useful to see how each of these systems calculate the Center of Pressure using the different sensors embedded in them.

Event detection: different events are explored in this literature review. Some of the most common and interesting events are compared. These events include detecting the *number of steps*, *gait cadence*, *walking direction*, and *kicking a ball* or contact with a physical object.

Configurability: finally, configurability determines if the implemented system is scalable and easy to use as a plug-and-play system. This is useful in the case of an update to the software, or most importantly, if more sensors are plugged in or more users are to be monitored using the same software/system. Configurability feature shows if the system can easily handle more users simultaneously.

2.4.2 Comparison Summary

The comparison features described in section 2.4.1 can be divided into 2 groups: technical specifications, and feedback features. This section shows a summary of comparison of related works using these two groups of features. Table 2 shows a summary of comparison using the first group of features: technical specifications, and Table 3 shows a comparison of the same works using the second croup of features: feedback features.

Table 2: Technical specifications comparison between different implementations of gait monitoring systems in recent research.

Work	Year	Sensing Parameters	# of sensing elements	Sampling rate (Hz)	Sensor types	Microcontroller	Comm.	Data storage location	System Feedback
Lopez-Meyer et al. Edgar et al.	2010	single	8	25	FSR, 3D Accel.	16-bit RISC	Bluetooth	Smartphone	Heatmap, Real-time Accel. graphs
Howell et al.	2013	single	12	118	FSR	n/a	Bluetooth	PC	n/a
Hafidh et al.	2013	multiple	2	n/a	FSR, Accelerometer	PICAXE	Bluetooth	Smartphone	n/a
Crea et al.	2014	single	64	100	Fabricated pressure sensors using LEDs	STM32F103x8	Bluetooth	PC	LabVIEW
Rouhani et al.	2014	multiple	123	200	Pressure sensors, 3D Accel, 3D Gyro	n/a	n/a	Local	manual post-routine analysis
Tan et al.	2015	single	5x15 grid	12	Bespoke Piezoelectric	TEENSY 3.1	Bluetooth	n/a	Real-time heatmap
Motha et al.	2015	single	3	n/a	Bespoke capacitive pressure sensors	n/a	Wired	n/a	LabVIEW
Gerlach et al.	2015	single	6	n/a	Fabricated FSR	PC	Wired	n/a	Manual post-routine analysis
Weizman et al.	2015	single	4x4 grid	2.5k	Bespoke piezoresistive material	Teensyduino (ATmega328p)	Wired	PC	n/a
Braun et al.	2015	multiple	17	50	Capacitive pressure sensors,3D Accel, Temp.	n/a	Wired	Local	Manual post-routine analysis
Godfrey et al.	2016	single	3	100	3D Accelerometer	n/a	Wired	Local	Manual post-routine analysis
Zhuang et al.	2016	multiple	6	96	3D IMU	n/a	Bluetooth	PC	n/a
Minto et al.	2016	multiple	20	500	Pressure sensors, 6-DOF IMU, 3D Accel., Ultrasonic transducer	n/a	Wi-Fi	MicroSD	Event-driven audio-tactile feedback
Jagos et al.	2017	multiple	10	200	FSR, 3D IMU	16-bit	Bluetooth	Local	Manual post-routine analysis
Mota et al.	2017	multiple	30	n/a	6-DOF IMU	ESP8266	Wi-Fi	PC	Manual post-routine analysis
Proposed system	**2018**	**multiple**	**18**	**50**	**FSR, 3D IMU**	**ESP8266**	**Wi-Fi**	**Cloud server**	**Real-time heatmap and 3D visualization**

Table 3: Feedback features comparison between different implementations of gait monitoring systems in recent research

Work	Reported outcomes	Gait analysis	Center of Pressure (CoP)	Event detection: Number of steps	Event detection: Cadence	Event detection: Walking Direction	Event detection: Kicking	Real-time Vis.	Configurability
Lopez-Meyer et al. Edgar et al.	Posture and activity classification	yes	no	yes	no	no	no	yes	no
Howell et al.	Plantar pressure	yes	n/a	n/a	no	no	no	n/a	no
Hafidh et al.	Walking vs biking graphs	no	no	no	no	no	no	n/a	no
Crea et al.	Number of steps. Center of pressure	yes	yes	yes	yes, manual	no	no	no	no
Rouhani et al.	Plantar pressure, 3D ankle joints movements	yes	yes	n/a	no	no	n/a	no	no
Tan et al.	Plantar pressure. Center of pressure	no	yes	no	no	no	no	yes	no
Motha et al.	Plantar pressure	no	no	no	no	no	no	yes	no
Gerlach et al.	Plantar pressure	yes	no	yes	no	yes	no	no	no
Weizman et al.	Kick magnitude	no	yes	no	no	no	yes	n/a	n/a
Braun et al.	peak pressures, pressure distribution, acceleration, motion sequences, gait patterns and temperature	yes	no	yes	no	no	no	no	no
Godfrey et al.	Spatiotemporal gait parameters, Step count	yes	no	yes	no	no	no	no	n/a
Zhuang et al.	Step-by-step gait parameters, walking speed	yes	no	yes	yes	no	no	n/a	no
Minto et al.	Spatiotemporal gait parameters	yes	no	yes	yes	no	no	no	no
Jagos et al.	Spatiotemporal gait parameters, Step count	yes	no	n/a	no	no	no	no	no
Mota et al.	Spatiotemporal gait analysis of angular movements	yes	no	no	no	no	no	no	no
Proposed system	**Spatiotemporal gait parameters, Step count, walking speed, walking direction, & Plantar pressure**	**yes**	**yes**	**yes**	**yes, manual**	**yes**	**yes**	**yes**	**yes**

2.5 Discussion

In this section, a discussion of the surveyed studies and implementations is presented with a focus on communication, sensors and hardware implementation, and general technical specs. A proposed system implementation is then described and discussed at the end.

2.5.1 Communication

Different systems are available to assist in gait measurement and analysis applications. Some lack accurate measurements and can only provide a general idea of the wellbeing of the user's gait. Other systems use wired devices to record sensory data [56], [65], [82], [83] which introduces drawbacks such as limiting the exercise/therapy area [45], or requiring bulky and expensive setups [56]. Wireless devices record data using Bluetooth [48], [54], [68], [84] or ZigBee [39], [85], which although consume very low power compared to Wi-Fi, have low range capabilities [84] and, thus, limit the user's free movement area. In addition, these technologies limit the number of connections a system can read from (Bluetooth and ZigBee are typically one-to-one connections). However, attempts have been made to time-share multiple Bluetooth connections while compromising stream bandwidth [86].

The systems designed by Tan et al. shows a low-cost method to measure plantar pressure [45], [47], [49]. However, the built system communicates via Bluetooth, which limit the measurement area. The hardware design by Motha et al. communicates using a wired method directly to the computer, and visualizes sensor readings on LabVIEW [66].

Some of the studies that use Wi-Fi as their data transmission medium have high potential for scalability and allow monitoring multiple users simultaneously. This is especially important for sports related applications where a trainer is able to monitor several athletes' activities form a single computing device. However, it is unclear if Wi-Fi-based systems have performed such tests.

2.5.2 Hardware and Technical Specs

Some studies show attempts at developing customized pressure sensors for specific applications. Gerlach et al., for instance, designed a printed pressure sensor system for flexible plantar pressure measurements [87]. Their tests explore the system's functionality and ability to distinguish between healthy and unhealthy walking patterns using only FSR sensors.

It is shown that most of the studies in the literature use some sort of force sensors to measure plantar pressure, which is deemed necessary in gait monitoring and analysis systems. A thesis by Adam Howell [83] explores the validity of using 12 FSR sensors as opposed to 32 FSR sensors. His tests were performed on controlled subjects as well as patients, and results were compared with results produced by the Movement Analysis Lab (MAL). The MAL is currently operating at Shriners Hospitals for Children (SHC) in Salt Lake City, Utah. To analyze gait for physical therapy applications, the lab uses “ten infrared motion capture cameras (Vicon, Oxford, U.K.) and two OR6 series multi-axis force plates (Advanced Mechanical Technology, Inc., Water- town, MA)” [65]. Results of Howell’s study concluded that the 12-sensor insole produces results with high correlation with results produced by the MAL, and validates a 12-sensor insole system to be used for gait analysis.

As seen in Table 2, the number of sensing elements varies for systems with a single or multiple sensing parameter. In addition, the sampling frequency ranges between 12 Hz (Tan et al. [49]) and 2.5kHz (Weizman et al. [70]). The implied hypothesis behind this is that different system implementations use different numbers of sensors and sampling frequencies depending on the required measurement resolution. However, some studies have reported decreasing/adapting their sampling frequencies for reasons such as reducing the consumed power or the high processing requirements for real-time applications [28], [54], [60]–[63]. The types of sensors used also varies depending on the type of application. For example, all systems that report plantar pressure information use some type of pressure measurement sensor. Most of these systems, however, report using FSR sensors, which are known for their high accuracy and durability [88], while some report fabricating their own pressure measurement sensors [49], [52], [66], [70], [87]. This comparison also shows that all systems reporting the capturing of physical movements, which require monitoring joint movements, use some sort of inertial measurement sensors such as an IMU or an accelerometer. Godfrey et al. [51], for example, used a 3D accelerometer, to report biomechanical analysis of gait, while Mota et al. used five IMU sensors, each reading from two 3D sensors (accelerometer and gyroscope) to perform a more detailed biomechanical data capturing and analysis of gait and transmit this information over Wi-Fi [53]. Table 2 also shows that, in terms of data storage, most wireless systems transmit data from the measurement device or the BSN to a PC or a nearby smartphone (Bluetooth systems) where they are stored for future analysis [54], [65], [68], [83], while wired systems store data locally in the measurement device [51], [56], [57]. Finally, two systems (Lopez-Meyer et al. [42] and Tan et al. [49]) report measured information in real-time, while others store measurements for post-routine analysis.

2.5.3 Proposed Solution

Since providing real-time feedback to users is essential in many medical applications, our proposed system takes this into consideration. The proposed approach in this thesis also uses FSR sensors for plantar pressure measurements, and an IMU, with a 3D accelerometer and a 3D gyroscope, to read 3D orientation of the foot in real-time. Various types of microcontrollers are used in the literature. In our proposed implementation, the ability of reading from multiple devices simultaneously is imperative. Although many studies have used Bluetooth to transmit data wirelessly, there is still the issue of reading from numerous devices simultaneously and in real-time, which is a problem that is not addressed in the compared studies. For these reasons, an ESP8266 microcontroller is used (similar to Mota et al. [53]) for its small size, Wi-Fi capabilities, and its relatively low power consumption.

Table 3 shows a summary comparing feedback features for gait monitoring systems in recent research. Every research provides a set of results that define the main application for that system. However, when we focus on reported feedback, all of the surveyed studies can provide at least one of the following specific outcomes: gait analysis, Center of Pressure (CoP), and some sort of event detection such as the number of steps, cadence, walking direction, and/or a soccer ball kick. In order to fairly compare the proposed solution to the current literature, all of these features are implemented and provided in the same system. However, one of the main contributions in this research is configurability, which is the ability of new devices or BSNs to join the system seamlessly and automatically. Table 3 shows that none of the surveyed studies enable configurability as part of their CPS.

In conclusion, the proposed system combines all the strengths of the aforementioned studies. It has the ability to read sensory information wirelessly, monitor numerous devices simultaneously, provides information and feedback in real-time, and stores historical data in a cloud server for future analysis. Similarly, as shown in Table 3, the proposed system in this research combines and provides all the feedback features in the surveyed literature, in addition to enabling configurability, which is a main contribution in this thesis.

2.6 Chapter Summary

Chapter 2 discussed the current literature in gait measurement systems and studies by grouping them by the methods and approaches used for measurement. After that, the proposed taxonomy is explained in detail clarifying the categorization features and their sub-categories. Then, an explanation of the technical features in the current literature is provided, followed by two technical comparison tables where the proposed system and its features and some of its contributions are highlighted. Finally, the literature and the comparison tables are discussed to provide some insight on the current direction of CPS design for gait monitoring and analysis.

Chapter 3 introduces the proposed CPS framework explaining modules of the physical and the cyber parts. Then, the proposed communication protocol is described, and an operation scenario showing auto-configurability is explained.

Chapter 3 Configurable MCPS Framework for Activity Monitoring

3.1 Introduction

The proposed framework is an abstract model for the design of cyber-physical systems for physical activities that enables continuous delivery and storage of activity sensory data. Modularity will enable designers to modify any part of the CPS without affecting the remaining modules and the overall operation of the system.

An input-output system that contains software and hardware components, collects historical data, and defines clear communication protocols can be considered a CPS (see Figure 2). CPS implementations in general, and proposed frameworks in the literature, use this general model, in some sense, and tailor the different blocks for specific applications, whether it is for healthcare [89], wellbeing, medical CPS [90], industrial CPS [30], or vehicular CPS [91], etc. However, none of these framework define a clear structure to standardize the design of cyber-physical systems in such a way that it becomes configurable and adaptable to future modifications of parts of the system. This is the main motivation for the design of the proposed framework.

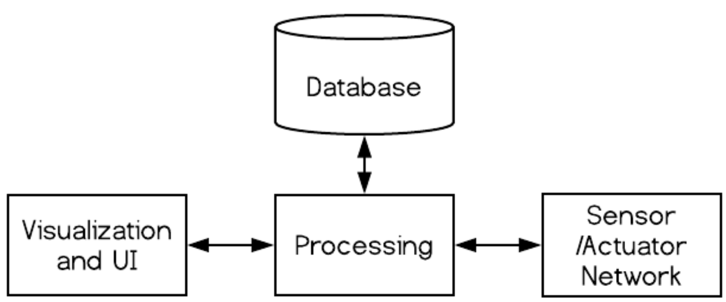

Figure 2 General concept of an input-output system with historical data collection

A modern description of a CPS extends the general concept described above. A CPS must perform distributed processing, which lowers the load on the central processing server to the minimum. Processing loads in a CPS application must be distributed within all system entities, and target an optimal dissemination solution, with the goal of minimizing power consumption while delivering high performance and quality input and output handling. Figure 3 explains this idea which involves multiple sensor networks with their own processing units and inputs/outputs. It also shows the central master processing component, cloud or remote storage, and the real-time input/output terminal that can be used by the human users to interact with the CPS. Each of the sensor networks acquire information from the environments and perform the processing that can be done at this level. This includes data acquisition, conversion, encoding, and compression. Handling sensory information at the lower-level physical network ensures optimal packaging of data which, in turn, lowers the load on the communication units in the network. The master processing unit can then receive compressed and encoded messages that contain sensory information from all the sensor networks. This master unit is in-charge of decoding this information and performing additional processing to produce user-friendly information such as visualization of sensory data and calculations that deliver more information based on these sensory information. In addition, the master unit prepares this information and communicates with the remote storage facility for historical data collection.

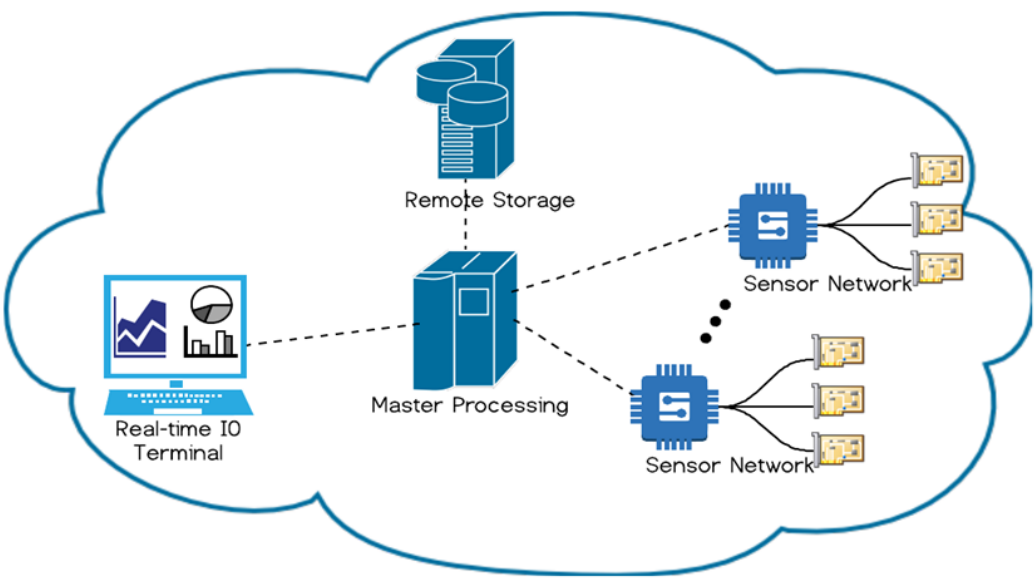

Figure 3: A CPS involves distributed intelligence in the cloud and end components

3.2 Proposed Framework

Several frameworks have been proposed in recent research to handle IoT systems for different purposes and applications. Mukherjee et al. comments on the high number connected IoT devices, and how they can outnumber the devices connected to the internet very quickly [92]. They propose a connectivity framework so that smart devices can be utilized in IoT data processing. Several surveys show different proposed frameworks in specific applications such as commercial [93], security [94], and cloud-based [95] IoT systems.

In this chapter, the proposed CPS framework is explained, which enables flexibility and adaptability by adding, removing, modifying and updating different modules within the CPS without affecting the remaining modules of the system. There are three main components that describe the framework. The first component is the physical part, which is mainly a sensor network that reads data from the environment through physical sensors and convert the data into digital signals that can be processed by a computer. The physical system can be a wireless sensor network (WSN), a body sensor network (BSN), a personal area network (PAN), or any number of sensors and actuators that produce valuable data and require action and outputs within the CPS.

The second component of the framework is the software part, and it is where most of the processing occurs. The software component is the “cyber” part of the CPS and is responsible for controlling sensory and UI inputs, handling data flow, and producing actuator and modeling/visualization data. The cyber component is also responsible for defining the set of guidelines to communicate with the data storage server where sensors historical data are stored for retrieval when needed. Since most of the processing and data handling in the proposed CPS framework occurs within the cyber component, it is mentioned hereafter as the *Data Handler*.

The third and final component is communication, which is considered as a vital and indispensable part of any CPS. The communication component is a cross-functional component and exists in all parts of the system. It is responsible for transmitting and receiving data between the cyber and the physical systems, as well as storing and retrieving data in the defined CPS data storage solution.

The components of the proposed framework are described in detail in the following sections.

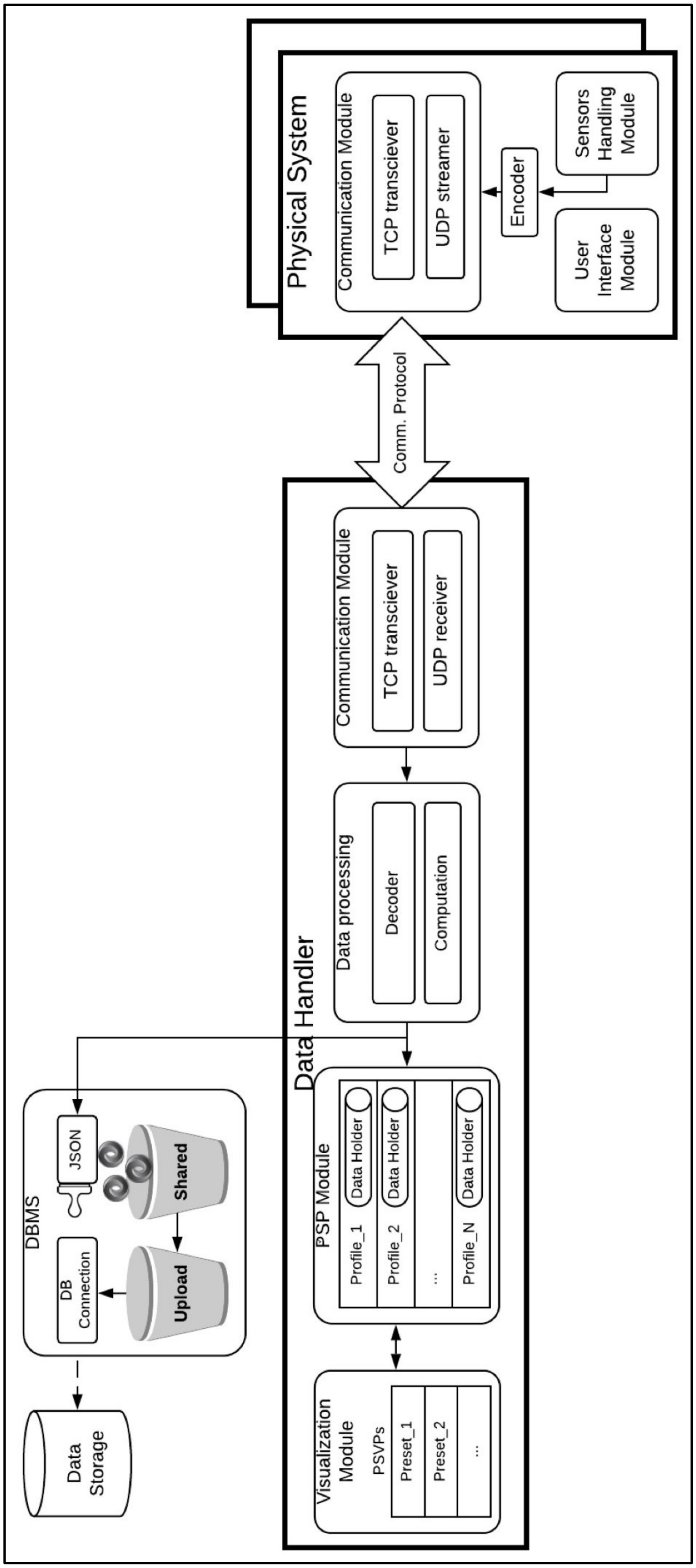

Figure 4 Proposed CPS framework

3.3 Physical System

The physical system represents an independent sensor network, which can be a WSN, BSN, PAN, etc. and is composed of at least 1 sensor and a microcontroller with communication capabilities. Most microcontrollers typically offer some mean of physical communication through which the user can program and interact with the attached sensors. Many microcontrollers, Arduino for example, provide a serial interface for that purpose. The microcontroller is then in-charge of communicating with physical sensors, encode and compress the retrieved data, and communicates it to the user interface (Figure 5).

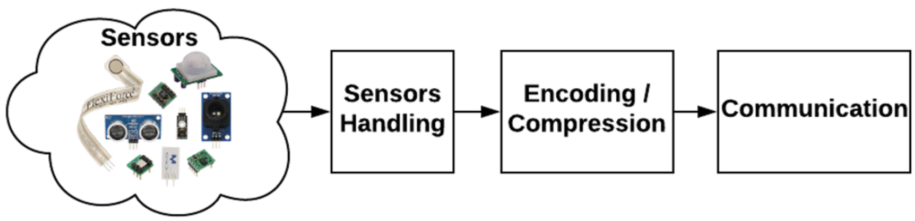

Figure 5: General block diagram of the physical system architecture

In order to prepare the sensor network for seamless integration to the proposed framework, four modules should be implemented in its main microcontroller: User Interface Module, Sensors Handling Module, Data Processing Module (Encoding), and a Communication Module. The Communication Module is described briefly here as it will be described in detail in section 3.5, which covers communication in the whole CPS.

3.3.1 User Interface Module

The User Interface (UI) module allows for troubleshooting and debugging of sensor operations in the physical system. It also provides a means of control from an external source (i.e. cyber part). The UI module should employ a set of instructions that allow for retrieving the system's status and sensory data, as well as configuring connection parameters that allow the main microcontroller to be added to a larger system as a slave in a master-slave architecture.

3.3.2 Sensors Handling Module

There are two general types of sensors: analog and digital. Analog sensors translate the sensed environment into a voltage range, where the minimum value is represented by producing zero volts, and the maximum sensed value is represented by forwarding the applied voltage to that sensor. In order to be able to process the analog sensor data in computer systems (including microcontrollers), an Analog to Digital Converter (ADC) is required. ADCs convert the voltage read by the sensor in the analog form into a digital signal that can be recognized and processed by the processor. There are several techniques to convert sensors' analog signals into digital signals [96]. However, most microcontrollers that read analog signals already take care of this conversion. Some digital sensors, on the other hand, have a built-in ADC that provides digital signals ready for processing.

Another way of categorizing sensors is by the type of interface they use to communicate with other electronics. There are several synchronous and asynchronous serial communication protocols. Some sensors simply produce data directly using Pulse Width Modulation (PWM), which can be recognized directly by almost all microcontrollers with digital interfaces. However, some sensors have their own controller that communicates through serial instructions, and many sensors use their own communication protocol such as UART, I2C, SPI, etc.

Each physical system in the CPS contains a number of physical sensors that connect to the user to collect physiological and/or biological data. These sensors can be of various types, sizes, and communication protocols. The sensors handling module is responsible for communicating and collecting the sensory data from all of these sensors, and converting them into digital signals to be stored and to be ready for compression and encoding.

3.3.3 Encoder Module

Reading sensors information and converting that information into digital values for storing and processing can take more storage space than actually required. Many CPS implementations do not discuss compression as part of their design, which leads to the assumption that each sensor value is stored in its own memory location. This may not be considered an issue. However, memory, processing power, and communication delays can accumulate and become troublesome, especially in systems with thousands, or tens of thousands of sensors.

Several compression techniques for cyber-physical systems and wireless sensor networks have been discussed in the literature [97]. However, in this section, a compression procedure is proposed providing a manual algorithm that can be used in embedded microcontroller systems controlling numerous sensors as part of the proposed CPS framework. The goal of this algorithm is to save memory space, use less processing and computation in the physical system, and to optimize bandwidth by transmitting smaller data over time. This compression algorithm trims each sensor reading, which results in compressing the overall streamed data, and creates a smoother and a more adaptable system performance. As configurability is one of the main objectives of the proposed framework, compression can be automated without affecting the overall CPS. It is worth noting that the implementation of the cyber part's decoder module must comply with the encoding and compression algorithms implemented here.

The proposed compression procedure starts by categorizing sensors based on the maximum memory space they need to occupy, trimming unused bits, and combining multiple readings to occupy the same bytes as needed. For example, if a sensor value ranges between 0-99, the maximum memory it needs to occupy is 7 bits. Integer values take between 2 and 8 bytes (depending on the compiler), which means that between 56% and 90% of that memory location is always wasted using integers. Using bytes would make much more sense since a byte occupies 8 bits. However, there is still one wasted bit in each transmitted byte in this example (12.5% waste). This waste of bits affects the communication and the performance of the overall CPS when thousands of sensors are involved. Therefore, each sensor in this category (the 7-bit category) should only occupy 7 bits. All sensors from all categories are then concatenated and sent as one packet.

At the start of sensory data streaming, a Packet Description Table (PDT) must be communicated to the Data Handler. This table describes the number of sensors in each category, and the order of sensors in the compressed packet. This table will then be used to decode the packets for further processing in the cyber part of the CPS (section 3.4.3).

Example:

Suppose we have 5 pressure sensors, 7 on-off switches, and an accelerometer that produces a floating point value between -250.00 and +250.00. We know from the sensor configuration that the reading range for the pressure sensors is between 0-100, which means that only 7 bits are required to store any value in that range. For the on-off switches, we will need 1 bit each. For the floating point values, we can use 8 bits for the whole number part, 7 bits for the decimal fraction, and 1 bit for the sign. Assuming C-language is used in our implementation, the total number of bits required to transmit the whole data load within a packet (excluding UDP headers) is going to be as follows:

Before compression	(5*8 bits)	+(7*8 bits)	+(1*32 bits)	= 128 bits
After compression	(5*7 bits)	+(7*1 bit)	+(1*16 bits)	= 58 bits

This shows that compressing sensor values by trimming excess bits result in a reduction in size of approximately 55%. The accompanying packet description table (PDT) should contain the following information:

Bit #	Sensor	Bit #	Sensor
0	Pressure sensor 1	38	On-off 4
7	Pressure sensor 2	39	On-off 5
14	Pressure sensor 3	40	On-off 6
21	Pressure sensor 4	41	On-off 7
28	Pressure sensor 5	42	Accelerometer (whole part)
35	On-off 1	50	Accelerometer (decimal part)
36	On-off 2	57	Accelerometer (sign)
37	On-off 3	58	N/A (packet termination)

Packets are streamed over UDP, and each transmitted packet will include a header that contains source and destination information (IP and port). This information will be used by the receiving end to identify the exact source of each sensor value in a CPS of thousands of sensors. Each sensor is also assigned a unique ID based on its representing information in the PDT. This assists the decompression/decryption in the Data Processing module in the cyber part to identify the exact sensor in the overall CPS.

Encryption and Cybersecurity

As MCPS can carry sensitive and private information within its network, the issue of cybersecurity and encryption arises. Kocabas et al. [27] state that healthcare monitoring is growing rapidly, and recording physiological signals from users can be done for several purposes. More devices are being developed every year and many solutions are being employed for communicating, presenting, and storing this private information. Thus, these solution must also take into account encryption schemes to protect and secure private information. Kocabas mentions that conventional encryption schemes, such as Advanced Encryption Standard (AES) and Elliptic Curve Cryptography (ECC), are widely used in medical systems due to their low resource requirements, which makes them suitable for distributed processing.

3.3.4 Physical System Communication (PSC) Module

A well-defined communication protocol must be implemented in both the Physical and the Cyber Systems in the CPS to facilitate seamless interaction between the system's nodes. The PSC module is in charge of working with the Cyber System's communication module to establish a secure connection, and to transmit encoded data produced by the data processing module. Details about the communication protocol are provided in section 3.5.

3.4 Data Handler - Cyber System

The Data Handler is the cyber part of the proposed CPS framework. It acts as a master in the master-slave architecture and controls all slaves represented by the network's physical systems described in section 3.3. The Data Handler handles several data streams coming from the attached sensor networks and is responsible for decompressing, decrypting, visualizing, and storing sensory data from all these sensor networks. It is able to do this by including six modules: UI Module, Physical System Profiler Module, Data Processing Module, Visualization Module, Data Storage Communicator, and the Communication Module (described in section 3.5).

3.4.1 User Interface

In order to allow user interaction with the CPS, a minimum instruction set must be provided to request information and control the Data Handler software. This concept has been implemented previously in similar applications to control hardware using a unified command set. A good example is the Hayes command set (also called AT command set, and Hayes AT commands) [98], [99]. Hayes AT commands are basically short text commands used to control and configure a wide range of communication modems. This command set was developed in the early 1980s and is still being used and updated regularly to control different hardware and electronics. For the purpose of controlling the proposed CPS framework, a similar-in-concept command set is developed.

3.4.2 Physical System Profiler

The Physical System Profiler is a management module in-charge of monitoring all physical systems in the CPS. It consists of two main parts: the Physical System Profiles (PSP) and the Physical System Data Holder (PSDH).

Physical System Profiles

The physical system profiles module is a static memory module that keeps track of the physical systems attached to the CPS, and keeps a record of detailed specifications of each of these physical systems so it can be accessed by other modules in the CPS. In other words, each PSP represents a digital twin of the physical system and includes descriptive information about it. This descriptive information includes details about the microcontroller in the physical system, its connection details, network status, number of sensors, types of sensors, other sensor descriptive information, etc.

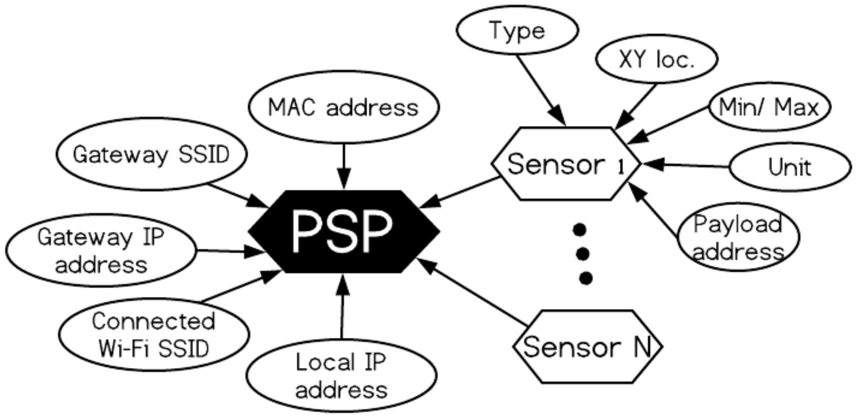

Figure 6: Example of a physical system description file (PSP)

```xml
<PSP>
    <uniqueID>
        <!-- automatically generated using MAC address -->
    </uniqueID>
    <connection>
        <MAC>1A:B2:3C:D4:76:9E</MAC>
        <gatewaySSID>ESP-769E</gatewaySSID>
        <gatewayIPv>4</gatewayIPv>
        <gatewayIP>192.168.4.1</gatewayIP>
        <wifiSSID>uOttawa</wifiSSID>
        <localIPv>4</localIPv>
        <localIP>192.168.0.105</localIP>
    <connection>
    <sensorList>
        <sensor>
            <type>FSR</type>
            <manufacturer>Interlink Electronics</manufacturer>
            <model>FSR402</model>
            <ID>FSR_00</ID>
            <coordinateX>2</coordinateX>
            <coordinateY>4</coordinateY>
            <minValue>0</minValue>
            <maxValue>100</maxValue>
            <unit>Newton</unit>
            <payloadStartAddr>0</payloadStartAddr>
            <payloadEndAddr>6</payloadEndAddr>
        </sensor>
        <!-- More FSR sensors... -->
        <sensor>
            <type>IMU</type>
            <manufacturer>Sparkfun electronics</manufacturer>
            <model>MPU-9250</model>
            <ID>IMU_00</ID>
            <accelerometer>
                <fullScale>16</fullScale>
                <x>
                    <payloadStartAddr>84</payloadStartAddr>
                    <payloadEndAddr>97</payloadEndAddr>
                </x>
                <y>
                    <payloadStartAddr>98</payloadStartAddr>
                    <payloadEndAddr>111</payloadEndAddr>
                </y>
                <z>
                    <payloadStartAddr>112</payloadStartAddr>
                    <payloadEndAddr>125</payloadEndAddr>
                </z>
            </accelerometer>
            <gyroscope>
                <!--Similar description to accelerometer... -->
            </gyroscope>
        </sensor>
    </sensorList>
</PSP>
```

Figure 7: Excerpt of a physical system description file (Physical System Profile - PSP)

The physical system description is communicated using SensorML (Sensor Model Language) which is a standard by the Open Geospatial Consortium (OGC) [100] for describing sensors and measurement processes. The OGC states that "the primary focus of the SensorML is to provide a robust and semantically-tied means of defining processes and processing components associated with the measurement and post-measurement transformation of observations. This includes sensors and actuators as well as computational processes applied pre- and post-measurement." [101]. In this work, the PSP is described using SensorML, as shown in Figure 6 and Figure 7, and the created description file is stored and updated in the PSP module in the Data Handler.

Physical System Data Holder

The physical system data holder (PSDH) acts as a buffer that is tied to the physical system profile, and is used for live data applications such as real-time processing and visualization. Each physical system (multi-sensory device) is assigned a specific dynamic memory location that holds sensory data for that device. This memory location acts similar to a computer's RAM, and is not used for storage. For each sample of data received by a physical system in the Data Handler, the PSDH is updated to only contain the most recent received sensory data from the associated physical system. This is useful for calculating real-time parameters based on the received sensory information, and for real-time visualization. In addition, a copy of each received sample is sent to the Data Storage Communicator where it can be processed for cloud storage.

3.4.3 Data Processing Module

The Data Processing Module on the Data Handler side mirrors the operation of its peer on the physical system side. Every TCP packet is assumed to be an instruction, or a response to a previously sent instruction (see Table 5). If an incorrect TCP packet is received by mistake, it is simply ignored. Similarly, each received UDP packet contains sensory information from the attached physical systems. The UDP header contains an IP address and a port number, which can be matched with a PSP to identify the source physical system. The data load will contain sensor values that conforms to the PDT stored in the PSP for that specific physical system. These sensor values are then stored in their dedicated memory locations in the PSDH.

In addition, the Data Processing Module forwards a copy of each received packet of sensory values to the Data Storage Communication Module which is in charge of keeping a record of all received information. The Data Storage Communicator is described in detail in section 3.4.5.

Real-time Parameters Calculations

In addition to all the tasks the Data Handler performs in real-time, it also performs different calculations that are needed to automate the process and make the CPS an intelligent system. The modular design of this framework takes into consideration the real-time processing of application-specific parameters. The Data Processing module in Figure 4 includes a “Computation” sub-module which takes care of parameters calculations based on the retrieved sensory information. It prepares these calculated parameters for storing in the configured database, as well as sending a copy to the PSDH for real-time outputs.

The focus of this thesis is on gait monitoring as a part of physical activity monitoring. All the models discussed in this research can be applied to other types of physical activity monitoring devices. However, for gait monitoring and analysis, the following parameters are calculated in the Data Processing Module, and prepared for real-time delivery and visualization as will be discussed in the Visualization Module.

Shoe/foot side detection: The designed hardware (SmartInsole) is identical for all sizes in terms of the placement of sensors. The same flexible PCB can be used for the right foot and the left foot. However, the Data Handler is able to identify which foot it is getting data from by performing simple verification at system startup. Once the system is started, the Data Handler requests data from all insoles accelerometers’ Z-axis, and uses an average of the collected data to determine if the insole is used for the left or the right foot.

Number of steps detection: Estimating the number of steps is performed by each foot individually, and adding the right and left foot’s steps result on the overall steps performed by the user. This information is calculated by many fitness trackers and smartphone apps, such as FitBit, using intelligent algorithms and embedded accelerometers. However, step count using the SmartInsole is difficult to cheat since a step is only counted if the user performs an actual step on the insole. In our implementation, the average pressure is measured and continuously updated to control a threshold. A step is detected each time the real-time average pressure exceeds the dynamic threshold. The threshold value is updated continuously throughout the user activity to account for the user’s changing pace. The step count for each insole is calculated as follows:

$$S(x) = \begin{cases} S(x) + 1, & A(x) - \alpha \geq 0 \\ S(x), & A(x) - \alpha < 0 \end{cases} \quad (1)$$

where $S(x)$ is the number of steps for insole x, $A(x)$ is the real-time average FSR values for insole x, and α is the dynamic threshold.

Cadence: Cadence is defined as the number of steps performed per minute. It is an important statistic in gait analysis for healthcare and for sports analysis. Cadence is not to be confused with gait speed, which is still an ongoing research and not easy to calculate in wireless systems due to the difficulty of estimating the step length (distance between feet). To calculate cadence, the system starts by extracting heel-strike (HS) and toe-off (TO) timestamps. These timestamps correlate with the minimum sum of forces imposed on the FSR sensors. The maximum sum of force imposed on the FSR sensors is what we consider the foot-flat (FF) instance. The sequence of these events for normal walking occurs in the following order: HS-FF-TO. By extracting the timestamps for HS and TO, we can calculate the number of steps and the duration between each step. Using this information, we calculate cadence as the number of steps occurring per minute.

Center of Pressure calculation: The Center of Pressure (*CoP*) is the point in between the object and the supporting surface where the sum of ground force vectors is applied. It is different from the Center of Mass (*CoM*) which is a specific point in the space any object occupies (linear, 2D, or 3D) where the sum of relative weight vectors sums to zero [102]. In other words, CoM is located at the center of the distribution of mass for that object. *CoM* and *CoP* are important variables in assessing energy consumption and walking stability, especially in health-related applications, such as in sports, and orthopedic treatment.

To calculate linear CoP (i.e. in one dimension), the following equation is applied:

$$CoP = \frac{\sum(F_{\mathrm{n}} \cdot x_{\mathrm{n}})}{\sum F_{\mathrm{n}}} \quad (2)$$

where F_{n} and x_{n} are the applied force and location of object n respectively.

This equation gives the location of the *CoP* relative to an origin point, where the distance of each applied force from it is noted by x_{n}. However, to calculate the *CoP* of a human subject on the ground, this equation cannot be applied directly since the *CoP* required is in 2D and not linear. Therefore, we need to follow two steps: we start by calculating the 2D CoP under each foot, and then calculating the linear *CoP*, using equation (2), between the *CoP* of the left and the right foot.

This results in finding the location of the total CoP imposed by the subject on the 2D ground (see Figure 8).

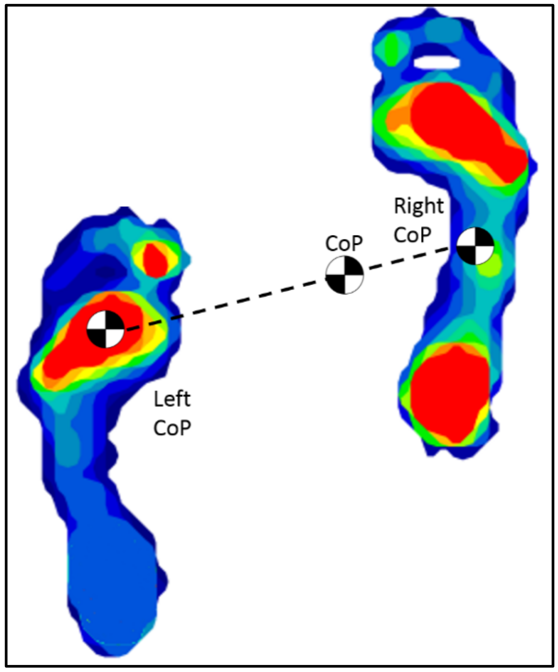

Figure 8: Proposed calculation of the CoP in 2D using the SmartInsoles

To calculate the *CoP* for each insole separately in 2D, equation (2) is used for the x- and the y-coordinates separately. This results in having two values; one for the x-coordinate, and the other for the y-coordinate. These are the XY-coordinates of the *CoP* for this foot.

Using the constant locations of FSR sensors defined in the PCB design, we can approximate the center of pressure for each insole using the below equations, which were adapted from equation (2):

$$\mathrm{CoP_x} = \frac{\sum(F_\mathrm{n} \cdot x_\mathrm{n})}{\sum F_\mathrm{n}} \tag{3}$$

$$\mathrm{CoP_y} = \frac{\sum(F_\mathrm{n} \cdot y_\mathrm{n})}{\sum F_\mathrm{n}} \tag{4}$$

After acquiring the XY-coordinate for each foot's *CoP*, we can use the linear *CoP* calculation method (i.e. equation (2)) to find the total *CoP* between the two feet. Since calculating the distance between two feet is still an ongoing research topic, we can calculate the center of pressure as a reference-based location between the two feet by referring to one foot as location zero, and the other as location 100. This way we can determine the distance ratio between the two feet as a linear relative location between 0-100. The total *CoP* can be calculated using the following equation:

$$\mathrm{CoP} = \frac{0 \cdot \mathrm{F_L} + 100 \cdot \mathrm{F_R}}{\mathrm{F_L} + \mathrm{F_R}} \tag{5}$$

where F_L and F_R are the average sensor values for the left and right insole's FSR sensors respectively. The result CoP can then be considered as the linear location, as a ratio, between F_L and F_R.

3.4.4 Visualization Module

The Visualization Module is in charge of visualizing sensor data from the physical system. Using the details in the physical system's PSP, the Visualization Module automatically recognizes what types of sensors need to be visualized, and uses pre-coded visualization presets to provide meaningful visuals to the user. The physical system visualization presets (PSVP) are presets that include information about visualizing a specific type of sensor, and are not dedicated to specific PSPs. The PSVP and the PSP work closely together such that each new type of sensors added in the PSP, should be accompanied by a PSVP. A PSVP can also describe and include presets of a complete physical system such that it represents a device, and not just a single sensor.

3.4.5 Data Storage Communicator

The Data Storage Communicator is in charge of data from the time it is processed by the Data Processing Module, until it is stored in the dedicated storage server. Once a data packet containing sensory values is received, the Data Storage Communicator creates a JSON object [103] with attributes that represent sensor IDs. The JSON object will also contain attributes to represent the timestamp, the unique physical system ID, and any descriptive information required by the CPS to identify or describe the physical system. The Data Handler's system clock is used to timestamp data, which ensures that all received data are synchronized using one common clock.

When JSON objects are prepared and ready to be stored, they are collected in a “shared bucket” that can hold a specified number of objects from all connected physical systems. Once a certain number of objects are collected in this bucket, the objects are then dumped into an “upload bucket” that will be uploaded as a bulk into the storage database. The reason for creating a shared bucket is to enable storing all JSON objects in one place without worrying about their source, or the storage server they are destined to reach. In addition, to avoid duplication of stored data, the shared bucket is emptied once its contents are copied into the upload bucket. Having a separate bucket for uploading eliminates errors that arise from concurrent modifications of the same data (i.e. storing and deleting of the same memory location). Furthermore, the upload bucket allows for bulk upload of smaller data, which reduces the load on the storage database and helps maintaining a stable database connection for bulks of data at a time.

The database connection sub-module is simply a code fragment that is responsible for storing database connection information, regardless of the type of data that the CPS needs to store. The database connector maintains a stable connection to the database, and sends all the data stored in the upload bucket to it. Once upload is confirmed, the upload bucket is then emptied to allow for the next batch coming from the shared bucket.

Storing historical data is necessary for retrieving important information and for enabling the sensory data to be modelled and visualized to facilitate the operation of users of the CPS. JSON is used because it is a descriptive object notation that uses less text in comparison to XML. JSON is easier to read, lightweight, and creates files with smaller sizes [103]. This reflects on communication performance where JSON requires less bandwidth, which made it popular for Internet of Things applications. This means faster processing, and faster data transmission, while serving the same purpose. Many cloud database servers use JSON for these reasons, which means more data (big data) can be stored for further processing.

3.4.6 Cyber System Communication (CSC) Module

The Cyber System, as the master in a master-slave architecture, controls the connected Physical Systems in the CPS. Depending on the type of networking used within the CPS (i.e. Wi-Fi, Bluetooth, Serial, etc.), the communication protocol is defined and used by the CSC and the PSC to set-up connections and to transmit and receive data between the system nodes. The CSC module is also designed to identify the source Physical System and use the received data according to its intended use in the application. More details about the communication protocol is provided in section 3.5.

3.5 Communication

Different systems are currently available for similar applications. Some use wired communication to record sensory data [56], [82], [83] in order to provide a rigid mode of data transmission, especially for heavy data such as 3D and haptic data. The drawback for wired devices is that they either limit the free physical movement area, or require bulky and expensive setups [56]. The second category includes devices that record data wirelessly using Bluetooth [48], [54], [68], [84] or ZigBee [39]. Bluetooth and ZigBee, although consume much less lower power compared to Wi-Fi, have low range capabilities [84] and, thus, limit the device's free movement area. In addition, these technologies limit the number of connections an application can use since Bluetooth and ZigBee are typically one-to-one connections. However, attempts have been made to time-share multiple Bluetooth connections while compromising stream bandwidth [86]. Therefore, we proposed a system that uses Wi-Fi, which potentially solves all of the aforementioned drawbacks for this kind of health-related applications.

One objective of this work is to provide a complete framework that uses conventional Wi-Fi networks to setup, monitor, and record detailed sensory data for multiple wearable devices simultaneously. Using Wi-Fi also allows us to record in a relatively large field (such as a soccer or a hockey field).

The communication in the proposed CPS framework switches between two modes: TCP for critical communication, and UDP for loss-tolerant communication. The communication modules in the Data Handler and the physical system are responsible for the switching between these modes of communication in the CPS, and the mode of communication is determined by type of data being transmitted between the endpoints. An implementation of the two sets of commands is described in section 4.4, which facilitates seamless communication between the Data Handler and the physical systems. Data Handler commands utilize the physical system's command set to interact and control the microcontroller in the CPS. These instructions and their responses are transmitted on top of TCP protocol using HTTP. When the system is streaming sensory information, the communication module switches to UDP mode until a "STOP_STREAM" request packet is received by the physical system.

There are a couple of advantages gained from this communication mode switch. One objective of this proposed framework is the ability to communicate and control the physical system over Wi-Fi to allow for a relatively wide-range communication. Using UDP for all types of communications could be risky due to the loss of critical commands being lost if no acknowledgements are being received on either side of the system. On the other hand, using TCP for all communications slows down the data stream dramatically and will result in a reduced transmission rate for sensory data. Therefore, a combination of TCP and UDP with switching ability is a more suitable solution for CPS applications using Wi-Fi as the medium of communication.

Communication in the proposed framework operates in a mechanism that can be summarized in the stacked layers diagram as shown in Figure 9. Sensors in each Physical System collect sensory information, and each Physical System is responsible for its own data collection from these sensors via the sensors handling layer. The acquired sensory data is handed over to the encoding layer where it can be wrapped and compressed to be ready for communication. After that, the communication layer prepares the packets by adding headers to the payload. Each header contains a unique identifier that enables the transmitted packet to be traced back to its source Physical System. After that, the communication layer in the Cyber System (which is represented by the CSC module) receives the encoded packet from a Physical System and forwards it to the decoding layer where it can be unwrapped to separate the unique identifier and the payload segments. The Physical System Profiler can then use the unique identifier to recognize the source system and its PSP. After that, it uses this information to assign each sensor value in the payload to its allocated value in the software. The computation layer can then use the software variables to calculate any parameters required by the application by referring to these sensor values. All of this prepared data can then be stored in a buffer for real-time visualization and output, and for preparation for cloud storage.

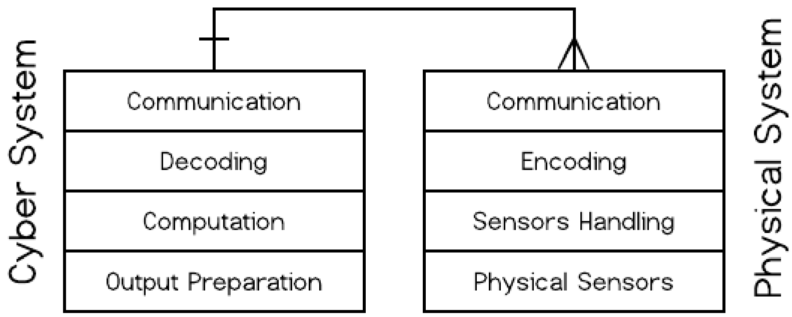

Figure 9: Stacked operation layers showing the one-to-many relationship between the Cyber and the Physical systems.

3.5.1 Communication Protocol

A protocol is a set of rules that dictate how communication is performed between two or more endpoints. In this section, the proposed communication protocol for configurable MCPS builds on top of existing protocols (TCP and UDP) to transmit information between the CPS endpoints. Depending on the type of transmitted information, the proposed communication protocol follows specific steps to switch between TCP and UDP depending on the type of transmitted information. In this proposed protocol, TCP and UDP are used as the transport layer protocols since they are the most widely used around the world. However, the same concept can be applied if other transport protocols become more popular. Optimizing bandwidth is important for real-time and data hungry applications, and this can be done by switching between lossy and a lossless transport protocols depending on how critical the transmitted information is.

The physical system does not include any user interface other than the power switch and the LED indicator. Therefore, a connectivity framework is employed in the Data Handler software to allow the user to wirelessly add a new physical system to the network and to the CPS. After setting up the new physical system's connection, the Data Handler software can monitor and visualize sensor readings from all connected devices added to the CPS using a single computing device.

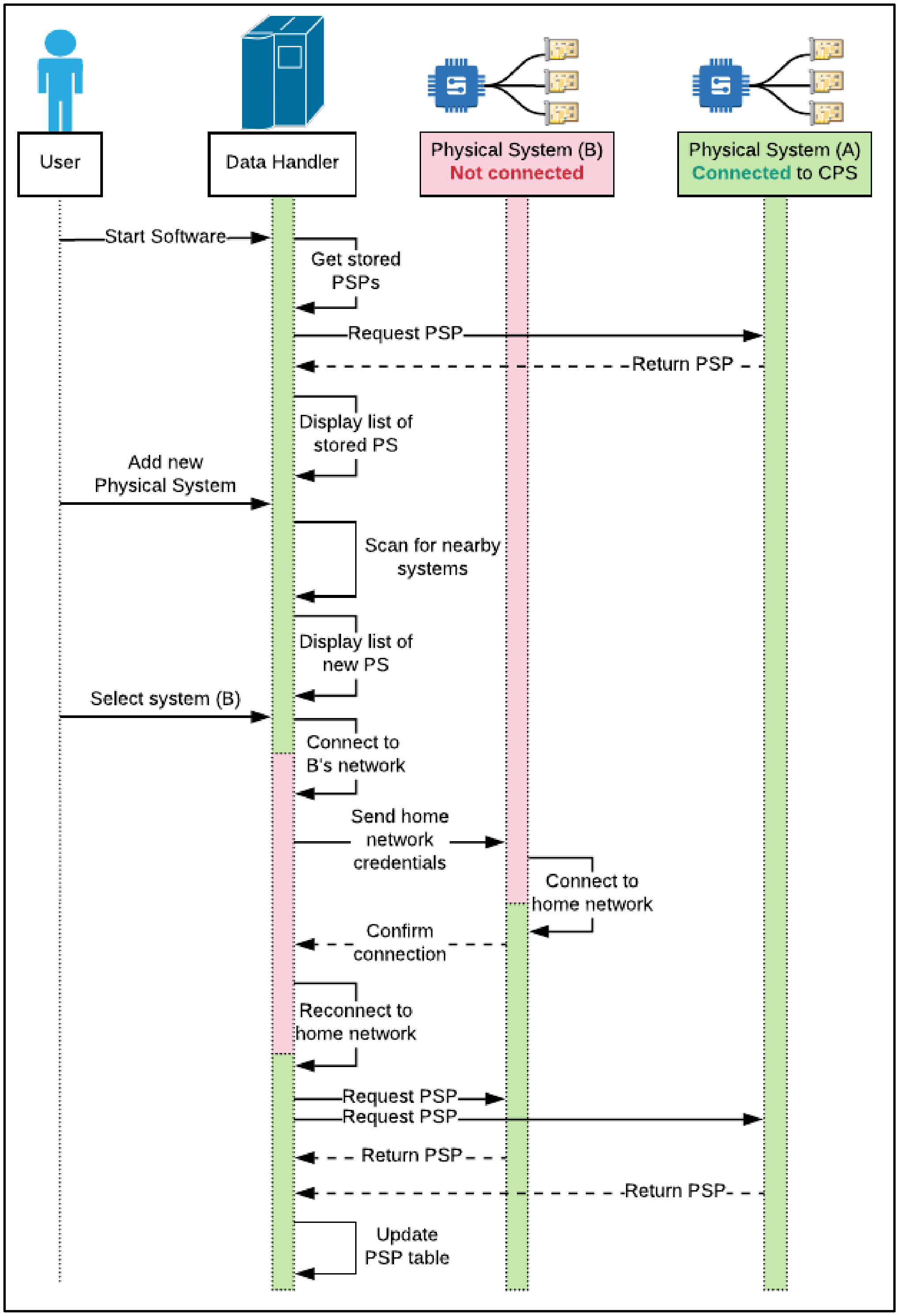

Figure 10: Sequence diagram showing how Data Handler performs adding a new Physical System to the CPS

Figure 10 shows how the Data Handler performs a number of operations to allow limited interaction and intervention from the user, which makes adding a new physical system to the CPS a seamless and easy procedure. The scenario in Figure 10 shows a CPS that contains only one physical system (A), and the user attempts to add a new physical system (B), which is an un-configured one, to the CPS. When the user starts the software, the Data Handler looks at the stored Physical Systems Profiles, and requests a quick update to each connected device's PSP. After that, it displays all connected devices to the user. In this scenario, the user requests to add a new Physical System. Since every Physical System is a data server with Wi-Fi capabilities, the Data Handler starts by searching for Wi-Fi networks, and filters and displays those that are broadcasted by a Physical System to the user. The user can, then, select a Physical System to add to the CPS. The Data Handler creates a copy of the home network's credentials (SSID and encryption key) from the current computer, where the Data Handler software is operating. After that, it connects the selected Physical System, and provides it with the copied credentials over ab encrypted channel so it can connect to the home network. These credentials are encrypted based on the physical address of the physical systems which is only visible to these two endpoints. Upon receipt of the network credentials, the Physical System attempts the connection and responds to the Data Handler whether its attempt is a success or not. The Data Handler re-connects to the home Wi-Fi and refreshes PSPs from all Physical Systems in the CPS.

The switch from TCP mode to UDP mode in this framework occurs when the cyber side sends a "start" command to all physical systems in the CPS in order to start the streaming process (Figure 11). The Data Handler starts by looking at the stored PSPs, which contains configurations of all devices in the CPS, and ensures that all devices are actually online. It then sends a TCP request to start streaming data, while providing the current machine's IP address, and a dedicated UDP port number to each Physical System. The Physical System responds with a TCP message containing the UDP port number it can be reached at during UDP mode. It then immediately switches to UDP mode and starts streaming to the Data Handler using the provided IP address and port number.

The Data Handler occasionally responds with a UDP message by sending an acknowledgement message to the Physical Devices. If a timeout occurs, the Physical System will know, and will stop streaming data. Once the user requests stopping the stream, the Data Handler sends a "STOP" signal to all Physical Systems through their previously provided UDP ports. However, in the case of a dying Physical Device (Physical System), the Data Handler will timeout at the assigned UDP port and will recognize immediately that this health device is no longer streaming data. In this case, an alarm can be triggered to notify the UI about this.

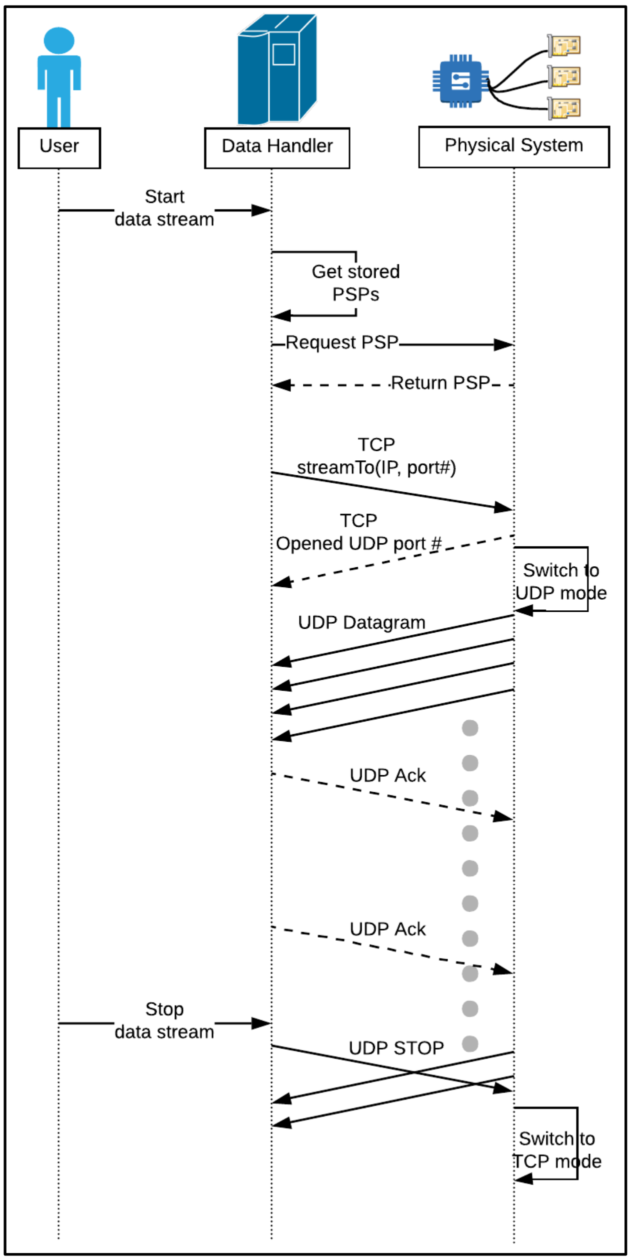

Figure 11: Sequence diagram showing the switch between TCP and UDP modes within the CPS

3.5.2 CPS Operation Scenario

In this section, a scenario will be presented to show how the proposed CPS described above would work in a real-life situation. We start with the Data Handler software and the main User Interface which the user will use to control the CPS.

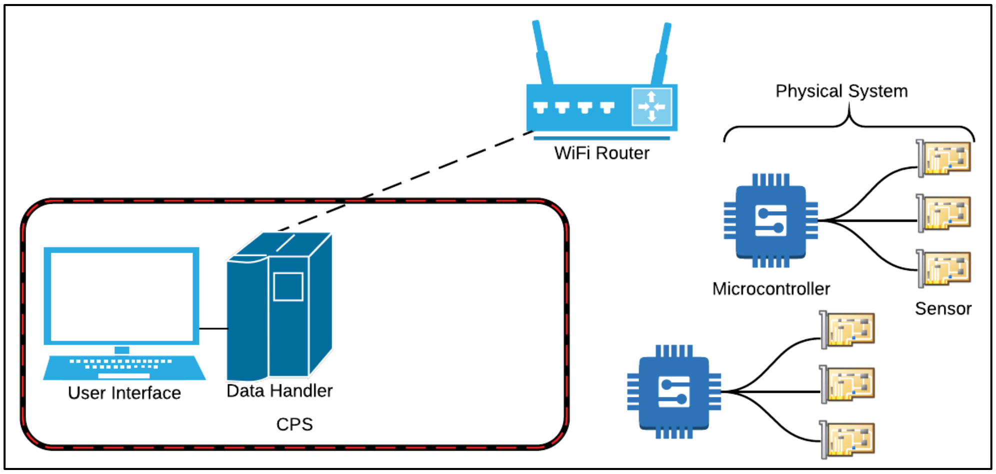

Figure 12: Data Handler and Physical Systems not yet connected

1. The user starts by using the *setup* DH command, which will search the surroundings for physical systems by sending a special request, and awaits an acknowledgement. When the system identifies physical systems in the surroundings, it will list their unique IDs in the UI to allow the user to select which physical system should be added to the CPS. The user then inputs the Wi-Fi credentials to be transmitted to the selected physical systems, which, in turn, will use to join the CPS Wi-Fi. The Wi-Fi credentials are transmitted to the physical system through an encrypted channel which uses the physical system's ID as the encryption key. This way, the transmitted credentials are protected from eavesdropping attacks and can only be decrypted by the Data Handler and the receiving physical system. By the end of this operation, the microcontrollers in the physical systems will be in the same network as the Data Handler, which will allow it to interact directly with them. However, these microcontrollers (and their physical systems) are not yet in the CPS.

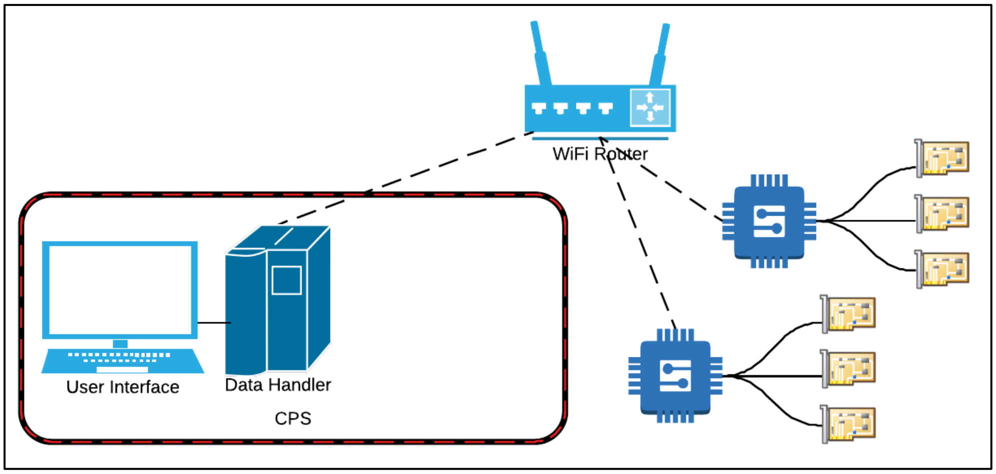

Figure 13: Physical Systems joined the Wi-Fi network

2. The new physical systems are then added to the CPS by using the *addAll* command, which searches the home Wi-Fi for physical systems (i.e. microcontrollers in the same Wi-Fi). Physical systems are identified by replying with specific responses to instructions sent by the Data Handler over the Wi-Fi network.

   Note: in our implementation of this system, the DH command *add* can be used to add one physical system at a time by using its unique ID or its IP address. The *setup* and *addAll* commands are separated to allow more freedom in configurability. This allows users to put some physical systems on hold by adding them to the Wi-Fi, but not to the CPS. However, these commands are conjoined to offer enhanced automation.

3. The two physical systems that were added to the Wi-Fi send their configuration files (PSP) to the Data Handler to join the CPS. The Data Handler can then create a PSP and PSDH location to hold the detailed specifications describing each of these added physical systems. The PSVP will automatically assign visualization presets to each physical system based on the sensors descriptions in the PSP. The user now can request the DH command *start*, which will send a request to start streaming from both physical systems. The cyber and the physical parts of the CPS will switch into the UDP transmission mode, to start live data stream and visualization.

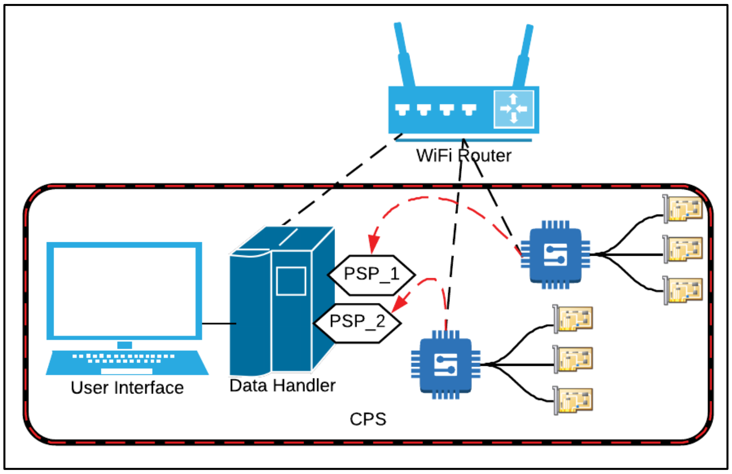

Figure 14: Physical Systems added to the CPS, and PSPs created in the Data Handler

4. Received streams in the Data Handler are filtered based on the IP address in the UDP header. The IP address and port number help the Data Handler match the stream with its corresponding physical system (using PSP files) to identify the source of the stream. Each packet is then decrypted and stored in its corresponding PSDH. The Visualization Module uses the buffered data in each PSDH to perform real-time visualization of sensory data. In addition, the Data Storage Module in the cyber system takes a copy of the data in the PSDH to create timestamped JSON objects, and places this sample in the shared bucket in preparation for storage.

5. The Data Storage Module monitors the shared bucket until a threshold is reached based on the number of JSON objects stored in it. Once this threshold is reached, this module dumps all JSON objects into the upload bucket. Finally, the DB Connection monitor starts a connection with the database and uploads all JSON objects in the upload bucket to the database. The upload bucket is then cleared and becomes ready for the next batch.

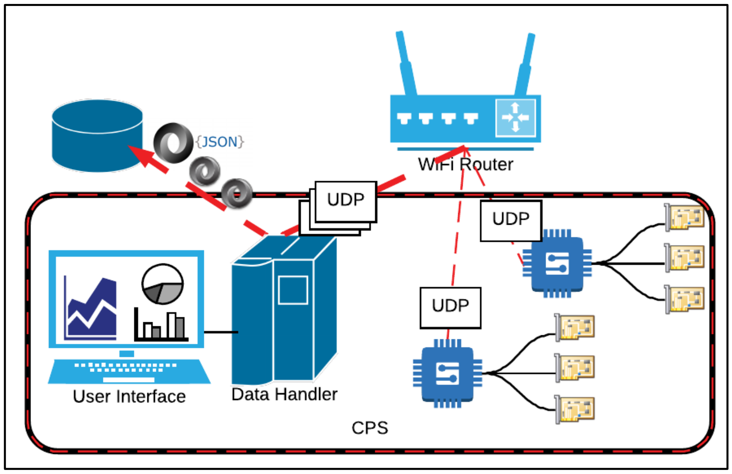

Figure 15: CPS with complete setup showing UDP sensory data stream, visualization on the UI, and data storage operations.

3.6 Chapter Summary

Chapter 3 introduced the proposed CPS framework by, first, explaining the general concept of input-output systems and distributed intelligence in connected systems. Then, a general description of the proposed cyber-physical system framework is provided, followed by details of the physical part and the cyber part. The different modules of the physical system are then laid out, including the encoder module where the proposed compression algorithm is explained. After that, modules of the cyber system are described, and some of the real-time parameter calculation methods are explained. Finally, the proposed communication protocol is explained in detail, followed by an operation scenario that explains how auto-configurability of the proposed system would work in a real-life situation.

Chapter 4 covers the implementation details of the proposed framework showing circuit designs and implementation details of the multi-sensory health device, and the software design of the cyber system.

Chapter 4 Wi-Fi-Based CPS for Gait Parameters Measurement and Analysis

Recording detailed foot kinetics and pressure points data can be beneficial in post-routine analysis such as sports-related performance enhancement or medical and rehabilitation diagnostics. Most of the current systems are used mainly for health and wellbeing applications, and research is still working on finding an optimum solution that can utilize gait data for fall prediction, detection, and avoidance [64], [104]. Solutions to this problem are especially imperative for people who have high risk of serious injuries as a result of falling or tripping.

As a proof of concept to the proposed framework, a configurable CPS model for health and wellbeing activities monitoring is designed and implemented. The implemented CPS is a gait activities measurement system that delivers sensory data from multi-sensory physical devices, to the cyber system and to the data storage server over conventional IEEE802.11 Wi-Fi. The objective is to design a complete Cyber-Physical System that aims at solving the drawbacks discussed in the literature, specifically for health and wellbeing applications. The developed insoles include FSR sensors as well as an IMU, and the circuitries are printed on flexible PCBs for improved durability. The Cyber-Physical System implemented here uses Wi-Fi to setup, monitor, and record detailed and real-time sensory data for multiple wearable insoles simultaneously. Using Wi-Fi allows us to record sensory data in a relatively large space such as a soccer or a hockey field. The developed system also permits seamless adaptability to different types of physical wearable devices and body sensor networks, configurable visualization, and data storage methods.

The implemented CPS contains different components including A) physical devices represented by SmartInsoles with different embedded sensors, B) a software component, referred to as the Data Handler, which is in-charge of controlling, managing, handling, and storing incoming sensory data, and C) networking controllers responsible for data traffic between the system's nodes. These components work together to achieve the task of measuring, visualizing, and storing gait parameters for multiple insoles and consequently users.

Each physical device is a gait measurement system by itself that can provide raw data through serial communication. However, the Data Handler was designed and tested to handle data streams in real-time and provide, store, and visualize detailed sensory data from all devices on the Wi-Fi network simultaneously on a single computing device (Figure 16).

The proposed solution adds configurability to the literature by enabling the system to wirelessly configure and monitor new physical systems. In addition, this work shows potential for applications involving multiple users that need their gait to be assessed in a concurrent manner, such as coaching sports teams or group physiotherapy.

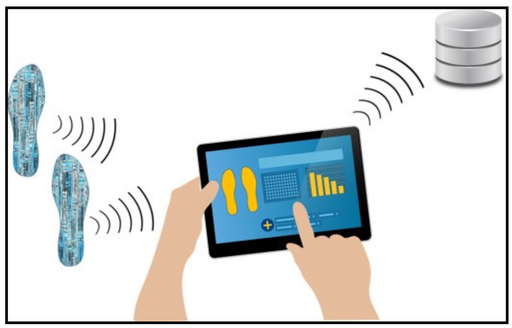

Figure 16: Multiple SmartInsoles transmit data to the User Interface for visualization while simultaneously storing data in the cloud database.

4.1 SmartInsoles

The physical system of the proposed CPS framework can be represented by the multi-sensory gait measurement instrument designed in the form of a smart insole. The hardware design and implementation of this insole is the result of a re-design of the SmartInsole gait measurement device initially started by Hafidh et al. [46] in 2013 at the MCRLab. Several versions were implemented during this progress of this research to ensure accurate measurement and durable hardware design. The design of these SmartInsoles went through multiple stages, and each stage had its own challenges. The following sub-sections provide information and description on each of the stages that were involved in the design of the SmartInsoles.

4.1.1 Circuit Design

A master's research conducted in 2012 at the University of Utah performed comprehensive analysis and several tests to validate a 12-sensor insole and compare gait data to a similar insole with 32 sensors [83]. The locations of sensors in the 12-sensor insole were derived from the 32-sensor insole, which determined which of the 32 sensors are most utilized. Results suggest that the effectiveness of 12 pressure sensors embedded in the insole are comparable to a similar insole with 32 pressure sensors. For this reason, the embedded circuit in this design includes 12 FSR sensors to measure pressure data. FSR sensors come in different shapes and sizes. In addition, some studies also use piezoelectric transducers and custom-built pressure sensors in their research. Piezoelectric transducers cannot produce a continuous force pressure reading, and can only produce pulse signals, while custom built pressure sensors are usually bulky and cannot fit in a shoe insole. For these reasons, FSR sensors are concluded to be the best fit for this application, and Interlink's force sensor "FSR-402 Short" was used in this design due to its high accuracy, large sensing area and small size. A comprehensive comparison and description of different FSR sensors can be found in [88], [105], [106].

Calculating foot orientation during gait can also add a great value to research and can produce valuable data for analysis related to health and wellbeing. For this purpose, an Inertial Measurement Unit (IMU) is added to the implementation. A 9-axix IMU breakout board by SparkFun [107] is used in the circuit design. The embedded MPU-9250 [108] chip contains an accelerometer, a gyroscope, and a magnetometer with high reading frequency, and communicates using I2C protocol. The small design of this breakout makes it a great fit for compact applications, such as the SmartInsole, since it can be embedded with minimal obstruction to the user. In addition, the high data acquisition rate makes it a suitable device for fast-paced applications such as running or playing different sports.

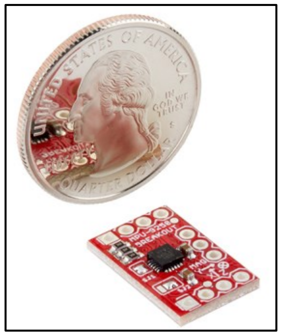

Figure 17: MPU-9250 breakout board by Sparkfun compared to a US quartar dollar

The microcontroller used in the implementation of the physical system is the ESP8266 by Espressif Systems [109]. The ESP8266 is a tiny 32-bit Wi-Fi 802.11 b/g/n enabled microcontroller with full TCP/IP stack. It has 16 GPIOs, and a CPU that can run with a frequency of 80 MHz. Many breakout boards are available from several manufacturers for the ESP8266. Each breakout has its own advantages and disadvantages including board size, available GPIOs, ease of programming and prototyping, etc. However, Adafruit's Feather HUZZAH [110] was used in this implementation because it provides access to 10 GPIO ports that can be used for I2C and SPI communication, the reset button allows easy bootloading for firmware updates, micro USB to FTDI adapter, and a built-in LiPo battery charger with a 3.3V regulator.

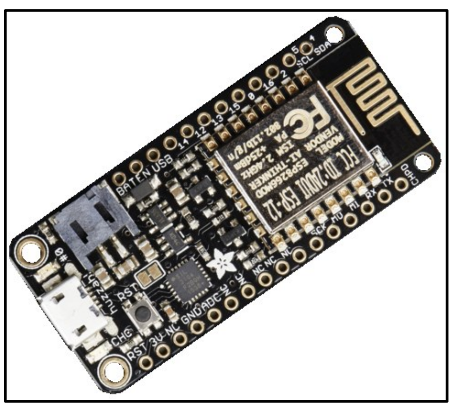

Figure 18: Adafruit's ESP8266 microcontroller breakout Feather HUZZAH

There are a few problems that required consideration and handling when using the ESP8266 microcontroller in this implementation. The microcontroller has only 1 ADC pin, and this pin can operate with a maximum of 1.0 volt. However, the microcontroller itself operates at 3.3V, which rises a problem when using the chip's power with variable resistors (such as FSR sensors) to determine analog signals coming into the ADC. The first issue is solved by using a 16-channel multiplexer at the ADC input. This added a bit of complexity to the firmware code, but was not considered a problematic setback. The second issue, however, had to be solved using hardware. In order not to destroy the ADC input or read incorrect analog signals, a voltage divider circuit was added to the ADC pin so analog signals are mapped from 3.3v to approximately 1 volt. Another problem that arose from using the ESP8266 microcontroller is the high power consumption when transmitting over Wi-Fi. The ESP8266 draws an average of around 170 mA of current when transmitting or receiving data (some tests show up to 450mA consumption), and approximately 1mA during its power save mode. Since this study's application requires at least 1 to several hours of operation, a 2000mAh 3.7V LiPo battery was used. The battery life and power consumption relationship for LiPo batteries can be calculated using (6) below [111].

$$Battery\ Life\ (hours) = \frac{Capacity\ (mAh)}{Load\ Current\ (mA)} \times 0.7 \tag{6}$$

Figure 19 shows the physical system circuit design and how all the physical components are wired together. This is the exact circuit that was used in all implementations of the SmartInsole device developed for this study. Figure 20 shows the layout of sensors and components relative to a human foot. Different foot sizes alter the locations of some of the components described in this figure. Section 4.1.2 describes how this issue is resolved in the fabrication stage of the system implementation.

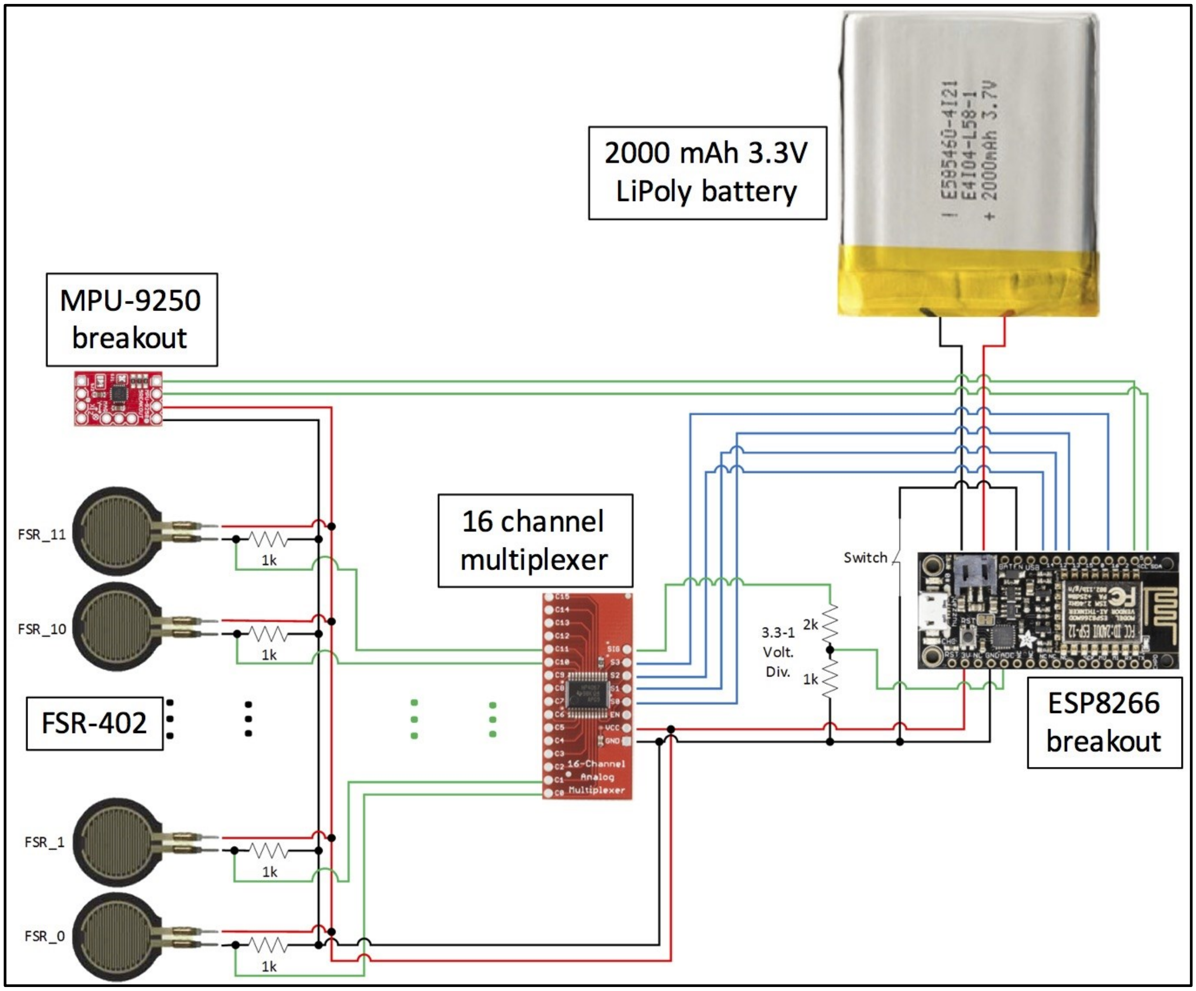

Figure 19: Physical System circuit diagram showing different components and how they are wired

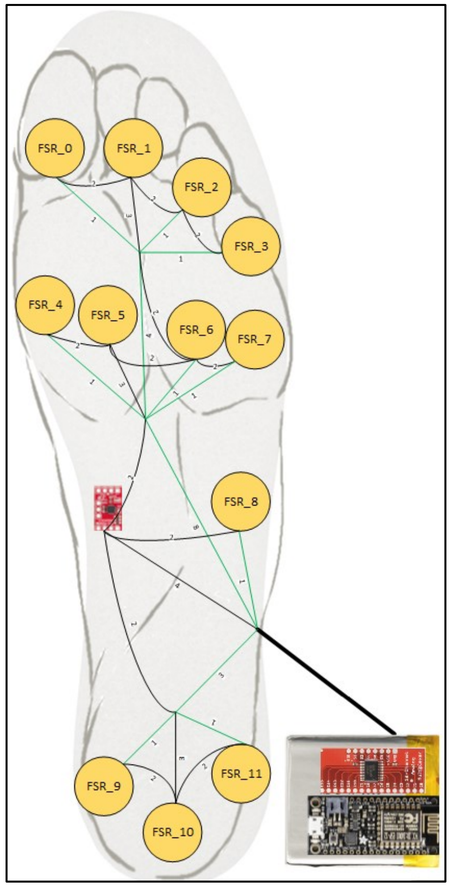

Figure 20: Physical System sensors and components layout design

4.1.2 Sensors Layout and PCB Fabrication

One of the challenges faced while designing the SmartInsoles is the difference in shoe sizes for different subjects. In order to solve this issue, SmartInsole sizes were divided into 3 groups: females of size US 6-7, males of size US 7-8, and males of size US 9-10. Several subjects from each group volunteered to have approximate frames drawn around their feet. After that, for each of these categories, gridlines were added to the drawn frame, and best sensor locations were estimated and determined. The XY coordinates of each sensor was recorded and used in the PCB design.

The twelve FSR [17] sensors are carefully placed on the insole in areas where pressure force could be detected. Multiple studies suggested that the force pressure on the medial midfoot (arch) is minimal/negligible [112]–[115], and thus are not measured in our design. The twelve FSR sensors are distributed such that they cover the remaining areas of the foot.

The IMU is also placed in the insole's medial midfoot area to utilize the space with minimal pressure, and to enable capturing of foot kinetics and orientation data. The IMU sensor is capable of capturing real time 3D acceleration, 3D rotation (via gyroscope data), and 3D orientation (via magnetometer data). For this purpose, we used MPU-9250 which is a 9-axis IMU used for applications requiring high precision such and fast target movement monitoring [116]. This IMU is capable of sampling at a frequency of 50Hz and consumes 3.7mA of power [117].

Using the above measurements and specifications, 2-layer flexible PCB boards were designed and manufactured. For each SmartInsole, FSR sensors were then connected to a multiplexer that forwards pressure data to the Wi-Fi enabled microcontroller.

All FSR sensors and the IMU are embedded in the insole, which is to be placed inside the user's shoe. However, we decided that the Wi-Fi enabled microcontroller and the rechargeable battery are to be placed outside the shoe (i.e. strapped to the ankle) in order to reduce the risk of battery overheat, and to allow for a better signal for wireless communication. Everything is wired together using different color-coded 30 AWG wire-wrapping wires for lighter and airy design.

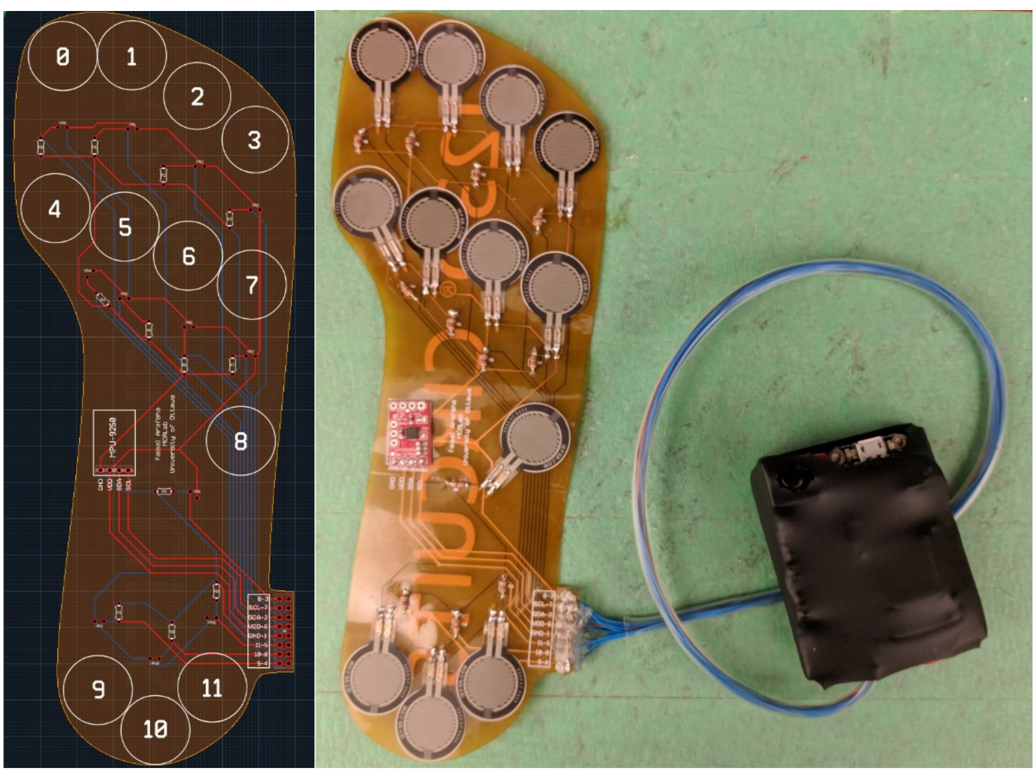

Figure 21: (left) PCB design of the male size US 9-10, (right) Picture of the final PCB with embedded FSR and IMU sensors, and the package containing the microcontroller, battery, and multiplexer.

4.1.3 Microcontroller Programming

As shown in the proposed framework in Figure 4, the physical system contains a Sensors Handling Module, a Data Processing Module, a User Interface Module, and a Communication Module (described in section 4.4).

The Sensors Handling Module is programmed in the microcontroller to communicate with the IMU, control the multiplexer, and retrieve FSR sensory data. The microcontroller communicates with the IMU using I2C protocol, and the 16-channel multiplexer, controlled using 4 select connectors, cycles through all connected FSR sensors to collect pressure data. The aim of the measurements performed by the insoles is qualitative and not quantitative. Therefore, high quality FSR sensors were used, which can measure forces between 0-100 Newton. All FSR signals are mapped to a 0-100 value representing applied force pressure and stored as 8-bit unsigned integers. On the other hand, accelerometer and gyroscope data are stored as floats.

After that, the Data Processing Module retrieves all sensory readings from the temporary memory storage, performs compression and encoding, and forwards this data to the Communication Module which packs sensory data in a UDP datagram and sends it to the Data Handler (section 4.2). FSR data and IMU data are refreshed and transmitted at 20Hz which is considered real-time while allowing some processing time at the receiving end for historical data storage and live visualization.

One LED is used to portray the current state of the microcontroller. The flashing pattern of the LED communicates to the user the current connection and processing status of insole.

Table 4: Status codes shown by LED indicator on the Insole Controller

Flashing rate	Status
Off	Power is off or battery is empty
2 quick flashes, then 1s delay	System ready
On-off with 500ms delay	Attempting connection to Wi-Fi
Solid on	System connected to Wi-Fi
Continuous flashing	System transmitting data

The microcontroller also performs calibration and detects if the insole is a right or a left insole by performing looking at the IMU's Z-axis upon system boot. Continuous negative Z-axis values concludes that the IMU is upside down, and therefore, the system applies left-side calibration values to all future IMU readings until next reboot.

All major firmware changes are stored in a separate version, and each version includes a note with the list of new changes done to the previous version. The current version of the installed firmware can be retrieved by the Data Handler or by the serial interface implemented as the UI of the Physical System.

A flowchart of the microcontroller's operation can be found in **Appendix A**.

4.2 Data Handler

This section describes the user interface, visualization, and calculations of certain gait-related parameters in the Data Handler. Information involved with the cyber and the physical systems such as communication, commands list, and details about some of the Data Handler functions will be described in communication section 4.4.

4.2.1 User Interface

The Data Handler software is designed completely using java for portability. Its main objective is to visualize sensor data from more than one insole and allow all CPS devices to be monitored on the same page. This is useful for monitoring both feet of a person simultaneously, or for a coach monitoring a sports team activity while holding a single mobile device.

Once all devices are set up and ready to stream data, the user can send a "start" command (details in section 4.4) to start the CPS operation. The Data Handler will start by connecting to the pre-defined database and create a new table/collection named after current date and time (i.e. YYYYMMDD:hhmmss). After that, the Data Handler will send a TCP command to all connected devices to start streaming data to its local IP address, and provides a unique port number to each device. In addition to the source IP address, assigned ports allow the Data Handler to tunnel different streams into different allocated memory locations and differentiate between different sources.

4.2.2 Visualization

The Visualization Module reads from these memory locations and visualize data for each source. Each physical device (source of data stream) will be allocated a separate panel depending on its setup. Figure 22 shows an example of two insoles streaming data to the Data Handler. The display is divided into two panels: the left panel shows data related to the left insole, and the right panel shows data related to the right insole. Each panel is also divided into three parts. Top left: shows a heat-point display of each FSR sensor to visualize force intensity. Under each heat-point, the value of that sensor's reading is displayed. Bottom left: shows 3-axis accelerometer readings and 3-axis gyroscope readings. Finally, the right side of the device's panel shows a 3D visualization of the actual orientation of this insole using the IMU data.

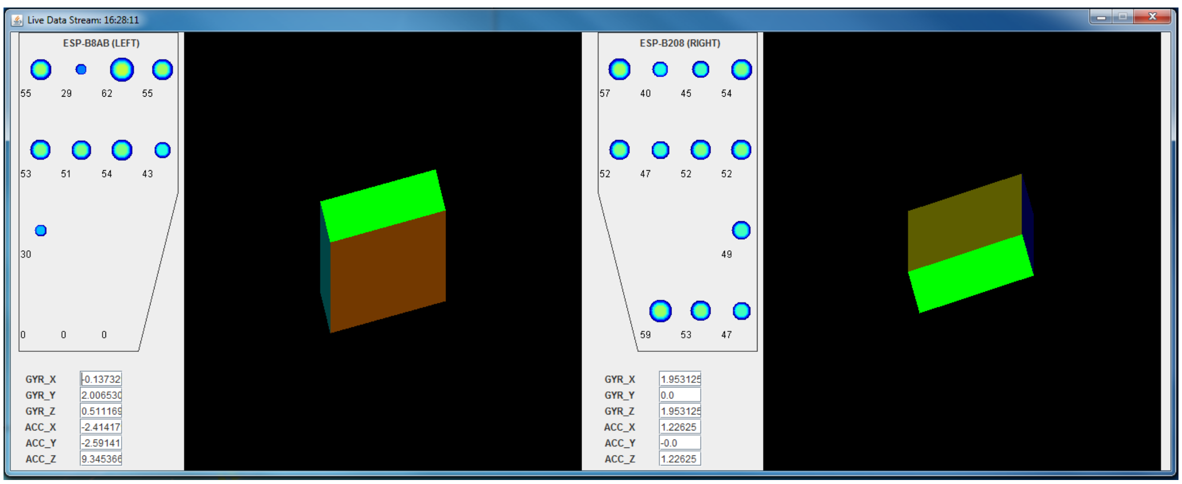

Figure 22: Live visualization in Data Handler showing two panels for two insoles (right and left). Each panel contains 3 parts. Top left: FSR sensor values and visualized force pressure. Bottom left: Accelerometer and gyroscope readings. Right: Live 3D visualization of insole orientation.

4.3 Data storage

A database management system (DBMS) is implemented within the CPS as the Data Storage Communicator Module of the framework shown in Figure 4. This DBMS collects processed data coming from the Data Processing Module. It converts each received set of sensory data from each source into a JSON object ready for uploading to the DB. This JSON object, in addition to current sensory data for this device, contains the source device's name, side (right or left), and a timestamp based on the instant it was received. In order to avoid concurrent modification errors that arise from manipulating the same memory location, all of the created JSON objects that are ready for upload are dumped into a "Shared Bucket" that contains data from all devices in the system, and once the number of objects in this bucket reaches a preset threshold, the Shared Bucket is dumped into an "Upload Bucket" that will be used to upload data to the DB. Using the pre-defined DB connection, the Data Storage Module uploads the whole "Upload Bucket" into the database. The threshold in this implementation is set at 1000 frames. However, this threshold can be changed depending on "freshness" requirements of the data stored in the in the database. For more up-to-date cloud data, this threshold should be decreased. If connectivity bandwidth is an issue, however, this threshold can be increased to minimize creation of connection links to the database.

For this implementation, dynamic NoSQL MongoDB was used [118]. A cloud database is defined in the system for when internet connection is available, and a local MongoDB database is also installed and set up in the testing computer. The Data Storage Module is also responsible for checking whether the cloud database is reachable or not. If not, all data in the Upload Bucket will be stored in a backup local database.

4.4 Communication

The communication module is responsible for switching communication modes between the cyber and the physical sides of the system. The mode of communication depends on the type of data being transmitted between endpoints. A set of instructions are predefined on each side of the system in order to facilitate different processes on either side. These instructions and their responses are transmitted on top of TCP protocol using HTTP. The whole system communication switches into UDP mode once data streaming is initialized. The system maintains UDP communication mode until a “STOP” request is sent to the physical system.

There are two sets of instructions predefined for each side of the system to facilitate seamless communication between the cyber side and the physical systems. The set of instructions implemented in the physical system are described in Table 5. These instructions allow the physical system to be controlled by requesting through the microcontroller’s serial communication interface, or using the communication module of the CPS, which is Wi-Fi in this case.

Table 5: Instruction set for the physical system (insole)

Instruction	Usage
vers	Returns the currently installed firmware version.
stat	Returns the current connection status of this physical system and provide connection details. It shows the device unique ID (composed of the last 2 bytes of its MAC address), IP address as a gateway, MAC address. If the insole is connected to a network, this instruction also adds the SSID of the connected network and the device's IP address as a client.
scan	Returns a list of Wi-Fi networks reachable by the microcontroller.
conn	The microcontroller requests an SSID and an encryption key, and attempts to connect using the provided credentials. Possible responses are: Success, Failed: cannot connect, and Failed: already connected.
disc	Disconnects the microcontroller from its currently connected Wi-Fi network.
data	Returns current readings of all available sensors. If this instruction is received over Wi-Fi, the microcontroller assumes this is a stream request and, thus, goes into UDP stream mode and stops when the device receives a "STOP" command over UDP (communication module described in section 3.5).

The Data Handler, on the other hand, uses a set of instructions called "DH commands" which are described in Table 6. These commands are designed to perform all kinds of operations required by the CPS including adding new physical systems (slaves) to the CPS, controlling and retrieving information from the attached physical systems, setting visualization and storage options, etc.

Each command in Table 6 translates into one or more sub-instructions that work together to achieve the described task. **Appendix B** contains a set of flowcharts to show and explain how some of the implemented DH commands work.

Table 6: Instruction set for the Data Handler (Cyber System)

Command	Usage
start	Start CPS operation: Request stream, visualize data and upload to DB
setup	Setup a new insole to connect to my Wi-Fi network.
setSide	Set existing insole side (Right or Left)
add	Add a new Insole using its IP address (must be in my network)
addAll	Add all detectable devices in my network to *StoredDevs* file
remove	Remove an insole from the CPS AND from the network
removeAll	Remove all devices from *StoredDevs* file
swap	Swap device locations in the visualization panel
stop	Force stop stream from ALL Insoles in the network
showAll	Show all stored devices
refresh	Request status update and network settings from all stored devices
export	Export the last stored collection to CSV
autoExp	Toggle auto-export to automatically export last stored collection to CSV
imu	Toggle IMU display (3D visualization of device orientation)
exit	Exit the program

4.5 Chapter Summary

Chapter 4 covered the implementation details of the proposed framework, as explained in Chapter 3, starting with the SmartInsoles, which are the physical system's proof of concept. The circuit design and PCB fabrication process is described showing how the final product is built as a multi-sensory health device. After that, the software design of the different modules of the cyber system is explained showing how visualization and database preparation and communication is performed. Finally, the implemented instruction set used in the overall CPS is listed and explained.

Chapter 5 shows two evaluation experiments to test mobility, configurability, data analysis, and data accuracy.

Chapter 5 Evaluation

Two experiments were performed to evaluate the implemented gait monitoring medical cyber-physical system. This chapter explains the details of the performed experiments, as well as the results and the derived conclusions.

5.1 Gait Parameters Measurement and Analysis

The first experiment involves using the implemented insoles and the Data Handler system to monitor gait in different situations for a soccer player. A young adult subject participated in the controlled experiment to validate the system. One insole was placed in each of the subject's shoes, and the subject was asked to perform different tests. Foot pressure data was recorded and then extracted from the database after the experiment for analysis. The goal of this experiment was to find distinguishable patterns from the collected data which can be used in the future to identify different gait phases (i.e. walking, running, gait direction, etc.). In addition, this experiment tested mobility of the system by performing it in a foreign Wi-Fi network. The subject was asked to repeat each test multiple times, and all tests were filmed using a video camera in order to use the footage as ground truth.

5.1.1 Measurement Protocol

A total of 16 tests were performed in this experiment. The subject was asked to repeat each test 3 times to compare results and ensure accurate readings. The tests were divided into two groups:

Group A: The first group of tests involved **normal** gait behaviors that can be accounted for all types of users. The tests included the following sets:

1) Normal walk – 30 meters:
    a) Walking forward.
    b) Walking backward.
2) Stationary jumping – 30 seconds.
3) Jogging – 30 meters:
    a) Jogging forward.

    b) Jogging backward.
4) Sprint – 50 meters.
5) Running around a circle – 3m diameter, 3 rounds:
    a) Clockwise.
    b) Counter-clockwise.

Group B: The second group of tests were specific to **soccer** related activities. These tests include the following sets:

1) Dribbling – 30 meters:
    a) Dribbling with right foot.
    b) Dribbling with left foot.
2) Running around a circle with a soccer ball – 3m diameter, 3 rounds:
    a) Clockwise.
    b) Counter-clockwise.
3) Receive and shoot:
    a) Receive from the left side.
    b) Receive from the right side.
4) Penalty kicks:
    a) Shooting with left foot.
    b) Shooting with right foot.

All the above tests were performed in an open field which resembles a normal everyday environment. Theoretically, this produces more realistic results compared to literature controlled experiments which were performed either indoor or with wired devices that restrict free movement. The subject was also wearing a Fitbit activity tracker that has the ability to monitor the number of steps and the user's heart rate.

5.1.2 Analysis Considerations

The legend described in Figure 23 will be used for describing data in all following graphs throughout this section. Each insole described in the implementation contains 12 FSR sensors. The FSR sensors in the left insole will be referred to as L[n], where [n] is the FSR sensor number (refer to Figure 20 or Figure 21 for sensor locations). Similarly, for the right insole, FSR[n] will be referred to as R[n]. The average of all FSR readings is an important factor used in the analysis of these results. Therefore, the average for the left and right FSR sensors' values will be referred to as Avg_L and Avg_R respectively.

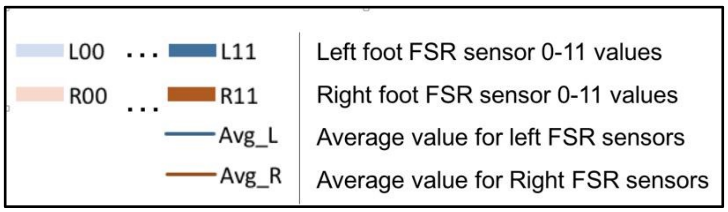

Figure 23: Legend description for below figures in this chapter

Before conducting this experiment, a number of tests were performed to find a pattern that can help identify different gait phases. Figure 24 and Figure 25 below show a sample of data from 1 step from each foot would look like for walking forward and backward respectively. Figure 24 shows that the average force reaches its peak at the end of the step for walking forward, while the opposite can be seen for walking backward (Figure 25). The reason for this is mostly due to the distribution of FSR sensors in the insole. More sensors are placed at the front of the foot compared to the heel area, and the average person steps on the heel before the front of the foot when walking forward, and the exact opposite when walking backward. This walking pattern results in more sensors being triggered at the end of the step when walking forward which results in a higher average force being applied on the overall sensors at the end of the step. The same concept can be considered for the average person walking backward. The front of the foot is normally stepped on first, and the heel is stepped on later when walking backward. Results show that, for the same reason of the distribution of sensors, the peak occurs at the beginning of the step.

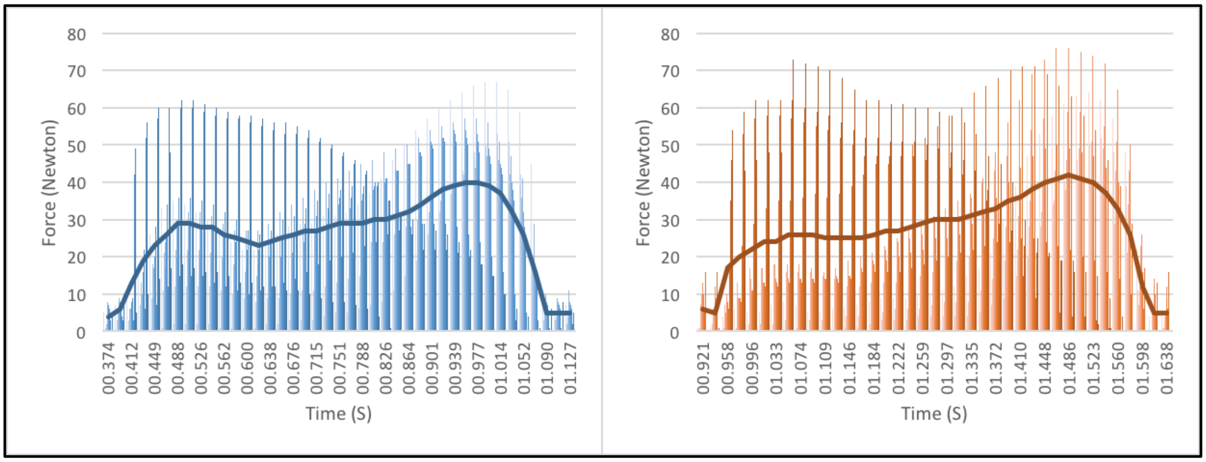

Figure 24: FSR readings for walking forward showing data for one step for the left foot (left) and the right foot (right)

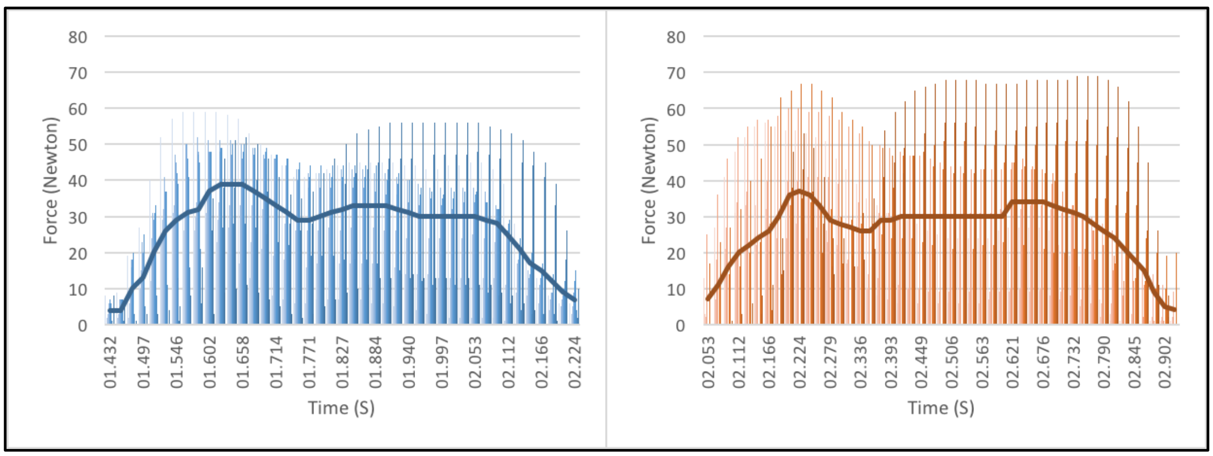

Figure 25: FSR readings for walking backward showing data for one step for the left foot (left) and the right foot (right)

5.1.3 Experiment Results

This section shows the graphs and the description of actual data retrieved from performing the experiment described above. The following section contains more details and analysis of the retrieved results, and shows conclusions driven from this analysis.

Figure 26 through Figure 33 show applied force pressure on FSR sensors for the left foot in blue (top), and for the right foot in orange (bottom) for 5 seconds. The applied pressure force on FSR sensors is measured and converted to Newton unit. However, this study is meant to be qualitative and not quantitative. For this reason, the applied pressure is not considered for its accuracy, but rather for the rate of change in the overall values of sensors. This is better represented by the average readings of the left and right FSR sensors, which are represented as Avg_L and Avg_R in the figures.

Figure 26 and Figure 27 show the data for when the subject is walking in a natural pace in both forward and backward directions respectively. What we notice in the forward walking test (Figure 26) is that the highest peak of the average values of FSR sensors, for both feet, always comes at the end of the step. For Figure 27, however, the opposite is shown where the peak occurs at the beginning of the step. This information helps in identifying the walking direction of the person using the insoles in terms of moving forward or backward.

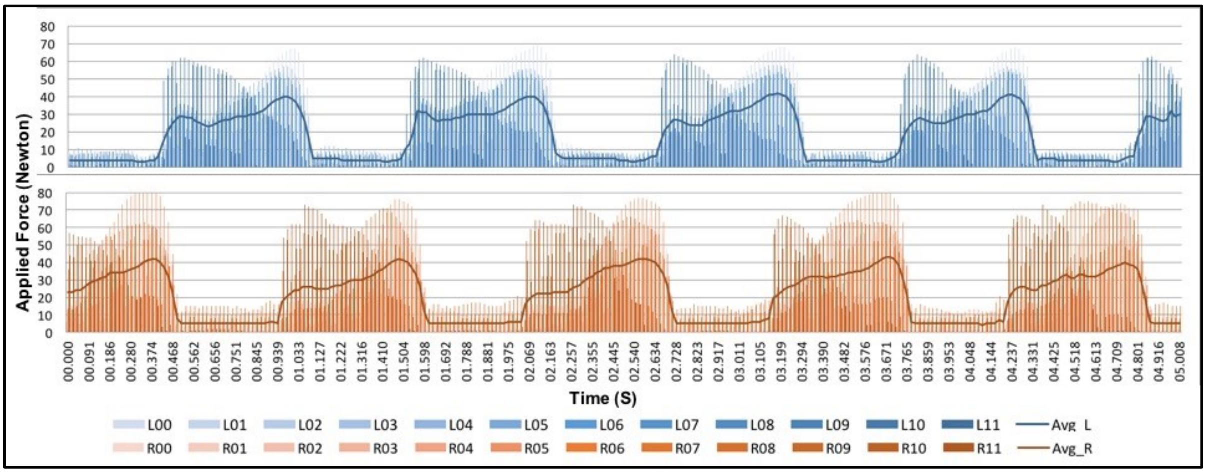

Figure 26: Walking forward

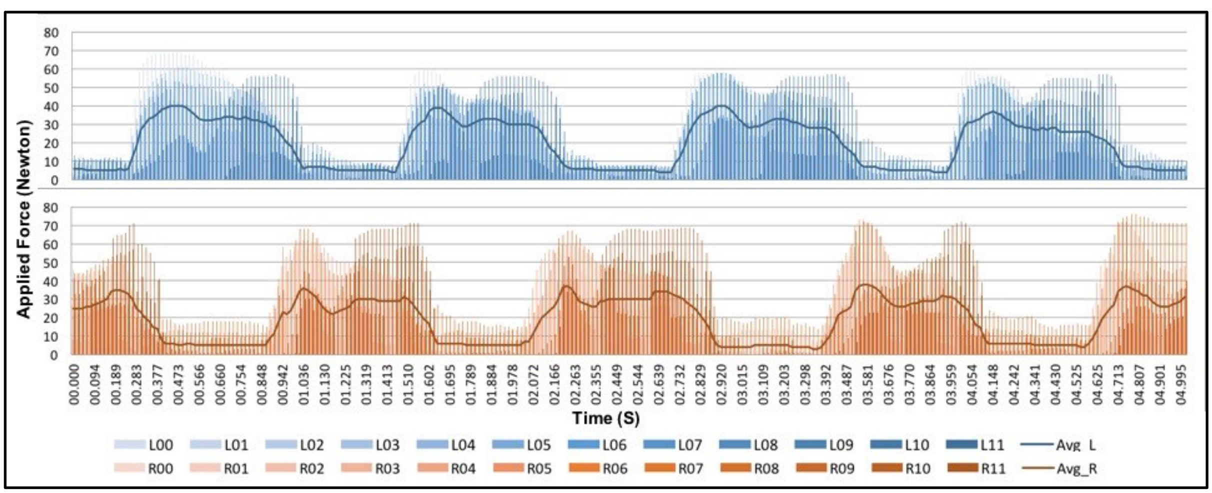

Figure 27: Walking backward

Figure 28 and Figure 29 show measurements for jogging forward and backward. Contrary to the walking graphs above, the direction (forward or backward) of the user cannot be always identified using the information from the peak average of FSR reading values. The ground truth video footage shows that the user always steps on the front of the foot while jogging (and sprinting). Therefore, identifying whether the user is first stepping on the front of the foot or on the heel cannot be used here as was the case for walking. However, the jogging phase can be easily distinguished from the walking phase by analyzing the stance time and the swing time (more details on gait parameters are discussed in section 5.2.1). The durations of these two parameters in the jogging phase are much smaller when compared to the walking phase.

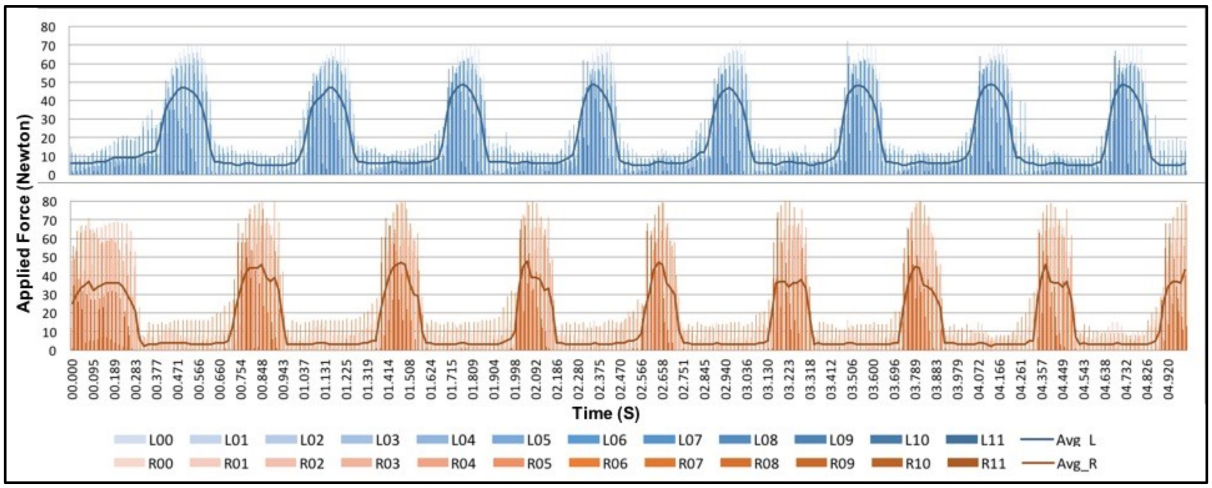

Figure 28: Jogging forward

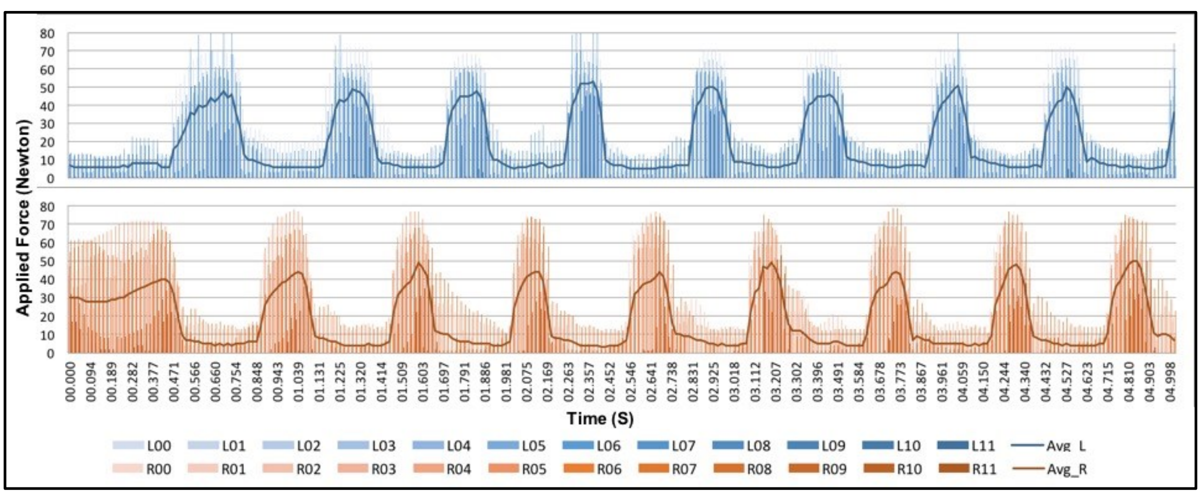

Figure 29: Jogging backward

Figure 30 shows results for when the user was jumping in a fixed location. Since all insoles are synchronized using one central clock, the samples' timestamps can be used to detect which step occurred before the other. The jumping phase graph clearly shows that the average pressure values in both feet are peaking at approximately the same time. This information is used to identify that the user is currently jumping. However, if the user is alternating feet while jumping (i.e. jumping on one foot before the other) results can be confused with jogging or sprinting. IMU values can be tested for the possibility to solve this problem in the future.

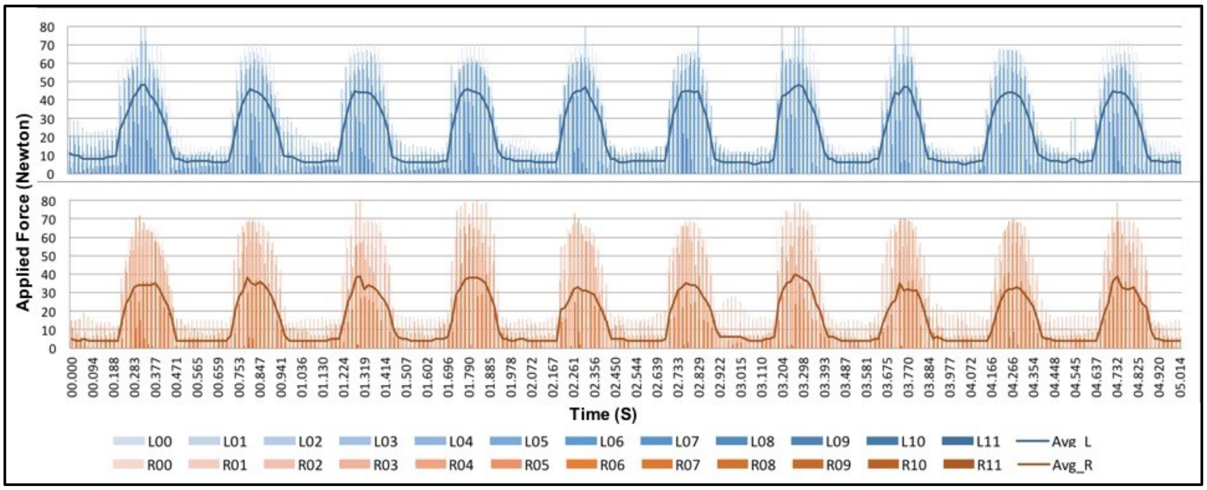

Figure 30: Jumping

Similar to the identification of the jogging phase compared to the walking phase, the sprinting phase's stance and swing times are even smaller, which naturally shows more steps occurring in the same time period. Figure 31 shows results for the user sprinting as fast as he could, and the number of steps occurring in the 5-second snippet is higher than its counterpart in jogging. This information is used to identify the sprinting phase for the user.

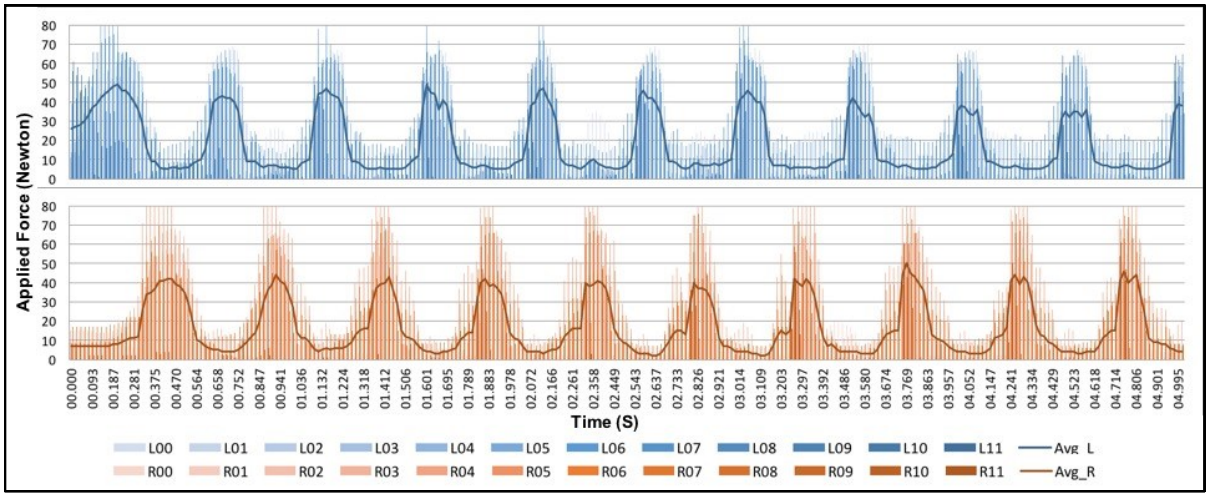

Figure 31: Sprint

One of the objectives of this study was to measure the difference in pressure points for the right and the left feet for when the user is running while turning right or left. Figure 32 and Figure 33 show results for when the user is running around a circle with 3-meter diameter. After analyzing data from these graphs, we conclude that the current information from FSR data is not sufficient to identify the turning direction. However, IMU values can be tested in the future for the possibility of identifying the user's turning direction. Having said that, the below figures do show an unstable pattern and an irregularity when compared to previous figures where the user is jumping, or moving forward/backward. Therefore, this piece of information can be used to identify that the user is currently turning to the left or to the right side.

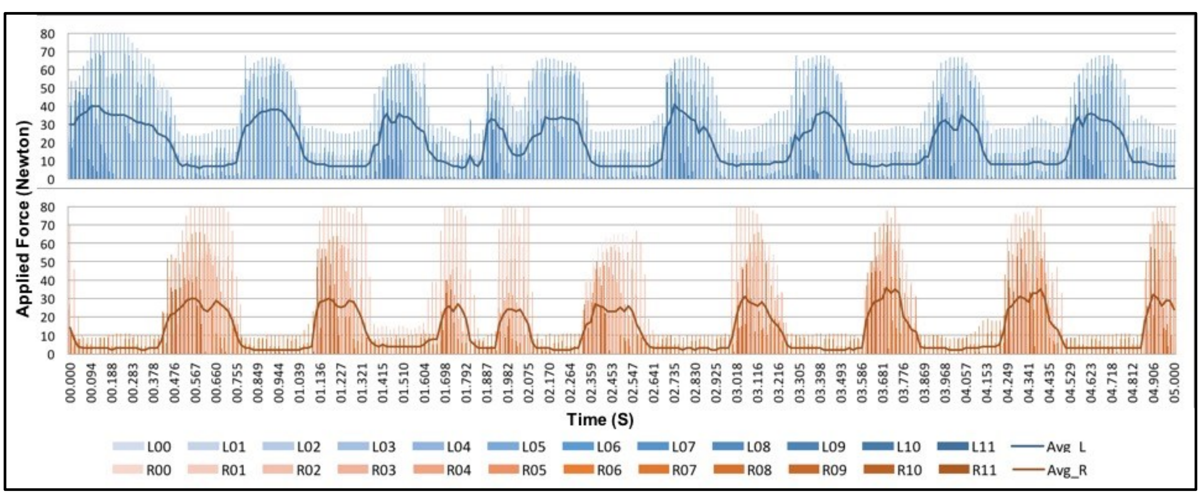

Figure 32: Running around a circle - clockwise

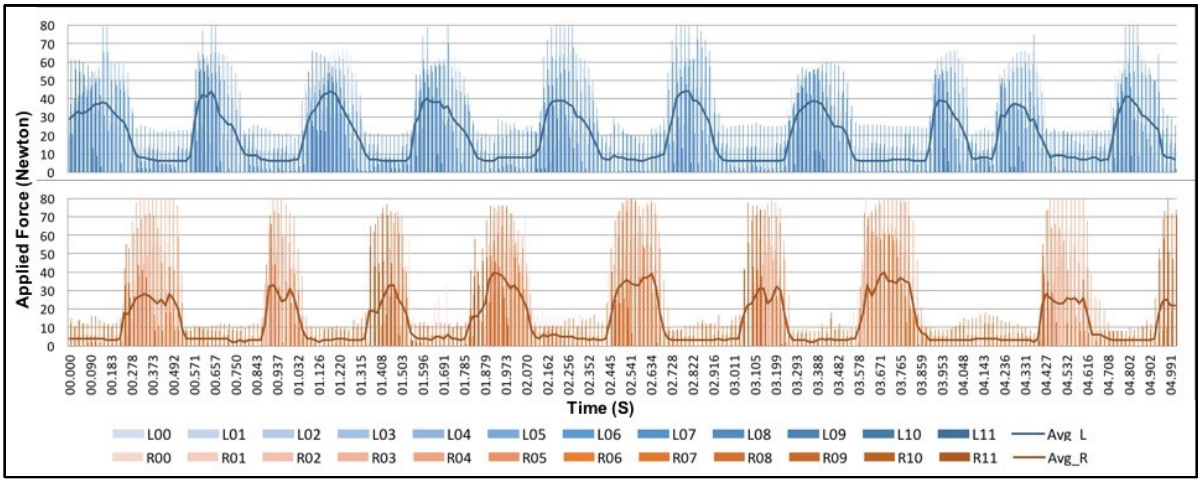

Figure 33: Running around a circle - counter-clockwise

5.1.4 Results Analysis

By analyzing the data in the above figures for forward and backward directions, a noticeable pattern can be observed. Results also show distinguishable patterns amongst different gait phases. The graphs in figures Figure 26 through Figure 33 show a 5-second sample of the data collected from the non-soccer related tests described earlier (Group A). The graph patterns for walking forward and walking backward show patterns that can be easily distinguished from each other. For walking forward, the peak average force appears to be at the end of the step, while a person walking backward normally steps on the front of the foot before the heel, which is why the graph shows a peak at the beginning of the step instead. Furthermore, for jogging, direction cannot be easily distinguished visually from the patterns using the average applied force, but an increase in the number of steps performed is visible in the same 5-second duration. Accordingly, we can identify a jogging, and sprinting, patterns from the 3rd, 4th, and 5th graphs. The last graph demonstrates a clear jumping pattern by showing that the average pressure forces imposed by both feet are increasing and decreasing at an approximately parallel pattern.

These results are derived from visual analysis of the collected sensory data. Automatic classification can be a very helpful and useful asset here, which is why machine learning is suggested to differentiate between different gait classes for the same user.

Some of the studies discussed in the literature review (Chapter 2) were able to detect step to step duration, which was also achieved here. However, the distribution of sensors in this work's implementation enabled us to get more details on the direction the user gait. In addition, designing a system that runs completely over Wi-Fi provided a more comfortable and natural environment to test user's gait. The performed tests were conducted outdoors without the restrictions of a lab setting or a treadmill, and the subject user was able to move up to 50 meters away from the system's main router.

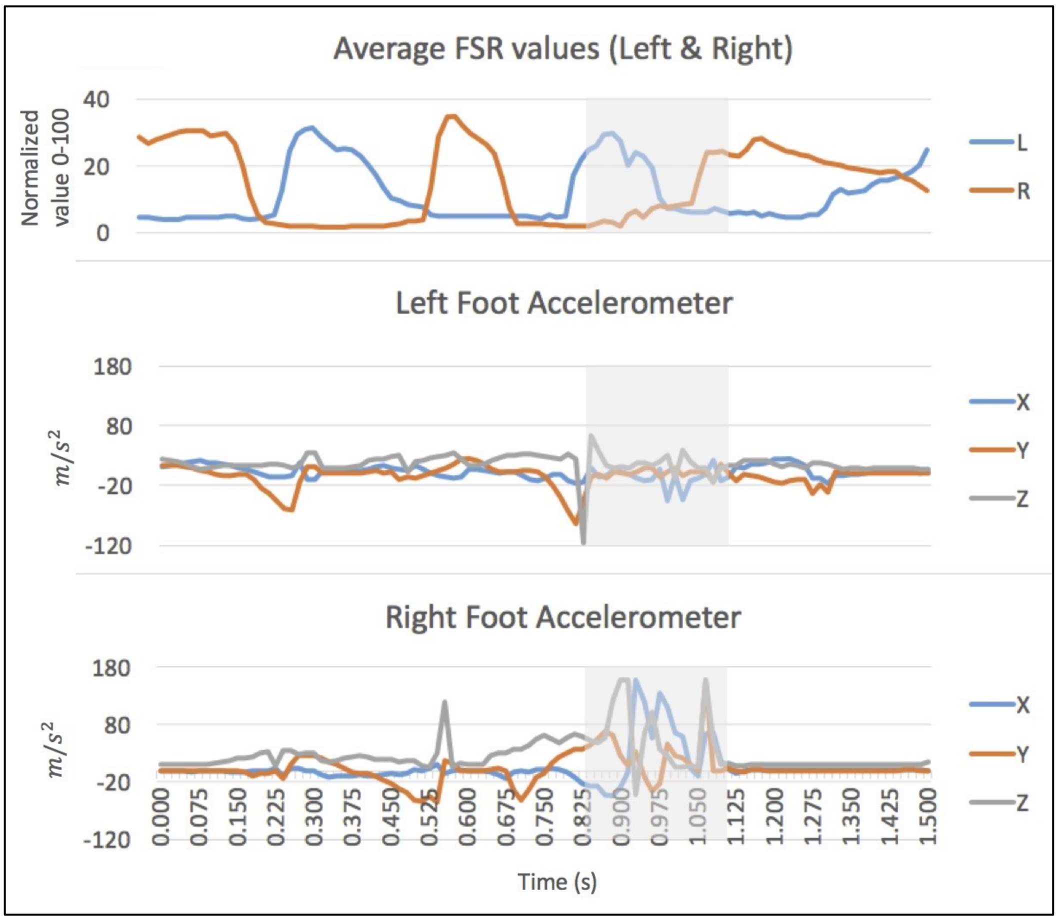

Figure 34: A graph showing FSR values and accelerometer values at the event of kicking a soccer ball. Highlighted area shows high disturbance in the right accelerometer which accurately estimates the time the ball was kicked using the right foot

A number of studies have used the IMU sensor to collect detailed information related to step detection [26], [51], [68], [82] or to measure foot orientation [56], [82], [119], which is calculated in our experiments. Weizman et al. have conducted an analysis on force applied to kicking a soccer ball [70]. However, difficulties arise when measuring the moment and the duration of which the soccer is in contact with the foot. We tested the use of the embedded IMU in detecting the short timeframe the soccer ball is kicked by the user, which is typically a window of about 10 milliseconds [71]. By analyzing the synchronized data produced by the embedded sensors in the insole, we were able to narrow down the timeframe where the ball was in impact with the shoe to approximately 100 milliseconds, as well as detecting the foot that was used to kick the ball Figure 34. This was achieved by finding a period of strong noise in IMU readings as can be seen in Figure 34, which shows a 1.5 second sample of FSR and IMU data for both feet of a single soccer player. The graphs show a clear disturbance in the right foot's IMU readings which identifies the exact instant the ball is kicked. This was confirmed by comparing this finding with the ground truth video which was recorded for the whole experiment. Multiple tests were performed for shooting a soccer ball, as described earlier, and data shows accurate results for all tests. Future work also involves machine learning for automatically detecting the instance a soccer ball is kicked and the magnitude of the kick using the IMU sensor in the implemented CPS.

5.2 System Validation with Tekscan Strideway™

The Tekscan Strideway is a modular human gait analysis system used to analyze detailed spatiotemporal parameters [120]. The system is packed into a large case and weighs about 40 kg. It mainly consists of six tiles, four of which contain hundreds of embedded force sensors, a tile for gait initiation and a tile for gait inhibition. The assembled system covers an area of approximately 4 m^2 and is wired via a USB cable to a nearby computer containing the Strideway software. This 4-m^2 area is where the subject's gait data can be collected. Although the full Strideway system was only officially released in 2017, the technology/software used for this system was built based on the one-panel MatScan® pressure mat, which has produced reliable measures for human gait [61], [121] and reliable and valid measures of postural stability [122], [123].

This system is widely used in physiotherapy and rehabilitation applications in many health organizations. The provided software automatically reads from the physical sensors in the mat and performs calculations to derive different gait parameters such as step time, gait time, cadence, velocity and walked distance. It can also automatically detect which foot is right and which is left, which is useful in calculating toe-in/toe-out angles to compare to the subject's line of progression. During the tests, the software shows a visualization of the performed steps in the form of a heat-map. This display shows the locations of the performed steps on the mat and each step's pressure intensity. After the test, the Strideway provides detailed analysis in the form of graphs and tables showing calculated spatiotemporal data collected in the test.

Validating the implemented CPS proves that it can replace an industry-level device that is used widely in physiotherapy and rehabilitation in many health organizations. Therefore, an experiment was performed with the help of the Health Sciences Department at the University of Ottawa.

In order to validate the implemented system, a comparison experiment was conducted with the help of the Dr. Sarah Fraser and her team in the Health Sciences Department. Proving that an in-house designed and implemented CPS can achieve similar, or even better, functions and tasks than a trusted industry CPS will enable further enhancement in this area of research and extending the use of the much cheaper system. In addition, the implemented CPS discussed here allows more freedom of movement and is not restricted to indoor use. The experiment in section 5.1, for example, was conducted completely outdoors.

Details and discussions on this validation study were published in the Journal of Sensor and Actuator Networks in August 2018 [55].

5.2.1 Study Design and Analysis Parameters

Fifteen subjects were tested in this experiment, 8 females and 7 males, and their ages ranged between 20 and 45 years (average 31 years). The Tekscan Strideway mat was assembled in a spacious area and connected to a laptop that contained the Strideway software. Green tape was used to create a line that marks the start and the end of the actual reading area of the mat, and a video camera was used to record the lower body of test subjects. By referring to the camera footage, we used the green line to identify the first step and synchronize the steps acquired by the two systems. All subjects signed a consent for inclusion before they participated in the study. The study was conducted in accordance with the University of Ottawa Research Ethics Board, and was approved by the Office of Research Ethics and Integrity at the University of Ottawa (H10-17-08).

The goal of this experiment was to allow different subjects to walk at a normal pace, while acquiring their foot pressure data by both systems simultaneously. The outcome we aimed towards in this experiment was having comparable, ideally identical, time-related results for different gait parameters. Having this outcome supports the goal of validating the implemented CPS to be used as a reliable gait monitoring and measurement system.

We focused on seven gait parameters in this study. Assuming the user was always walking forward, the initial contact of the person with the ground was normally at the heel. Therefore, one stride (gait cycle) began with heel contact (initial contact) of one foot with the floor and ended with the next heel contact of that same foot. Consequently, one complete gait cycle consisted of two steps, one of the right foot and one of the left foot [124]. Each gait cycle was divided into two phases: stance phase and swing phase. Stance phase began with the initial contact of one foot and ended with the toe-off of that same foot. Swing phase began with toe-off of one foot and ended with the subsequent heel contact of that same foot [125]. Stance phase usually made up about 60% of one gait cycle, and swing phase made up the remaining 40% of that gait cycle [59].

There were two periods during one gait cycle of normal walking where both feet were in contact with the ground; these were the initial and terminal double support periods. Double support time began with initial contact of one foot and ended with toe-off of the other [59]. Therefore, considering the right-foot to be the dominant foot, initial double support would start from the initial contact of the right foot and end at toe-off of the left foot. Consequently, terminal double support would start from initial contact of the left foot and end at toe-off of the right foot. For the purpose of this study, we added the average durations of the initial and terminal double support times and reported the average of total double support time for each trial.

The step time was the time duration from the initial contact of one foot to the initial contact of the subsequent step of the other foot. The gait time was the duration between the first contact of the first step and the first contact of the last step in each test. It started from the initial contact of the first step and ended with the toe-off of the last step, regardless of which foot the participant started or ended his/her gait. Cadence is calculated as the number of steps the user performed per minute. Figure 35 below shows different described gait phases and indicates each parameter's starting and ending point. In this study, the following components are collected from both systems:

- Stride time (gait cycle time)
- Stance time
- Swing time
- Double support time
- Step time
- Gait time
- Cadence

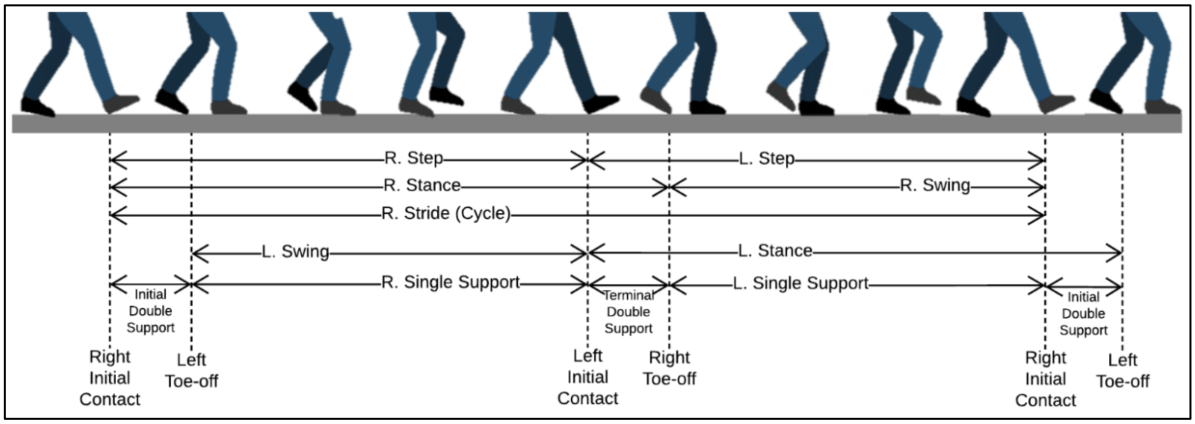

Figure 35. Description of start time and end time for different gait phases in a normal gait cycle.

5.2.2 Measurement Protocol

Multiple pairs of the SmartInsole were developed for different shoe sizes. Each insole was placed in a size-adjustable sandal, and subjects were asked to wear the pair with a size closest to their shoe size. Before each subject performed a test, the experiment procedure and purpose were explained to them. They were, then, asked to sign a consent form that explained the purpose of the experiment again and indicated that the collected data would be used strictly for research purposes. Identifying information was not collected. Since the first plate in the Strideway did not contain sensors and could not collect data, we used it as the gait initiation stage. Therefore, at the beginning of each walk, subjects were asked to stand on one of the marked boxes (as seen in Figure 36), walk normally towards the other end of the mat and stop at the final box. After storing data and labeling them for post-experiment analysis, subjects were directed to stand on the starting box again for their next walk. As indicated earlier, each subject was directed to perform 10 rounds of walks.

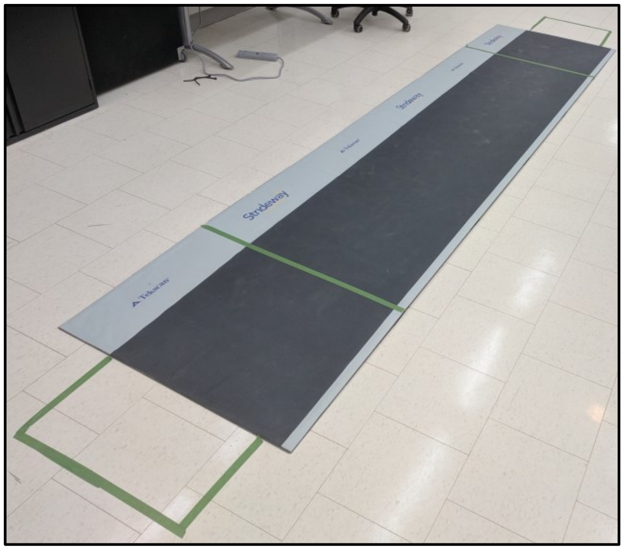

Figure 36: Tekscan Strideway system showing marked start and stop boxes, as well as lines to indicate the start and end of the reading area (the 4 tiles with reading capability)

5.2.3 Methodology for Agreement Analysis

One of the main features of the Tekscan software is its ability to automatically produce spatiotemporal data details for tests performed on the Strideway mat. The results of each test are shown in the form of a video and information tables. The video shows a digital representation of the mat and the location and pressure intensity and distribution, in the form of a heat-map, of each step preformed during the test. In addition to the video, the software provides five tables for five different groups of data. These tables provide rich information on the step-stride times and forces applied, gait cycle times and the overall gait characteristics such as the distance, velocity, cadence, etc. In addition, two more tables are provided to show differential and symmetry characteristics by comparing data acquired from each foot.

The designed CPS, on the other hand, provides accurate force and time details for each of the 12 embedded pressure sensors from all of the connected SmartInsoles. These force values are used to extract timestamps for initial contact and toe-off of all steps that occurred during the operation of the system. After that, these timestamps were used to calculate all of the required gait parameters in this experiment.

Since the designed CPS reads all data coming from the SmartInsoles, some steps may not be read by the mat. Therefore, video footage was required to determine the number of first steps that occurred on the reading area of the Strideway. This process was needed to match/synchronize the correct step data acquired from each system.

Since it is almost impossible to get identical values acquired by two different systems, the amount of difference that was likely to occur between them was investigated, and an accuracy indicator was derived based on the results. The agreement accuracy was calculated for the results of this experiment using two methods. The first method, overall accuracy, looks at the overall values produced by both systems and derives the accuracy percentage based on the summations of values. The second method, average accuracy, looks at individual measures and calculates the accuracy for each pair of results from the two systems. Then, it is calculated as the average of all accuracy values derived from each pair of values (from the Strideway and the insole system). Overall accuracy and average accuracy were calculated using Equations (7) and (8), respectively.

$$Overall\ Accuracy = \left(\sum_{x=1}^{n} SW(x) - \left|\sum_{x=1}^{n} SW(x) - \sum_{x=1}^{n} IN(x)\right|\right) \Big/ \left(\sum_{x=1}^{n} SW(x)\right) \tag{7}$$

$$Average\ Accuracy = \left(\sum_{x=1}^{n} \frac{SW(x) - |SW(x) - IN(x)|}{SW(x)}\right) \Big/ n \tag{8}$$

where:

x	is the trial/test number
n	is the total number of trials/tests
SW, IN	are the gait parameter values derived from the Strideway and the insole systems, respectively.

5.2.4 Experiment Results

Ten walks were performed by each of the 15 participants on the Strideway while wearing the SmartInsoles. Spatiotemporal data for over 450 unique steps were collected and recorded. Each step collected by the SmartInsoles was analyzed carefully and compared to its counterpart from the Strideway. In this section, we present the results of this experiment, where SW represents the results acquired by the Strideway system, and IN represents the results acquired by the implemented SmartInsoles' CPS.

Each of the 12 FSR sensors embedded in the SmartInsoles produced its own force value between zero and 100 Newton. The values acquired from all tests that were performed in this experiment were analyzed to retrieve each step's heel-contact and toe-off timestamps. These timestamps were then used to calculate the gait parameters as described in Figure 35.

The data used to plot the graphs in Figure 37 were the results produced after averaging left and right foot data for each of the seven analyzed gait parameters. This way, a single value was produced for each trial by each system. MATLAB (R2018) was used for all statistical analysis and to generate all graphs in Figure 37. In some cases, during the experiment, one of the systems could fail to deliver results for a specific test. Data from both systems were eliminated in such cases, and their results are not reported here. The final used sample, based on which all the results and graphs were derived, included 81 pairs of successful tests, which was 54% of all the performed tests.

Agreement Comparison Graphs

Three types of graphs were generated for each gait parameter in this study: a scatterplot, a histogram and a Bland–Altman graph.

Scatterplots were used to find a correlation between two sets of data. In this study, we compared data from two systems: Strideway data (x-axis) and the insoles' CPS (y-axis). A trend line (or a line of best fit) on a scatterplot was used to represent the trend of the data and indicated the likelihood of results if other data were present. In each parameter's scatterplot shown in Figure 37, a trend line is shown as a continuous straight line. Since this was a comparison between the results of two systems measuring the same information, the trend line should ideally fall exactly on the 45-degree reference line ($y = x$), which is shown as a straight dashed line. This would mean that the data from both systems were identical. Therefore, a trend line that was close to the reference line indicated a trending agreement between the two systems' results.

Histograms show the number of occurrences of a value in a specific range. The values shown in the histograms in Figure 37 represent the time difference between the Strideway data and the insoles' data for each of the analyzed parameters. The width of each bar (bin) represents the range of error, and the y-axis shows how many times an error falls within this bin's range during our experiment. A straight line is also shown in all histograms to indicate the average difference between the values read by each system. The ideal case is when this line stands on the x-axis' zero mark. In these histograms, values should follow a bell-shaped normal distribution, and the peak y-axis value should be around the zero x-axis value.

A Bland–Altman graph is another way to analyze the agreement between datasets acquired by two systems. The y-axis represents the difference between the two values, while the x-axis represents the average of these two values. Furthermore, three horizontal lines can provide more information on the acquired data in a Bland–Altman graph. The solid straight line represents the average difference between the values read by each system (also called the bias line), which has the same value of the vertical line in histograms, and the two dashed lines are the Limits of Agreement (LoA). The upper and lower LoA lines were calculated as the average value ±1.96 standard deviations. If the differences were normally distributed, 95% of the values would be between these two limit lines [126].

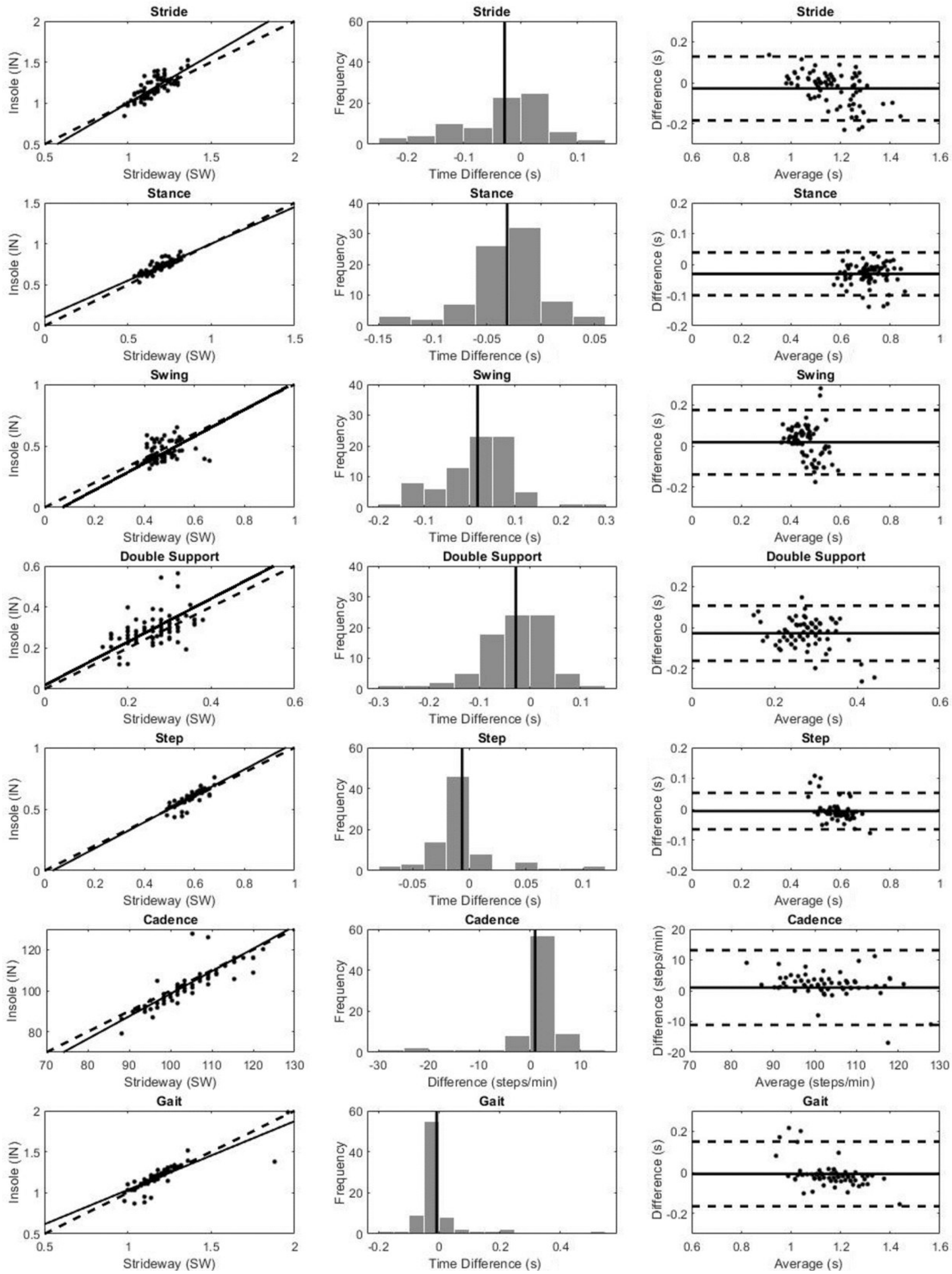

Figure 37. Scatterplots, histograms and Bland–Altman graphs that compare the data acquired by the Strideway (SW) and the SmartInsoles Cyber-Physical System (CPS) (IN).

Analysis of Agreement

Table 7 shows a numerical analysis and a summary of the data in Figure 37. The first column represents the analyzed parameters, and the mean difference is shown in the second column. The mean difference in this validation experiment is ideally aimed to be close to zero. As can be seen from this summary, the mean difference for time-related parameters (measured in seconds) ranged between −0.03 s and 0.02 s. This means that the average error between the acquisitions of both systems did not exceed 30 milliseconds. For cadence, however, the mean difference was measured at 1.01 steps per minute, which is also considered an insignificant error. The "Inside LoA%" shows the percentage of data that had a difference between the lower and upper LoA values. For example, data showed that 95.06% of the acquired cadence data had an error between −11.13 steps/min and 13.16 steps/min. The lower and upper LoA values in Bland–Altman graphs should wrap approximately 95% of the data, and a smaller range between the lower and upper LoA values meant higher agreement between the two sets of data.

Table 7: Analysis of agreement summary. Mean, median, standard deviation, range, lower and upper Limits of Agreement (LoA) values are all measured in seconds, while cadence is measured in steps/min. Range values represent the difference between the absolute min

Parameter	Mean	Median	St. Dev.	Range	Lower LoA	Upper LoA	Inside LoA%
Stride	−0.03	−0.01	0.08	0.37	−0.18	0.13	93.83%
Stance	−0.03	−0.03	0.04	0.18	−0.10	0.04	92.59%
Swing	0.02	0.03	0.08	0.46	−0.14	0.18	96.30%
Double Support	−0.03	−0.02	0.07	0.41	−0.16	0.11	93.83%
Step	−0.01	−0.01	0.03	0.19	−0.07	0.05	93.83%
Cadence	1.01	1.86	6.20	37.85	−11.13	13.16	95.06%
Gait	−0.01	−0.02	0.08	0.66	−0.17	0.15	95.06%

Table 8 shows the results of the five parameters analyzed in this study and compares the values acquired by both systems. The unit of measurement for all time-related parameters is seconds, and the unit for cadence values is steps per minute. Each row in Table 8 shows a calculated result that went through two stages. The first stage was averaging the left and right foot data for each parameter, and the second stage was averaging the first stage data for all 10 walks. Equations (9) and (10) below show how these values were calculated.

$$IN_{s,p} = \left(\sum_{x=1}^{10} \frac{IN_L(s,p,x) + IN_R(s,p,x)}{2} \right) \Big/ 10 \quad (9)$$

$$SW_{s,p} = \left(\sum_{x=1}^{10} \frac{SW_L(s,p,x) + SW_R(s,p,x)}{2} \right) \Big/ 10 \quad (10)$$

where:

s	represents the subject/participant
p	represents the gait parameter
x	is the trial number
IN_L, IN_R	are the left and right average values acquired by the SmartInsoles CPS
SW_L, SW_R	are the left and right average values acquired by the Strideway
$IN_{s,p}$	is the two-stage value for subject s parameter p acquired by the SmartInsoles CPS
$SW_{s,p}$	is the two-stage value for subject s parameter p acquired by the Strideway

This two-stage calculated value produces a single value for each subject and parameter, by each system, which makes them easily comparable. The last two rows in Table 8 show the overall accuracy and the average accuracy as described earlier using Equations (7) and (8).

Table 8: Average gait parameter values for each participant acquired by the Strideway (SW) and the Insole system (IN) after performing the 2-stage estimation (Equations (3) and (4))

Participant	Stride Time (s)		Stance Time (s)		Swing Time (s)		Double Support Time (s)		Step Time (s)		Cadence (steps/min)		Gait Time (s)	
	SW	IN	SW	IN	SW	IN	SW	IN	SW	IN	SW	IN	SW	IN
1	1.15	1.22	0.71	0.76	0.44	0.48	0.29	0.37	0.57	0.57	104.83	106.23	1.23	1.22
2	1.05	1.04	0.61	0.65	0.46	0.39	0.18	0.25	0.52	0.54	114.72	112.70	1.05	1.07
3	1.05	1.09	0.62	0.64	0.43	0.45	0.20	0.21	0.52	0.54	114.85	111.53	1.05	1.08
4	1.22	1.23	0.70	0.78	0.53	0.45	0.27	0.30	0.61	0.64	98.07	93.53	1.43	1.29
5	1.21	1.28	0.75	0.75	0.49	0.55	0.31	0.25	0.61	0.63	99.69	95.22	1.21	1.26
6	1.14	1.17	0.72	0.75	0.43	0.42	0.32	0.34	0.57	0.58	105.24	104.32	1.14	1.15
7	1.27	1.26	0.76	0.78	0.51	0.47	0.29	0.28	0.63	0.65	94.83	92.21	1.27	1.30
8	1.23	1.36	0.77	0.78	0.46	0.56	0.34	0.32	0.61	0.62	98.24	96.53	1.23	1.25
9	1.14	1.13	0.70	0.72	0.46	0.41	0.27	0.29	0.57	0.58	105.33	104.37	1.14	1.15
10	1.03	1.01	0.64	0.65	0.40	0.36	0.25	0.28	0.51	0.51	116.98	118.75	1.03	1.02
11	1.11	1.02	0.56	0.64	0.65	0.39	0.24	0.22	0.56	0.50	108.10	120.92	1.11	1.01
12	1.21	1.24	0.70	0.75	0.52	0.49	0.23	0.29	0.62	0.61	99.73	98.92	1.21	1.21
13	1.18	1.20	0.70	0.75	0.50	0.45	0.23	0.29	0.59	0.62	102.18	98.36	1.18	1.23
14	1.10	1.05	0.62	0.66	0.53	0.39	0.20	0.26	0.55	0.48	109.10	126.05	1.10	0.95
15	1.08	1.01	0.60	0.64	0.48	0.37	0.16	0.20	0.54	0.54	111.00	110.09	1.08	1.09
Overall Accuracy	**99%**		**95%**		**91%**		**89%**		**100%**		**100%**		**99%**	
Average Accuracy	**96%**		**95%**		**86%**		**83%**		**97%**		**96%**		**96%**	

5.2.5 Discussion

The current study is similar to Jagos et al. [59] in that the insoles were compared to a walkway pressure-sensitive system (Strideway by Tekscan). While it is certain that there will always be certain measurement differences due to the number of sensors in the insole versus the number of sensors in the gait mat, this study clearly demonstrates a high agreement between the Strideway and the insoles system. Jagos et al. revealed in their validation study that, for the healthy group, the average difference between the eSHOE and GAITRite acquisitions ranged between −0.029 and 0.029 s. The average difference in this study, as shown in Table 1, is between −0.03 and 0.02 s, which is very close to Jagos' results. However, it is worth mentioning that their subjects consisted of older adults (age 40.8 ± 9.1 years), while our sample's average age was 31 years, which makes it difficult to make a direct comparison.

In comparison to other validation studies [49], [65], [83], this study has the advantage of a large sample of steps (450 unique steps) with which to compare several gait parameters. In addition, based on previous research, the proposed SmartInsoles cyber-physical system was constructed to have the optimal number of sensors for sensitivity to gait parameters without unnecessarily increasing cost. Further, although many of the fabricated systems are wireless, this was one of the few systems that used Wi-Fi for real-time viewing of data from multiple devices simultaneously rather than Bluetooth [47] or integrated internal storage [57], [127].

5.3 Chapter Summary

Chapter 5 showed two evaluation experiments performed as part of this thesis. The first experiment focuses on gait parameters measurement and analysis using the implemented CPS described in Chapter 4. The subject user is a young adult athlete, and the objectives of the experiment were to detect different gait patterns using the designed CPS, and to test mobility by performing the tests using a foreign network. The second experiment was a validation study. The objective was to read spatiotemporal gait characteristics of 15 subjects using the designed CPS while walking on an industry-level device used by healthcare organizations. Results from both experiments were published in highly reputable conferences and journals. The objectives of the experiment described in section 5.1 were to test the mobility of the implemented CPS while simultaneously acquiring sensory data that can be used in detecting events and different gait phases. Results prove mobility of the system since it was used outdoors and using a foreign Wi-Fi network that was not previously configured in the lab. For the experiment in section 5.2, the objectives were to test configurability by using multiple sets of insoles, and multiple users. Users were using different sets of insoles at different times during the day, and the system could automatically recognize the operating insoles and start reading data from them. In addition, the collected data was used to acquire different spatiotemporal characteristics to compare with the Tekscan Strideway mat under the supervision of the faculty of Health Sciences. This experiment proved the reliability of the system in collecting gait information comparable to the mat that is a standard used by healthcare organizations. These experiments prove the reliability of the proposed communication protocol, which takes care of configurability of the system components and the mobility of users. Furthermore, the modularity of the proposed framework proves to be beneficial due to the ability of the system to provide data in real-time, and for post-routine analysis.

Chapter 6 concludes this thesis by providing a summary and a brief discussion and future work in the physical activity measurement research.

Chapter 6 Conclusions and Future Work

6.1 Conclusions

The proposed framework and communication protocol evolved by finding gaps in the current research in mobile gait monitoring systems. A framework was required to provide spatiotemporal data in real-time, while keeping the requirement of user and device mobility. The proposed framework is applicable to many physical activity monitoring applications, and is especially useful for group physical activity monitoring.

Performing the handshake setup stage to configure a new device in the CPS allows the system to have a digital twin of the physical system. This means that, while streaming data in real-time, we can also see exactly where the physical action is being performed. The PSP provides sensory details including its actual physical location within the health device. This enables mapping sensory data to its physical location in the user's body.

This thesis starts by providing a literature review on several research studies in medical systems focused on physical activity monitoring and gait analysis for well-being applications. Research directions show a clear trend towards wireless and real-time patients- and athletes-monitoring. These features require lightweight and easily configurable health devices that are able to wirelessly transmit data in real-time for monitoring, visualization, and storage. A great portion of these studies show implementations that require the user to perform analysis of their physical activities after all data has been collected [53], [59]. This is due to the high requirements of real-time processing of sensory data, and even higher requirements of real-time visualization of these sensory data. Therefore, some tried to overcome this by sacrificing mobility and using wired hardware that can transmit data in real-time to be processed on a nearby computer [51], [56], [57]. In addition, all of the systems surveyed in this research require users to manually configure the data-collecting health device within the CPS in order for it to operate. This makes it very difficult for systems that require group activity monitoring such as physiotherapy or sports coaching. Therefore, a gap is available in the literature for a physical activity monitoring CPS that is mobile, configurable, and able to process and present data in real-time. A framework is required for such a CPS that allows for mobile and wireless sensor networks and physical systems to transmit data, and perform real-time processing and visualization of these sensory information. In order to fill these gaps, a well-structured communication protocol is also needed to allow for seamless configurability and data transmission within the proposed CPS.

In this thesis, a CPS framework for measurement and analysis of physical activities and wearable sensor networks is proposed. We show the design and implementation of the CPS framework that enables real-time visualization of sensory data acquired from multiple multi-sensory devices simultaneously. As a proof of concept, a gait parameters measurement and analysis system was implemented that runs completely over Wi-Fi. The developed system performs gait analysis by transmitting compressed and encoded sensory information wirelessly to a cloud server, which in turn, performs historical data collection, as well as the required analysis to extract gait parameters.

To evaluate the proposed framework, a number of reliability studies were performed on the implemented CPS to test its ability to provide the necessary information, including one experiment that was performed in collaboration with the Interdisciplinary School of Health Sciences at the University of Ottawa. Evaluation results show high reliability and agreement between the two sets of data, and shows the ability of the CPS to measure physical activities wirelessly in a restriction-free environment. This shows potential for gait analysis applications, physiotherapy, sports, fall prediction and prevention, and many others. The proposed solution also provides an alternative to the current high-cost and non-mobile solutions that have existed for decades. At a fraction of the cost of the commonly-used gait measurement method in healthcare organizations, the in-house designed and implemented CPS proves that it can reliably provide most of this information with high accuracy.

Studies in healthcare technologies and mHealth are moving towards facilitating remote patient monitoring [23], [128], [129], which can reduce costs and improve quality of care [130]. One of the objectives of this thesis is to provide a reliable sensor network framework that provides the necessary information in a digital twin city where patients can live their normal lives and have their activities monitored by their caretakers and physicians. The developed system performs gait analysis by transmitting compressed and encoded sensory information wirelessly to a cloud server, which in turn, performs historical data collection, as well as the required analysis to extract gait parameters. In addition, this solution provides an alternative to the current high-cost and non-mobile solutions that have existed for decades. The SmartInsoles, as health data collecting devices, can work by connecting them to a nearby smartphone device via Bluetooth. This opens the door for future experiments where people would wear these insoles and go on with their daily lives. The system would then be able to collect more realistic data that can be uploaded into a cloud database that physicians can access to monitor and analyze gait trends of their patients continuously without any location restrictions.

6.2 Framework Enhancement and Customization

One of the main advantages of the proposed framework in this thesis is its modularity which enables enhancements in future designs of similar MCPS. For example, the proof of concept implementation shows the design of a smart insole with 12 force sensors and an IMU. However, the modularity of the framework, is generic in such a way that any types and numbers of sensors following the same embedded software approach would work seamlessly with this CPS. As long as a PSP is generated, and as long as the communication protocol is followed, the cyber system will recognize a new physical system and will be able to read data from it and visualize it in real-time.

The way this framework works also gives independence to the design of the physical system in terms of the sampling rate. Each physical system is assigned a unique port on the cyber side, which gives flexibility to the physical system to stream data at its own pace without interrupting the overall CPS. The communication protocol's timeout threshold can be adjusted to account for slower physical systems as well.

The number of users in the CPS and the performance limitations of the overall system depend on several factors. For example, using IP addresses to uniquely identify users means that the system can, theoretically, handle as many users as the number of available IP addresses. However, each physical systems is also assigned a unique port number in the cyber system, and the performance of the cyber system highly depends on the number of streams it can handle from different ports. In addition, the frequency of received packets from each port is an important factor in determining how complex the processing can be. On the physical system's side, the sampling rate depend on the capabilities of the used microcontroller in terms of speed and communication abilities. The microcontroller speed can affect the sampling speed of the Sensors Handling Module, and the communication speed can affect the streaming speed to the cyber system.

6.3 Future Work

Future directions include adding feedback handling in the framework to allow for a well-structured protocol and two-directional communication between the cyber and the physical systems. Literature applications require several types of feedback including haptic, audible, and visual. These types of feedbacks should be taken into consideration when designing the upgraded framework and communication protocol since each type of feedback has specific networking, encoding, and compression requirements.

The current proposed framework performs real-time visualization using the PSVPs. These presets require prior visual construction that can work with specific types of sensors or health devices. The PSVP module works with the PSP module in order to match the sensor (or health device) to the required visualization preset based on the type of sensor (or device). However, if new type of sensor is introduced to the system, the PSVP will not be able to recognize it. Therefore, this module can be improved to adapt an intelligent algorithm for auto-construction to enable visualization of newly added, unrecognized, sensors and devices.

In regards to the implemented CPS, there remains some processes that are performed manually. Such processes include finding the heel-contact and toe-off timestamps from each footstep performed on the SmartInsole. Therefore, machine learning algorithms can be used to automate this process and enable auto-classification of gait phases. Automating this analysis can enhance research in fall detection, prediction, and prevention. Furthermore, the Tekscan Strideway system used for the validation study can calculate toe-in and toe-out angles in comparison to the subject's line of progression. These data provide information on how the subject's feet are tilted inwards or outwards, which has a direct effect on the subject's knee adduction according to [131]. For future work, we plan to calculate the toe-in and toe-off angles by using the information collected from the IMU sensor, which is already embedded in each SmartInsole.

In our study, we evaluated young healthy subjects only. Future studies should include patients with conditions that can affect their gait such as diabetes [132], Parkinson's [133] or patients with Multiple Sclerosis (MS) disease [134]. Several studies have also confirmed a relationship between gait variability and age progression [135]–[137]. With this being a pilot study, our inclusion criteria consisted solely of young adults. For future studies, data from older individuals acquired with the wearable SmartInsoles and the Strideway gait mat should be evaluated, as there is typically more variability in collected data from older adults.

Therefore, collecting gait data for patients or long periods and away from the restrictions of a laboratory setting can provide rich information that can help physicians monitor the development of symptoms and the progression of disease for such cases.

6.4 Chapter Summary

Chapter 6 concludes this thesis by providing a summary of the overall thesis, a brief discussion, and conclusions. Different directions for future work are discussed that have potential in the future of the physical activities measurement field.

APPENDICES

Appendix A Physical System Control Flowchart

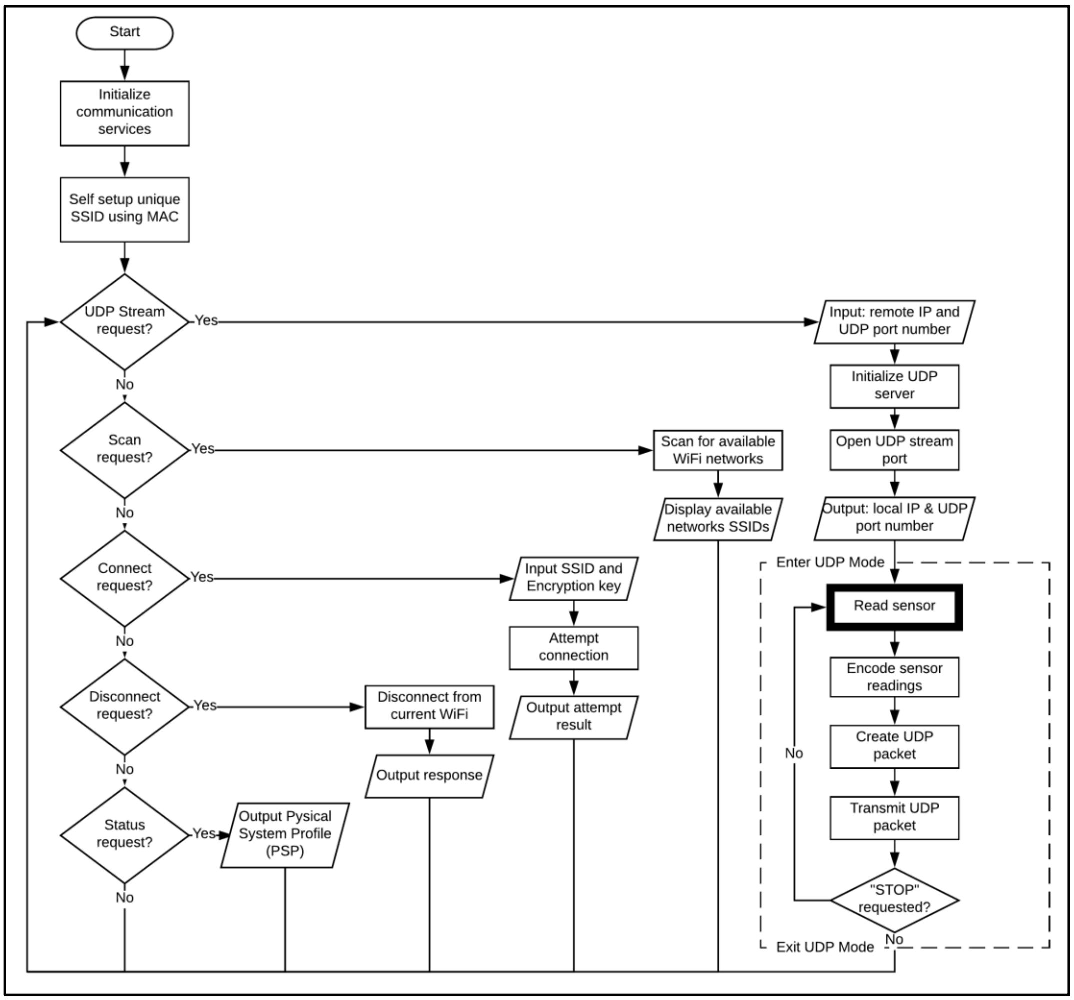

Figure 38: Physical System control flowchart

Appendix B DH Commands Flowcharts

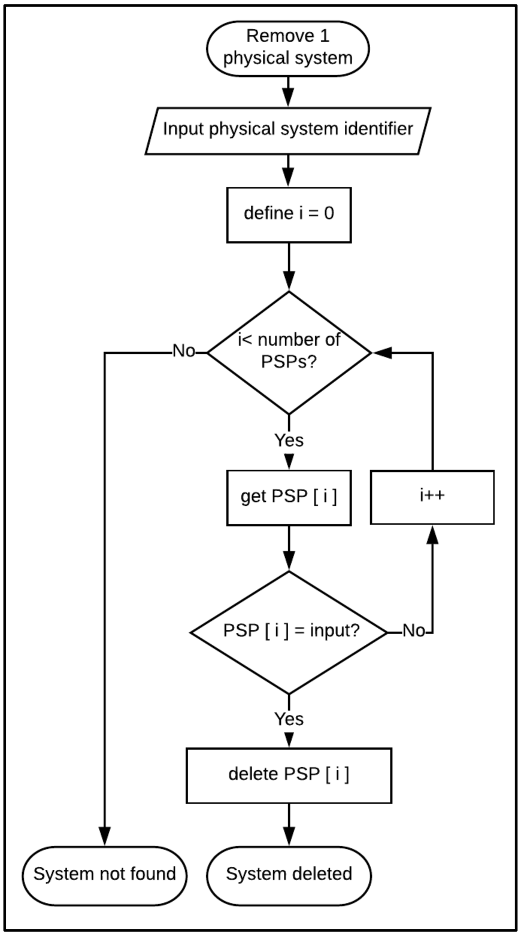

Figure 39: Flowchart for command: remove

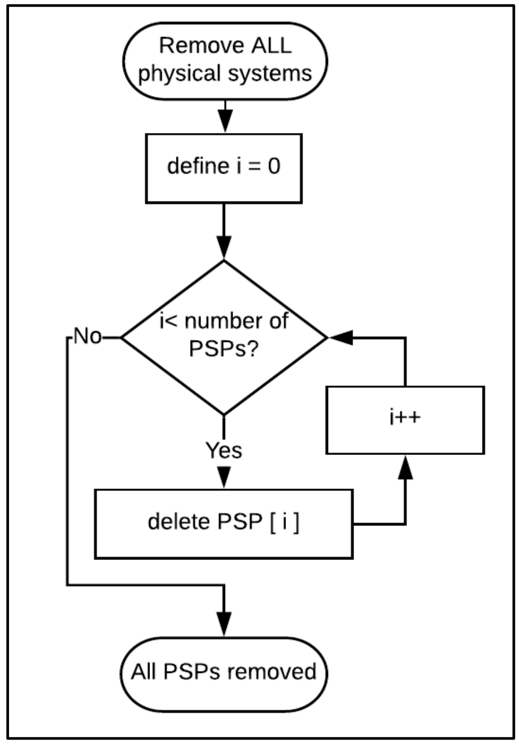

Figure 40: Flowchart for command: removeAll

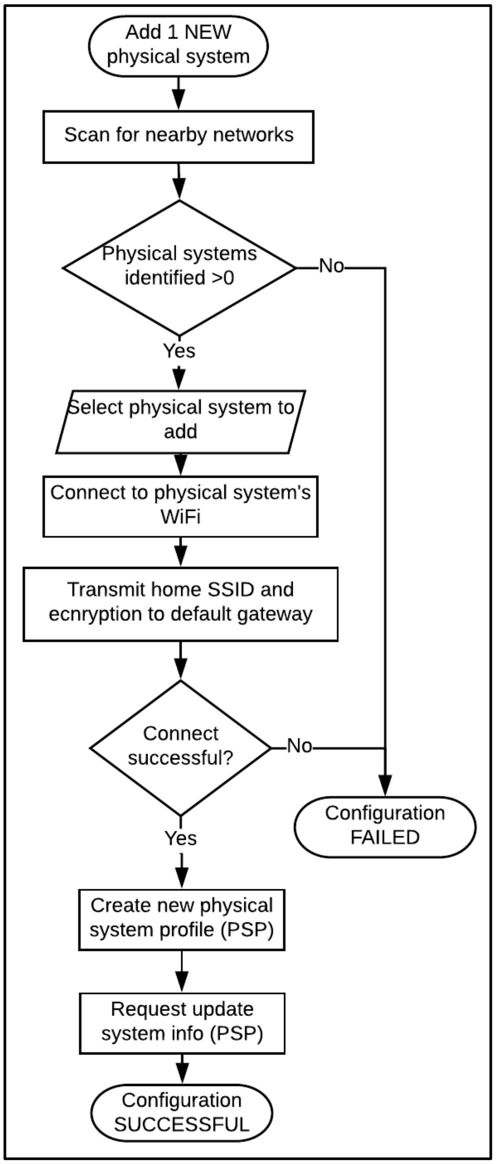

Figure 41: Flowchart for command: add

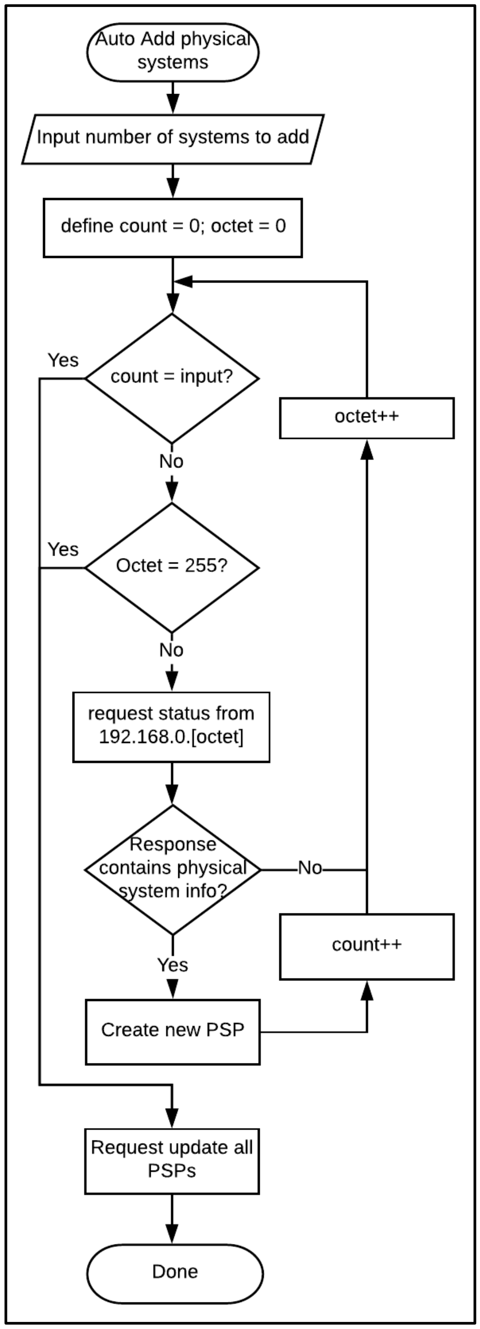

Figure 42: Flowchart for command: addAll

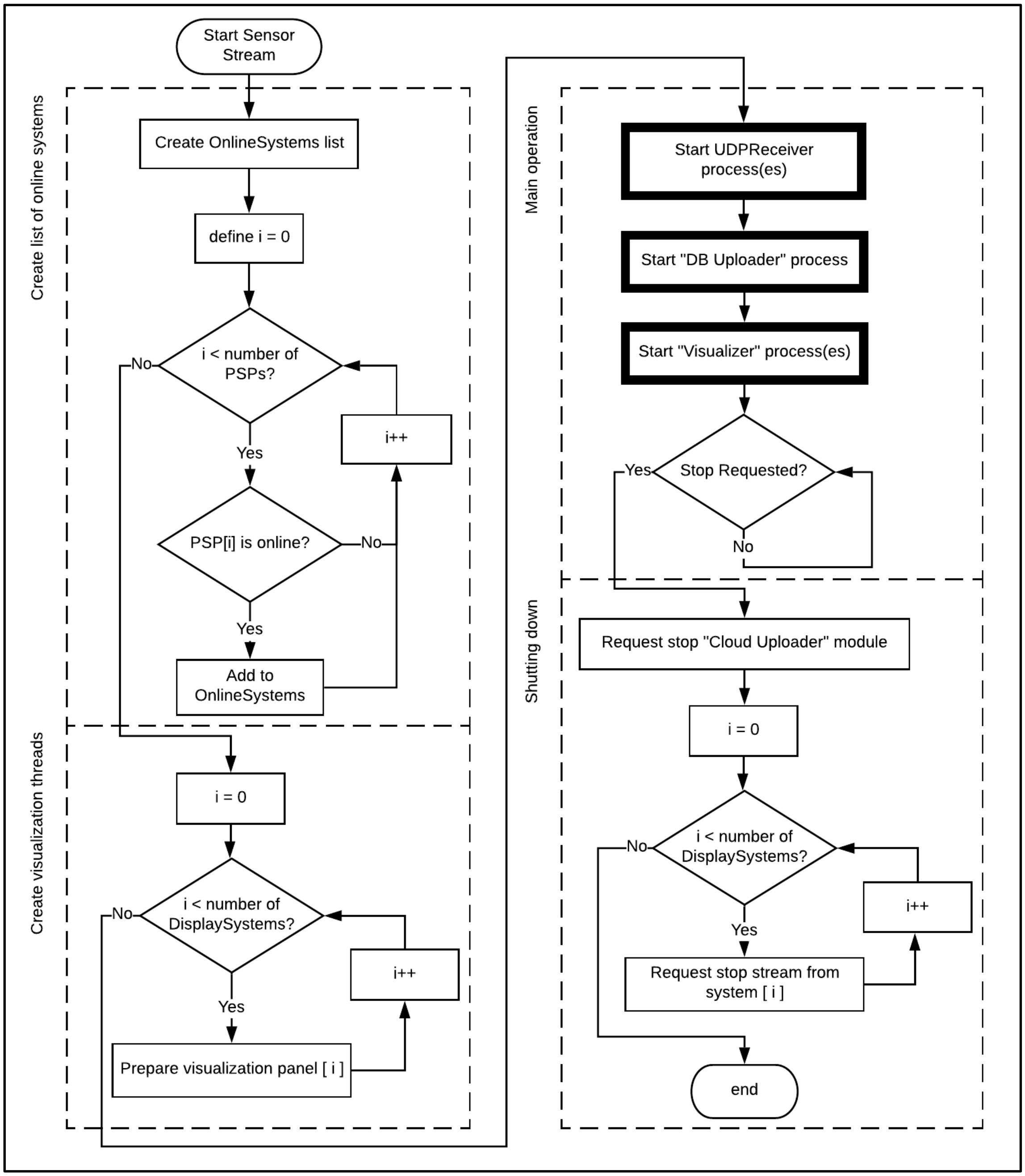

Figure 43: Flowchart for command: start

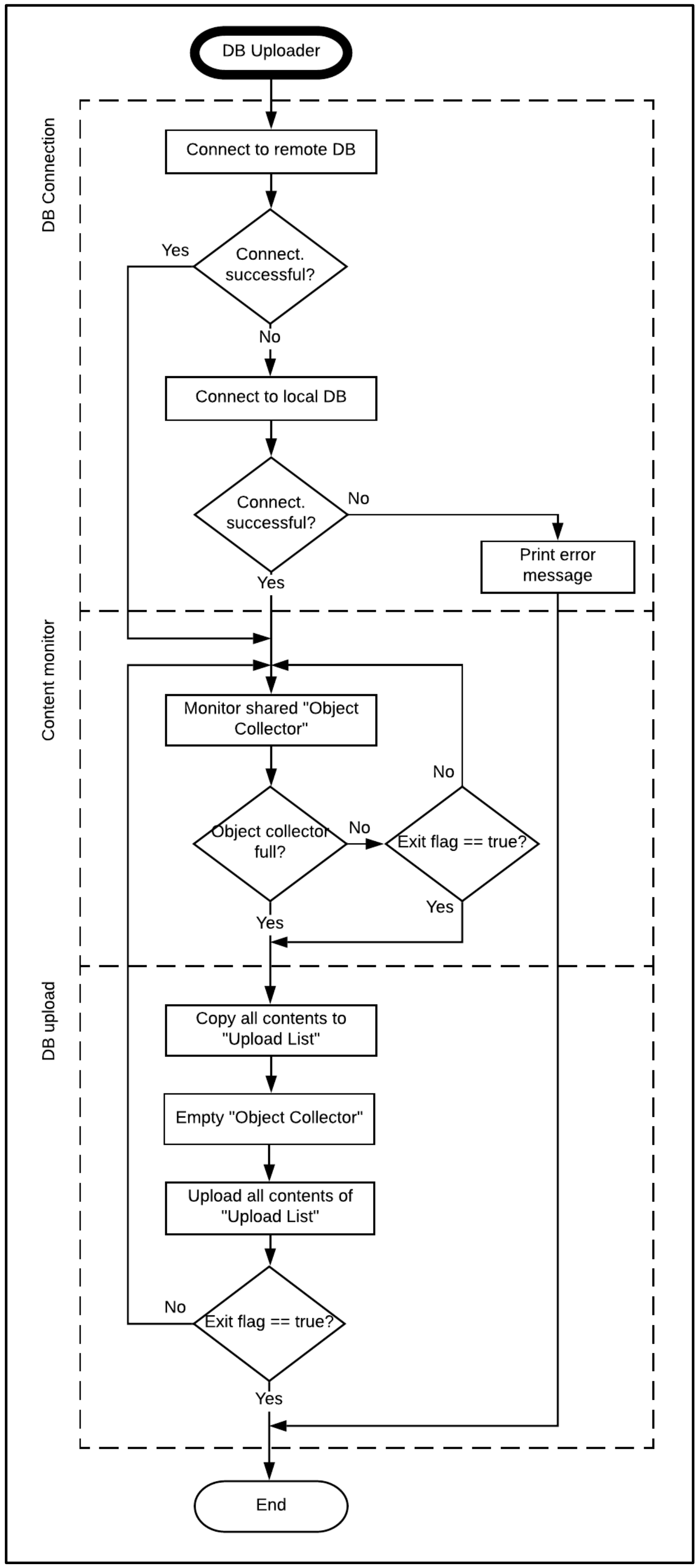

Figure 44: Flowchart for internal process running when "start" command is requested

References

[1] A. El Saddik, “Digital Twins: The Convergence of Multimedia Technologies,” *IEEE Multimed.*, vol. 25, no. 2, pp. 87–92, Apr. 2018.

[2] Q. Wu, K. Sum, and D. Nathan-Roberts, “How Fitness Trackers Facilitate Health Behavior Change,” *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 60, no. 1, pp. 1068–1072, Sep. 2016.

[3] R. Yang, E. Shin, M. W. Newman, and M. S. Ackerman, “When fitness trackers don’t ‘fit,’” in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing - UbiComp ’15*, 2015, pp. 623–634.

[4] J. Casselman, N. Onopa, and L. Khansa, “Wearable healthcare: Lessons from the past and a peek into the future,” *Telemat. Informatics*, vol. 34, no. 7, pp. 1011–1023, Nov. 2017.

[5] C. a. Rothkopf and J. B. Pelz, “Head movement estimation for wearable eye tracker,” in *Proceedings of the Eye tracking research & applications symposium on Eye tracking research & applications - ETRA’2004*, 2004, pp. 123–130.

[6] S. O’Regan and W. Marnane, “Multimodal detection of head-movement artefacts in EEG,” *J. Neurosci. Methods*, vol. 218, no. 1, pp. 110–120, Aug. 2013.

[7] A. Al-Rahayfeh and M. Faezipour, “Eye Tracking and Head Movement Detection: A State-of-Art Survey,” *IEEE J. Transl. Eng. Heal. Med.*, vol. 1, no. May, pp. 2100212–2100212, 2013.

[8] M. Haghi, K. Thurow, and R. Stoll, “Wearable Devices in Medical Internet of Things: Scientific Research and Commercially Available Devices,” *Healthc. Inform. Res.*, vol. 23, no. 1, p. 4, 2017.

[9] H. Jin, Z. Yang, S. Kumar, and J. I. Hong, “Towards Wearable Everyday Body-Frame Tracking using Passive RFIDs,” *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 1, no. 4, pp. 1–23, Jan. 2018.

[10] M. Geryes, J. Charara, A. Skaiky, A. Mcheick, and J.-M. Girault, “A novel biomedical application for the Myo gesture control armband,” in *2017 29th International Conference*

on Microelectronics (ICM), 2017, pp. 1–4.

[11] J. Kuschan, H. Schmidt, and J. Krueger, "Improved Ergonomics via an Intelligent Movement and Gesture Detection Jacket," in *Proceedings of ISR 2016: 47st International Symposium on Robotics*, 2016, pp. 373–378.

[12] H. A. Javaid, N. Rashid, M. I. Tiwana, and M. W. Anwar, "Comparative Analysis of EMG Signal Features in Time-domain and Frequency-domain using MYO Gesture Control," in *Proceedings of the 4th International Conference on Mechatronics and Robotics Engineering - ICMRE*, 2018, pp. 157–162.

[13] N. P. Brillantes, H. Kim, R. Feria, M. R. Solamo, and L. L. Figueroa, "Evaluation of a 3D physics classroom with Myo gesture control armband and unity," in *8th International Conference on Information, Intelligence, Systems & Applications (IISA)*, 2017, pp. 1–6.

[14] A. Alamri, M. Eid, R. Iglesias, S. Shirmohammadi, and A. El Saddik, "Haptic Virtual Rehabilitation Exercises for Poststroke Diagnosis," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 9, pp. 1876–1884, Sep. 2008.

[15] A. Alamri, J. Cha, and A. El Saddik, "AR-REHAB: An Augmented Reality Framework for Poststroke-Patient Rehabilitation," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 10, pp. 2554–2563, Oct. 2010.

[16] Y. Xin, X. Li, H. Tian, C. Guo, C. Qian, S. Wang, and C. Wang, "Shoes-equipped piezoelectric transducer for energy harvesting: A brief review," *Ferroelectrics*, vol. 493, no. 1, pp. 12–24, Mar. 2016.

[17] D. Giovanelli and E. Farella, "Force Sensing Resistor and Evaluation of Technology for Wearable Body Pressure Sensing," *J. Sensors*, pp. 1–13, 2016.

[18] F. Arafsha, F. Laamarti, and A. El Saddik, "Development of a wireless CPS for gait parameters measurement and analysis," in *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2018, pp. 1–5.

[19] A. Perrier, N. Vuillerme, V. Luboz, M. Bucki, F. Cannard, B. Diot, D. Colin, D. Rin, J.-P. Bourg, and Y. Payan, "Smart Diabetic Socks: Embedded device for diabetic foot prevention," *IRBM*, vol. 35, no. 2, pp. 72–76, Apr. 2014.

[20] S. K. Khaitan and J. D. McCalley, "Design Techniques and Applications of Cyberphysical Systems: A Survey," *IEEE Syst. J.*, vol. 9, no. 2, pp. 350–365, Jun. 2015.

[21] L. Gu, D. Zeng, S. Guo, A. Barnawi, and Y. Xiang, "Cost Efficient Resource Management in Fog Computing Supported Medical Cyber-Physical System," *IEEE Trans. Emerg. Top. Comput.*, vol. 5, no. 1, pp. 108–119, Jan. 2017.

[22] L. Wang, M. Törngren, and M. Onori, "Current status and advancement of cyber-physical systems in manufacturing," *J. Manuf. Syst.*, vol. 37, pp. 517–527, Oct. 2015.

[23] H. Alemdar and C. Ersoy, "Wireless sensor networks for healthcare: A survey," *Comput. Networks*, vol. 54, no. 15, pp. 2688–2710, 2010.

[24] M. H. Cintuglu, O. A. Mohammed, K. Akkaya, and A. S. Uluagac, "A Survey on Smart Grid Cyber-Physical System Testbeds," *IEEE Commun. Surv. Tutorials*, vol. 19, no. 1, pp. 446–464, 2017.

[25] D. Jia, K. Lu, J. Wang, X. Zhang, and X. Shen, "A Survey on Platoon-Based Vehicular Cyber-Physical Systems," *IEEE Commun. Surv. Tutorials*, vol. 18, no. 1, pp. 263–284, 2016.

[26] A. Muro-de-la-Herran, B. García-Zapirain, and A. Méndez-Zorrilla, "Gait Analysis Methods: An Overview of Wearable and Non-Wearable Systems, Highlighting Clinical Applications," *Sensors*, vol. 14, no. 2, pp. 3362–3394, Feb. 2014.

[27] O. Kocabas, T. Soyata, and M. K. Aktas, "Emerging Security Mechanisms for Medical Cyber Physical Systems," *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 13, no. 3, pp. 401–416, May 2016.

[28] Insup Lee, O. Sokolsky, Sanjian Chen, J. Hatcliff, Eunkyoung Jee, BaekGyu Kim, A. King, M. Mullen-Fortino, Soojin Park, A. Roederer, and K. K. Venkatasubramanian, "Challenges and Research Directions in Medical Cyber–Physical Systems," *Proc. IEEE*, vol. 100, no. 1, pp. 75–90, Jan. 2012.

[29] S. A. Haque, S. M. Aziz, and M. Rahman, "Review of Cyber-Physical System in Healthcare," *Int. J. Distrib. Sens. Networks*, vol. 10, no. 4, p. 217415, Apr. 2014.

[30] P. Leitão, A. W. Colombo, and S. Karnouskos, "Industrial automation based on cyber-

physical systems technologies: Prototype implementations and challenges," *Comput. Ind.*, vol. 81, pp. 11–25, Sep. 2016.

[31] R. F. Babiceanu and R. Seker, "Big Data and virtualization for manufacturing cyber-physical systems: A survey of the current status and future outlook," *Comput. Ind.*, vol. 81, pp. 128–137, Sep. 2016.

[32] S. Soegijoko, "A brief review on existing cyber-physical systems for healthcare applications and their prospective national developments," in *2013 3rd International Conference on Instrumentation, Communications, Information Technology and Biomedical Engineering (ICICI-BME)*, 2013, pp. 2–2.

[33] G. S. Avrunin, L. A. Clarke, L. J. Osterweil, J. M. Goldman, and T. Rausch, "Smart checklists for human-intensive medical systems," *Proc. Int. Conf. Dependable Syst. Networks*, 2012.

[34] S. A. Haque and S. M. Aziz, "False Alarm Detection in Cyber-physical Systems for Healthcare Applications," *AASRI Procedia*, vol. 5, pp. 54–61, 2013.

[35] E. K. Kolodner, S. Tal, D. Kyriazis, D. Naor, M. Allalouf, L. Bonelli, P. Brand, A. Eckert, E. Elmroth, S. V. Gogouvitis, D. Harnik, F. Hernandez, M. C. Jaeger, E. B. Lakew, J. M. Lopez, M. Lorenz, A. Messina, A. Shulman-Peleg, R. Talyansky, A. Voulodimos, and Y. Wolfsthal, "A Cloud Environment for Data-intensive Storage Services," in *2011 IEEE Third International Conference on Cloud Computing Technology and Science*, 2011, pp. 357–366.

[36] S. Wang, J. Min, and B. K. Yi, "Location Based Services for Mobiles: Technologies and Standards." LG Electronics Mobile Research, USA, 2008.

[37] C. G. Cassandras, "Online Control and Optimization for Cyber-Physical Systems," in *Cyber-Physical Systems*, Elsevier, 2017, pp. 31–54.

[38] P. M. N. Martins and J. A. McCann, "Network-Wide Programming Challenges in Cyber-Physical Systems," in *Cyber-Physical Systems*, Elsevier, 2017, pp. 103–113.

[39] E. Jovanov, A. Milenkovic, C. Otto, and P. C. de Groen, "A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation.," *J. Neuroeng. Rehabil.*, vol. 2, no. 1, p. 6, Mar. 2005.

[40] P. Bonato, “Advances in wearable technology and applications in physical medicine and rehabilitation,” *J. Neuroeng. Rehabil.*, vol. 2, pp. 2–5, 2005.

[41] G. Chen, C. Patten, D. H. Kothari, and F. E. Zajac, “Gait differences between individuals with post-stroke hemiparesis and non-disabled controls at matched speeds,” *Gait Posture*, vol. 22, no. 1, pp. 51–56, Aug. 2005.

[42] P. Lopez-Meyer, G. D. Fulk, and E. S. Sazonov, “Automatic Detection of Temporal Gait Parameters in Poststroke Individuals,” *IEEE Trans. Inf. Technol. Biomed.*, vol. 15, no. 4, pp. 594–601, Jul. 2011.

[43] E. S. Sazonov, G. Fulk, N. Sazonova, and S. Schuckers, “Automatic Recognition of postures and activities in stroke patients,” in *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2009, pp. 2200–2203.

[44] B. Yang, H. Dong, and A. El Saddik, “Development of a Self-Calibrated Motion Capture System by Nonlinear Trilateration of Multiple Kinects v2,” *IEEE Sens. J.*, vol. 17, no. 8, pp. 2481–2491, Apr. 2017.

[45] A. M. Tan, F. K. Fuss, Y. Weizman, and O. Troynikov, “Development of a smart insole for medical and sports purposes,” *Procedia Eng.*, vol. 112, pp. 152–156, 2015.

[46] B. Hafidh, H. Al Osman, and A. El Saddik, “SmartInsole: A foot-based activity and gait measurement device,” in *2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, 2013, pp. 1–4.

[47] A. M. Tan, F. K. Fuss, Y. Weizman, Y. Woudstra, and O. Troynikov, “Design of Low Cost Smart Insole for Real Time Measurement of Plantar Pressure,” *Procedia Technol.*, vol. 20, no. May 2016, pp. 117–122, 2015.

[48] L. Buthe, U. Blanke, H. Capkevics, and G. Troster, “A wearable sensing system for timing analysis in tennis,” in *2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, 2016, pp. 43–48.

[49] A. M. Tan, F. K. Fuss, Y. Weizman, and M. F. Azari, “Centre of Pressure Detection and Analysis with a High-resolution and Low-cost Smart Insole,” *Procedia Eng.*, vol. 112, pp. 146–151, 2015.

[50] S. Minto, D. Zanotto, E. M. Boggs, G. Rosati, and S. K. Agrawal, "Validation of a Footwear-Based Gait Analysis System With Action-Related Feedback," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 9, pp. 971–980, Sep. 2016.

[51] A. Godfrey, S. Del Din, G. Barry, J. C. Mathers, and L. Rochester, "Instrumenting gait with an accelerometer: A system and algorithm examination," *Med. Eng. Phys.*, vol. 37, no. 4, pp. 400–407, Apr. 2015.

[52] S. Crea, M. Donati, S. De Rossi, C. Oddo, and N. Vitiello, "A Wireless Flexible Sensorized Insole for Gait Analysis," *Sensors*, vol. 14, no. 1, pp. 1073–1093, Jan. 2014.

[53] F. A. O. Mota, V. H. M. Biajo, H. O. Mota, and F. H. Vasconcelos, "A wireless sensor network for the biomechanical analysis of the gait," in *2017 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2017, pp. 1–6.

[54] S. R. Edgar, T. Swyka, G. Fulk, and E. S. Sazonov, "Wearable shoe-based device for rehabilitation of stroke patients," *2010 Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBC'10*, no. November 2016, pp. 3772–3775, 2010.

[55] F. Arafsha, C. Hanna, A. Aboualmagd, S. Fraser, and A. El Saddik, "Instrumented Wireless SmartInsole System for Mobile Gait Analysis: A Validation Pilot Study with Tekscan Strideway," *J. Sens. Actuator Networks*, vol. 7, no. 3, p. 36, 2018.

[56] H. Rouhani, J. Favre, X. Crevoisier, and K. Aminian, "A wearable system for multi-segment foot kinetics measurement," *J. Biomech.*, vol. 47, no. 7, pp. 1704–1711, May 2014.

[57] B. J. Braun, N. T. Veith, R. Hell, S. Döbele, M. Roland, M. Rollmann, J. Holstein, and T. Pohlemann, "Validation and reliability testing of a new, fully integrated gait analysis insole," *J. Foot Ankle Res.*, vol. 8, no. 1, p. 54, Dec. 2015.

[58] S. Del Din, A. Godfrey, and L. Rochester, "Validation of an Accelerometer to Quantify a Comprehensive Battery of Gait Characteristics in Healthy Older Adults and Parkinson's Disease: Toward Clinical and at Home Use," *IEEE J. Biomed. Heal. Informatics*, vol. 20, no. 3, pp. 838–847, May 2016.

[59] H. Jagos, K. Pils, M. Haller, C. Wassermann, C. Chhatwal, D. Rafolt, and F. Rattay, "Mobile gait analysis via eSHOEs instrumented shoe insoles : a pilot study for validation

against the gold standard GAITRite," *J. Med. Eng. Technol.*, vol. 41, no. 5, pp. 375–386, 2017.

[60] G. D. Clifford, W. J. Long, G. B. Moody, and P. Szolovits, "Robust parameter extraction for decision support using multimodal intensive care data.," *Philos. Trans. A. Math. Phys. Eng. Sci.*, vol. 367, no. 1887, pp. 411–29, 2009.

[61] A. Coda, T. Carline, and D. Santos, "Repeatability and reproducibility of the Tekscan HR-Walkway system in healthy children," *Foot*, vol. 24, no. 2, pp. 49–55, Jun. 2014.

[62] C. Giacomozzi, "Hardware performance assessment recommendations and tools for baropodometric sensor systems," *Ann. Ist. Super. Sanita*, vol. 46, no. 2, pp. 158–167, 2010.

[63] E. Sazonov, G. Fulk, J. Hill, Y. Schutz, and R. Browning, "Monitoring of Posture Allocations and Activities by a Shoe-Based Wearable Sensor," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 4, 2011.

[64] O. A. Postolache and G. B. Postolache, "Development and selection of balance sensing devices," *IEEE Instrum. Meas. Mag.*, vol. 20, no. 1, pp. 38–48, Feb. 2017.

[65] A. M. Howell, T. Kobayashi, H. A. Hayes, K. B. Foreman, and S. J. M. Bamberg, "Kinetic gait analysis using a low-cost insole," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 12, pp. 3284–3290, 2013.

[66] L. Motha, J. Kim, and W. S. Kim, "Instrumented rubber insole for plantar pressure sensing," *Org. Electron. physics, Mater. Appl.*, vol. 23, pp. 82–86, 2015.

[67] T. Kaya, B. Shiari, K. Petsch, and D. Yates, "Design of a MEMS Capacitive Comb-drive Accelerometer," *Boston*, 2011.

[68] Y. Zhuang, J. Gong, D. C. Kerrigan, B. C. Bennett, J. Lach, and S. Russell, "Gait tracker shoe for accurate step-by-step determination of gait parameters," in *2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, 2016, pp. 13–18.

[69] B. Mariani, C. Hoskovec, S. Rochat, C. Büla, J. Penders, and K. Aminian, "3D gait assessment in young and elderly subjects using foot-worn inertial sensors," *J. Biomech.*,

vol. 43, no. 15, pp. 2999–3006, Nov. 2010.

[70] Y. Weizman and F. K. Fuss, “Development of Instrumented Soccer Footwear for Kicking Analysis and Training Purposes,” *Procedia Eng.*, vol. 112, pp. 157–162, 2015.

[71] A. Lees, T. Asai, T. B. Andersen, H. Nunome, and T. Sterzing, “The biomechanics of kicking in soccer: A review,” *J. Sports Sci.*, vol. 28, no. 8, pp. 805–817, Jun. 2010.

[72] H. Rouhani, J. Favre, X. Crevoisier, and K. Aminian, “Ambulatory assessment of 3D ground reaction force using plantar pressure distribution,” *Gait Posture*, vol. 32, no. 3, pp. 311–316, Jul. 2010.

[73] G. Mastorakis and D. Makris, “Fall detection system using Kinect’s infrared sensor,” *J. Real-Time Image Process.*, vol. 9, no. 4, pp. 635–646, Dec. 2014.

[74] D. Triantafyllou, S. Krinidis, D. Ioannidis, I. N. Metaxa, C. Ziazios, and D. Tzovaras, “A Real-time Fall Detection System for Maintenance Activities in Indoor Environments,” *IFAC-PapersOnLine*, vol. 49, no. 28, pp. 286–290, 2016.

[75] D. Arney, S. Fischmeister, J. M. Goldman, I. Lee, and R. Trausmuth, “Plug-and-play for medical devices: Experiences from a case study,” *Biomed. Instrum. Technol.*, vol. 43, no. 4, pp. 313–317, 2009.

[76] M. Imhoff and R. Fried, “The crying wolf: Still crying?,” *Anesth. Analg.*, vol. 108, no. 5, pp. 1382–1383, 2009.

[77] Clinical Alarms Task Force, “Impact of clinical alarms on patient safety: a report from the American College of Clinical Engineering Healthcare Technology Foundation,” *Journal of Clinical Engineering*. 2007.

[78] J. Edworthy and E. Hellier, “Alarms and human behaviour: Implications for medical alarms,” *Br. J. Anaesth.*, vol. 97, no. 1, pp. 12–17, 2006.

[79] T. K. Nuckols, A. G. Bower, S. M. Paddock, L. H. Hilborne, P. Wallace, J. M. Rothschild, A. Griffin, R. J. Fairbanks, B. Carlson, R. J. Panzer, and R. H. Brook, “Programmable infusion pumps in ICUS: An analysis of corresponding adverse drug events,” *J. Gen. Intern. Med.*, vol. 23, no. 1 SUPPL., pp. 41–45, 2008.

[80] J. M. Feldman, M. H. Ebrahim, and I. Bar-Kana, “Robust sensor fusion improves heart

rate estimation: Clinical evaluation," *J. Clin. Monit.*, vol. 13, no. 6, pp. 379–384, 1997.

[81] L. A. Tang, X. Yu, S. Kim, J. Han, C. C. Hung, and W. C. Peng, "Tru-alarm: Trustworthiness analysis of sensor networks in cyber-physical systems," *Proc. - IEEE Int. Conf. Data Mining, ICDM*, pp. 1079–1084, 2010.

[82] S. Bamberg, a. Y. Benbasat, D. M. Scarborough, D. E. Krebs, and J. a. Paradiso, "Gait Analysis Using a Shoe-Integrated Wireless Sensor System," *IEEE Trans. Inf. Technol. Biomed.*, vol. 12, no. 4, pp. 413–423, 2008.

[83] A. M. Howell, "Insole-based gait analysis," 2012.

[84] M. Benocci, L. Rocchi, E. Farella, L. Chiari, and L. Benini, "A wireless system for gait and posture analysis based on pressure insoles and Inertial Measurement Units," in *Proceedings of the 3d International ICST Conference on Pervasive Computing Technologies for Healthcare*, 2009, vol. 13, no. 2, pp. 1–6.

[85] D. Antolín, N. Medrano, B. Calvo, and F. Pérez, "A Wearable Wireless Sensor Network for Indoor Smart Environment Monitoring in Safety Applications," *Sensors*, vol. 17, no. 2, p. 365, Feb. 2017.

[86] "Connect multiple Bluetooth devices to one computer." [Online]. Available: support.apple.com/HT201171. [Accessed: 13-Nov-2017].

[87] C. Gerlach, D. Krumm, M. Illing, J. Lange, O. Kanoun, S. Odenwald, and A. Hubler, "Printed MWCNT-PDMS-Composite Pressure Sensor System for Plantar Pressure Monitoring in Ulcer Prevention," *IEEE Sens. J.*, vol. 15, no. 7, pp. 3647–3656, 2015.

[88] "Interlink Electronics FSR Integration Guide." [Online]. Available: https://www.interlinkelectronics.com/request-integration-guides. [Accessed: 02-Feb-2018].

[89] Y. Zhang, M. Qiu, C. Tsai, M. M. Hassan, and A. Alamri, "Health-CPS: Healthcare Cyber-Physical System Assisted by Cloud and Big Data," *IEEE Syst. J.*, vol. 11, no. 1, pp. 88–95, Mar. 2017.

[90] Insup Lee, O. Sokolsky, S. Chen, J. Hatcliff, E. Jee, BaekGyu Kim, A. King, M. Mullen-Fortino, S. Park, A. Roederer, and K. K. Venkatasubramanian, "Challenges and Research

Directions in Medical Cyber-Physical Systems," *Proc. IEEE*, vol. 100, no. 1, pp. 75–90, Jan. 2012.

[91] H. Abid, L. T. T. Phuong, J. Wang, S. Lee, and S. Qaisar, "V-Cloud: Vehicular Cyber-Physical Systems and Cloud Computing," *Proc. 4th Int. Symp. Appl. Sci. Biomed. Commun. Technol. - ISABEL '11*, pp. 1–5, 2011.

[92] A. Mukherjee, H. S. Paul, S. Dey, and A. Banerjee, "ANGELS for distributed analytics in IoT," in *IEEE World Forum on Internet of Things (WF-IoT)*, 2014, pp. 565–570.

[93] H. Derhamy, J. Eliasson, J. Delsing, and P. Priller, "A survey of commercial frameworks for the Internet of Things," in *IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA)*, 2015, pp. 1–8.

[94] M. Ammar, G. Russello, and B. Crispo, "Internet of Things: A survey on the security of IoT frameworks," *J. Inf. Secur. Appl.*, vol. 38, pp. 8–27, Feb. 2018.

[95] F. Li, M. Vogler, M. Claessens, and S. Dustdar, "Towards Automated IoT Application Deployment by a Cloud-Based Approach," in *IEEE 6th International Conference on Service-Oriented Computing and Applications*, 2013, pp. 61–68.

[96] D. R. Smith, "Analog-to-Digital Conversion Techniques," in *Digital Transmission Systems*, 3rd ed., Springer Science+Business Media, 2003, pp. 84–176.

[97] T. Srisooksai, K. Keamarungsi, P. Lamsrichan, and K. Araki, "Practical data compression in wireless sensor networks: A survey," *J. Netw. Comput. Appl.*, vol. 35, no. 1, pp. 37–59, Jan. 2012.

[98] J. P. Walsh, "Apparatus for providing dynamic selection of modem protocol to support multiple modem types," US5202899 A, 1993.

[99] Q. Zhicong, L. Delin, and W. Shunxiang, "Analysis and Design of A Mobile Forensic Software System Based on AT Commands," in *2008 IEEE International Symposium on Knowledge Acquisition and Modeling Workshop*, 2008, pp. 597–600.

[100] "The Open Geospatial Consortium." [Online]. Available: http://www.opengeospatial.org/. [Accessed: 31-Aug-2018].

[101] M. Botts and A. Robin, "OGC® SensorML: Model and XML Encoding Standard." 2014.

[102] C. R. Nave, "Center of Mass," *Department of Physics and Astronomy, Georgia State University*. [Online]. Available: http://hyperphysics.phy-astr.gsu.edu/hbase/cm.html. [Accessed: 12-Feb-2018].

[103] T. Bray, "The JavaScript Object Notation (JSON) Data Interchange Format," RFC Editor, Dec. 2017.

[104] D. De Venuto, V. F. Annese, M. Ruta, E. Di Sciascio, and A. L. Sangiovanni Vincentelli, "Designing a Cyber–Physical System for Fall Prevention by Cortico–Muscular Coupling Detection," *IEEE Des. Test*, vol. 33, no. 3, pp. 66–76, Jun. 2016.

[105] "Force Sensing Resistors An Overview of the Technology," *FSR Integr. Guid. Eval. Parts Cat.*

[106] A. Hollinger and M. M. Wanderley, "Evaluation of Commercial Force-Sensing Resistors," *Proc. Int. Conf. New Interfaces Music. Expr.*, pp. 4–8, 2006.

[107] "SparkFun IMU Breakout - MPU-9250." [Online]. Available: https://www.sparkfun.com/products/13762. [Accessed: 02-Feb-2018].

[108] "MPU-9250." [Online]. Available: https://www.invensense.com/products/motion-tracking/9-axis/mpu-9250/. [Accessed: 02-Feb-2018].

[109] "ESP8266." [Online]. Available: https://www.espressif.com/en/products/hardware/esp8266ex/overview. [Accessed: 02-Feb-2018].

[110] "Adafruit Feather HUZZAH." [Online]. Available: https://www.adafruit.com/product/2821. [Accessed: 02-Feb-2018].

[111] "Battery Life Calculator." [Online]. Available: https://www.digikey.ca/en/resources/conversion-calculators/conversion-calculator-battery-life. [Accessed: 02-Feb-2018].

[112] M. J. Hessert, M. Vyas, J. Leach, K. Hu, L. A. Lipsitz, and V. Novak, "Foot pressure distribution during walking in young and old adults," *BMC Geriatr.*, vol. 5, no. 1, p. 8, Dec. 2005.

[113] A. Höhne, C. Stark, and G. P. Brüggemann, "Plantar pressure distribution in gait is not

affected by targeted reduced plantar cutaneous sensation," *Clin. Biomech.*, vol. 24, no. 3, pp. 308–313, 2009.

[114] K. J. Mickle, B. J. Munro, S. R. Lord, H. B. Menz, and J. R. Steele, "Gait, balance and plantar pressures in older people with toe deformities," *Gait Posture*, vol. 34, no. 3, pp. 347–351, 2011.

[115] R. Periyasamy, A. Mishra, S. Anand, and A. C. Ammini, "Preliminary investigation of foot pressure distribution variation in men and women adults while standing," *Foot*, vol. 21, no. 3, pp. 142–148, 2011.

[116] D. Palossi, J. Singh, M. Magno, and L. Benini, "Target following on nano-scale Unmanned Aerial Vehicles," in *7th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI)*, 2017, vol. 1, pp. 170–175.

[117] F. T. Liu, Y. T. Wang, and H. P. Ma, "Gesture recognition with wearable 9-axis sensors," *IEEE Int. Conf. Commun.*, 2017.

[118] "MongoDB." [Online]. Available: https://www.mongodb.com/. [Accessed: 13-Nov-2017].

[119] H. Rouhani, J. Favre, X. Crevoisier, and K. Aminian, "Ambulatory measurement of ankle kinetics for clinical applications," *J. Biomech.*, vol. 44, no. 15, pp. 2712–2718, Oct. 2011.

[120] "Strideway System," *Tekscan*. [Online]. Available: https://www.tekscan.com/products-solutions/systems/strideway-system. [Accessed: 12-Feb-2018].

[121] G. V. Zammit, H. B. Menz, and S. E. Munteanu, "Reliability of the TekScan MatScan®system for the measurement of plantar forces and pressures during barefoot level walking in healthy adults," *J. Foot Ankle Res.*, vol. 3, no. 11, Dec. 2010.

[122] A. Brenton-Rule, J. Mattock, M. Carroll, N. Dalbeth, S. Bassett, H. B. Menz, and K. Rome, "Reliability of the TekScan MatScan® system for the measurement of postural stability in older people with rheumatoid arthritis," *J. Foot Ankle Res.*, vol. 5, no. 21, Dec. 2012.

[123] J. Goetschius, M. A. Feger, J. Hertel, and J. M. Hart, "Validating Center-of-Pressure Balance Measurements Using the MatScan® Pressure Mat," *J. Sport Rehabil.*, vol. 27, Jan. 2018.

[124] W. Zijlstra and A. L. Hof, “Assessment of spatio-temporal gait parameters from trunk accelerations during human walking,” *Gait Posture*, vol. 18, no. 2, pp. 1–10, Oct. 2003.

[125] E. B. Titianova, P. S. Mateev, and I. M. Tarkka, “Footprint analysis of gait using a pressure sensor system,” *J. Electromyogr. Kinesiol.*, vol. 14, no. 2, pp. 275–281, Apr. 2004.

[126] P. S. Myles and J. Cui, “I. Using the Bland–Altman method to measure agreement with repeated measures,” *Br. J. Anaesth.*, vol. 99, no. 3, pp. 309–311, Sep. 2007.

[127] T. Stöggl and A. Martiner, “Validation of Moticon’s OpenGo sensor insoles during gait, jumps, balance and cross-country skiing specific imitation movements,” *J. Sports Sci.*, vol. 35, no. 2, pp. 196–206, Jan. 2017.

[128] B. Yuan and J. Herbert, “Web-based real-time remote monitoring for pervasive healthcare,” in *2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2011, pp. 625–629.

[129] B. Latré, B. Braem, I. Moerman, C. Blondia, and P. Demeester, “A survey on wireless body area networks,” *Wirel. Networks*, vol. 17, no. 1, pp. 1–18, Jan. 2011.

[130] M. Landolina, G. B. Perego, M. Lunati, A. Curnis, G. Guenzati, A. Vicentini, G. Parati, G. Borghi, P. Zanaboni, S. Valsecchi, and M. Marzegalli, “Remote Monitoring Reduces Healthcare Use and Improves Quality of Care in Heart Failure Patients With Implantable Defibrillators: The Evolution of Management Strategies of Heart Failure Patients With Implantable Defibrillators (EVOLVO) Study,” *Circulation*, vol. 125, no. 24, pp. 2985–2992, Jun. 2012.

[131] S. S. Khan, S. J. Khan, and J. Usman, “Effects of toe-out and toe-in gait with varying walking speeds on knee joint mechanics and lower limb energetics,” *Gait Posture*, vol. 53, pp. 185–192, Mar. 2017.

[132] P. R. Cavanagh, J. A. Derr, J. S. Ulbrecht, R. E. Maser, and T. J. Orchard, “Problems with Gait and Posture in Neuropathic Patients with Insulin-Dependent Diabetes Mellitus,” *Diabet. Med.*, vol. 9, no. 5, pp. 469–474, Jun. 1992.

[133] J. D. Schaafsma, N. Giladi, Y. Balash, A. L. Bartels, T. Gurevich, and J. M. Hausdorff, “Gait dynamics in Parkinson’s disease: relationship to Parkinsonian features, falls and

response to levodopa," *J. Neurol. Sci.*, vol. 212, no. 1–2, pp. 47–53, Aug. 2003.

[134] C. L. Martin, B. A. Phillips, T. J. Kilpatrick, H. Butzkueven, N. Tubridy, E. McDonald, and M. P. Galea, "Gait and balance impairment in early multiple sclerosis in the absence of clinical disability," *Mult. Scler. J.*, vol. 12, no. 5, pp. 620–628, Sep. 2006.

[135] A. J. Campbell, J. Reinken, B. C. Allan, and G. S. Martinez, "Falls in old age: a study of frequency and related clinical factors," *Age Ageing*, vol. 10, pp. 264–270, 1981.

[136] A. Gabell and U. S. L. Nayak, "The Effect of Age on Variability in Gait," *J. Gerontol.*, vol. 39, no. 6, pp. 662–666, Nov. 1984.

[137] K. Jahn, A. Zwergal, and R. Schniepp, "Gait Disturbances in Old Age," *Dtsch. Aerzteblatt Online*, vol. 107, no. 17, p. 306–315; quiz 316, Apr. 2010.