# Gait Monitoring using Wireless Sensors

Final Report 18-748
Spring 2017

Iljoo Baek, Mengwen He and Emily Ruppel
Carnegie Mellon University
ibaek,mengwenhe,eruppel@andrew.cmu.edu

***Abstract*—In this project we demonstrate Safewalk, a gait monitoring system comprised of Firefly wireless sensor motes that relay data from inertial measurement units mounted on a subject's body to a central server. Using the acceleration and orientation information reported at six different points along a subject's legs and feet, we were able to visualize the subject's movements in real time and capture the relevant motion data to be used for future analysis. Our system also provides a low-processing solution for fall detection through thresholding analysis of the orientation measurements reported by a subset of the sensor nodes in the system. After testing our system on three different subjects, we believe it represents a proof of concept of low cost gait monitoring systems that will improve the quality of patient care in a multitude of settings.**

## I. Introduction

Gait patterns, the measurable characteristics of an individuals walking movement, can provide insight into a variety of components of an individuals health status. Certain gait pathologies may be associated with specific medical conditions. Sudden changes in the way an individual walks can be indicative of the occurrence of an undetected but deleterious medical event, i.e. undetected stroke. A shift in gait pattern over time may come about as the result of muscular or neural decline that could increase the probability of a dangerous event, such as a fall, in the future. However, diagnosing conditions or predicting an increased fall risk from gait patterns requires the attention of a trained medical professional, which limits the settings in which gait analysis can be used.

Wireless Body Area Networks have been proposed as a promising solution for capturing useful health status indicators without inconveniencing a patient [6], [5]. The additional data enables a new field of computer-assisted physical therapy that allows therapists to deliver more targeted care to their patients. Advanced treatment centers for patients with unique cases, such as the Walter Reed Center for Performance and Clinical Research (CPCR) amputee clinic, use extensive computer modeling to aid in patient rehabilitation [4]. An apparent barrier to the widespread adoption of computer assisted therapy is the cost of the motional analysis systems. For instance, the CPCR uses 23 infrared cameras to track the movement of reflective markers attached to a patient's body and six force plates in the floor to fully capture a patient's movement. Though the movement data can be captured and used at a later time, the gait analysis is still limited to a clinical setting.

Safewalk's goal was to expand the usefulness of gait analysis beyond a professionals office to an individuals day-to-day environment. We accomplished our goal by demonstrating a proof-of-concept gait monitoring system that relies on wireless motes instrumented with Inertial Measurement Units (IMUs) to capture relevant data and relay them to a powerful backend machine for offline and real time processing. Our system extracted a variety of components of the subject's gait pattern from linear and angular acceleration data measured at specfic points along his or her legs. The subject's movements were then visualized and features of the gait were plotted in real time. We also demonstrate the practicality of real-time fall detection by using a simple thresholding technique to indicate when a subject is no longer upright. Throughout the trials with our prototype system we note that wireless sensor networks are particularly promising for accomplishing gait detection because the small nodes and lack of constraining wires minimizes the impact on the patient and allows for an honest gait assessment without the residual impact of cumbersome measurement devices.

## II. Related Work

This project is based on prior work in gait analysis and wireless sensors networks used to aid healthcare objectives. In this section we will describe how the prior work influenced the design of our project

### A. Gait Analysis

Gait analysis using wearable sensors is a well-studied approach to characterizing gait patters [11]. The goal of the gait analysis approaches is to analyze the various phases of the human gait to identify abnormalities. A normal walking gait is comprised of eight stages as shown in Figure 1.

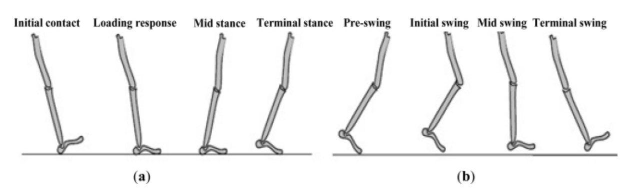



Fig. 1. **Phases of the stance (a) and swing (b) periods [11]**

Since the first study using wearable sensors in 1973, researchers have captured the data from these phases using a variety of transducers. Recently, multiple studies have used IMUs attached to a subjects thigh, calf, and foot to study gait kinematics. Gait kinematics investigates the movement of segments of the lower limbs while walking or running [7]. This is distinct from gait kinetics which analyzes the forces exerted on different joints while walking. The difficulty with kinetic analysis is the hardware required to capture accurate measurements. Previous studies have relied on specialized, instrumented footwear that measures the forces acting on a patients foot. Gait kinematics, on the other hand, has moved towards less intrusive measurement techniques. For instance, [8] used small wireless sensor nodes to report information about the gait characteristics of test subjects. Graurock, et. al went one step further in targeting patient convenience by building a "plug-and-play" system wherein each sensor node self calculated its position on a patient's lower extremities [3].

### *B. IMU Data Processing*

A large amount of work has been performed in processing the acceleration and orientation values reported by inertial measurement sensors to make use of the data. Specifically in terms of gait analysis, the work comes from both the robotics and motion capture communities. For instance, Yost labs has characterized the different joint types of the human body in terms of the necessary interpretation of data from IMUs mounted on adjacent limbs [1]. The complexity of the analysis ranges from the simple calculation to determine the angle of a hinge joint to the vector transformations required to assess the pitch, yaw and roll of multi-axis joints. Complementary work from the medical community puts the raw measurements in context for clinicians by providing a point of comparison. For instance, in [2] inertial sensor data was collected from a large population of adults and translated into a database of typical gait parameter values that researchers and healthcare professionals can use as a point of comparison. The study combines a multitude of well characterized algorithms for assessing gait parameters using only two IMUs, one on each of the subject's feet, and is able to build a model for normal gaits in terms of parameters such as foot clearance and step cadence.

## III. Technical Description

Our system follows in the spirit of prior work using IMUs to measure gait parameters, and work that uses wireless sensors to improve the quality of patient care. The system architecture we employ to achieve our goal of measuring a subject's gait using wireless sensors is shown in Figure 2. Note that it contains six Firefly slave devices each with a Sparkfun Razor 9 DoF Razor IMU M0 board attached. The data are then transmitted from the slave nodes to a master node which finally pushes the data to a server. To implement our gait monitoring system we had to handle three major design points- transmitting data from the IMUs, analyzing the IMU data, and building a useful interface to the information. This section will describe the choices our team made with regards to these three subsystems.

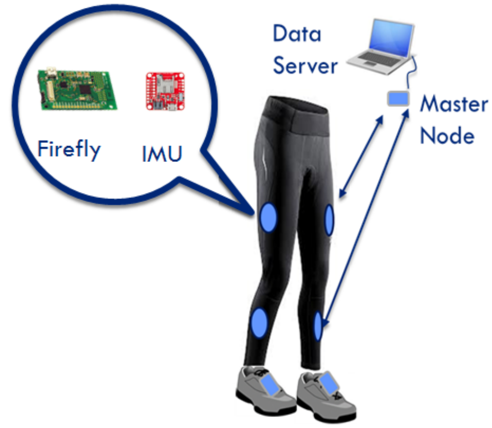



Fig. 2. **High level system design**

### *A. Device Communication*

Moving data from the IMU to the remote server can be separated into two distinct transitions– one from IMU to firefly slave device and another from the slave to the master node. As shown in Figure 3, the IMU board attaches directly to the Firefly. The connection provides power from the Firefly batteries to the IMU and allows for two way communication over one of the Firefly's auxiliary UART ports and the configurable IMU communication ports. The majority of the data transported over the serial connection is the IMU readings transported from the IMU board to the Firefly whenever the IMU has data available.

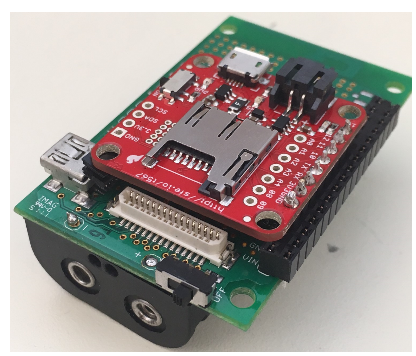


Fig. 3. **Firefly node with IMU board attached**

The format of the packet passed from IMU to Firefly is shown in Figure 4. The data are encoded as ASCII characters to simplify debugging as data move across the multiple platforms in our system. While the ASCII encoding greatly simplified debugging, it does mean that the packet size may change given different accelerometer values. Throughout the project, several unsuccessful attempts were made to tightly control the data transmission by the IMU board, but the nature of the UART protocol made timing discrepancies between the two devices difficult to overcome, and was further complicated by the variable packet size. Instead, we settled on a four character pattern to indicate the beginning of an

IMU transmission followed by a single special character to indicate the end. In contrast to the long packets transmitted from the IMU board, the firefly needs only to issue commands to the IMU, so the protocol from firefly to IMU is far simpler. If the IMU board receives a character from the Firefly, the corresponding command is retrieved from a look-up table and carried out. A final feature of the IMU-Firefly communication protocol is automatic reboot of the firefly.

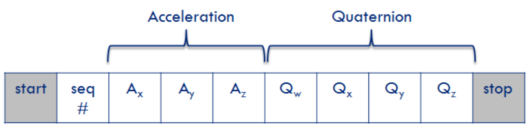



Fig. 4. **Packet Format from IMU to Firefly**

Once packets are passed from the IMU to the Firefly slave device, the firefly transmits the packet using the Point Coordination Function (PCF) TDMA protocol [10]. Each slave device is statically assigned a time slot according to its preprogrammed MAC address and only transmits during its assigned slot. The PCF TDMA protocol is designed for a network with a star topology, where all slave nodes are within one hop of the master device because all slave nodes must synchronize based on a message sent from the master at the beginning of every TDMA cycle. Despite its simplicity, this protocol met our design goals of maximizing data throughput by preventing collisions without incurring the time overhead of carrier sense [9]. During its designated time slot, a slave node transmits the packet passed to it from the IMU after stripping off the start and stop bytes. An added advantage of the TDMA protocol is the implicit ID and time information coupled with each transmission. Based on the time slot during which a packet is received, the master node can automatically attach the ID of the node that transmitted the packet before passing the entire packet to the server for processing. Further, the time synchronization as a result of TDMA allows for the assumption that the measurements from different IMUs received during a given TDMA cycle occurred at the same time. Therefore, the server can affix a timestamp to data based on when it receives the data rather than transmitting a timestamp from the slave node indicating the precise moment when the IMU measurement was made.

The protocol delivers a data rate of approximately 10 packets per second from each of the slave nodes. This rate could be improved by decreasing the packet size, and more tightly matching the TDMA slot length to the transmission time of a packet. In the current configuration, the slot length is fixed at 10ms per node. Further exploration of the transmit and receive task lengths on the slave nodes could also improve throughput. Currently both the receive and transmit tasks operate on a period of 250ms. In the transmit task, a slave device transmits as many times as possible given the TDMA protocol before switching to the receive task, but during the receive task the node will receive a message with low probability because the master sends commands to the slaves infrequently. Reducing the length of the receive period will therefore reduce the slave node's idle time with little impact on the system's performance.

The only command transmitted from the master to the slaves beyond the TDMA synchronization is a ”Recalibrate” command that signals the slave nodes to send messages to their IMUs to perform static calibration. This command is sent only when a users pushes the auxiliary button the master firefly at the beginning of a long period of system operation. While the communication protocol is complicated by this single command, it is important that the master node be able to issue a synchronous recalibration command to all of the nodes to improve the accuracy of the following measurements. The recalibration phase samples the baseline static noise floor of each IMU, so the subject wearing the devices must be stationary while the devices are undergoing recalibration. It is difficult for the subject to remain stationary if each of the devices must be manually turned off and then on to initiate recalibration. The master's recalibrate command circumvents the issue by causing all IMUs to recalibrate instantaneously which greatly reduces the amount of time the patient must remain still.

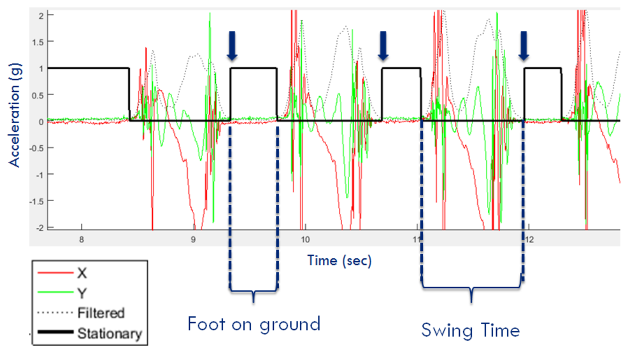
### *B. Gait Analysis*

Our work analyzing subject's gait parameters focused on stride length, swing time, and knee extension. Each of these parameters required different analysis techniques that will be described below.

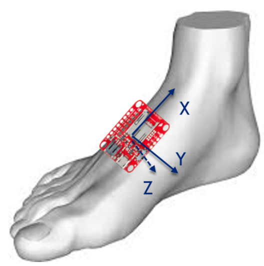
**Stride Length** To determine the stride length, we first assume that the lengths of each segment of a subject's limbs are known. Then we use the attached IMUs to measure their orientations. After getting the length and direction of each limb, we can simulate the gait motion using a model of linked limbs with a fixed rotation axis of the hip. To do that, we can first calculate the 3D position of the knee using the measurement of the thigh, and then calculate the 3D position of ankle using the measurement of the calf and the estimated knee position. Furthermore, if we can detect the step, we can calculate the stride length equals the longitudinal distance between two ankle points.

**Swing Time** Our analysis focuses on detecting the subject's step to in turn ascertain when a subject's foot is on the ground, and when it is in motion. By making the assumption that the acceleration of the foot is close to zero when it is on the ground, we were able to detect the beginning and end of a subject's stride based strictly on acceleration data from the x-y plane. Figure 5(b) shows the orientations of the x,y,z planes relative to a human foot in our calculation. Our implementation is a simplified version of the technique used in [12]. Figure 5(a) shows the basic shape of accelerometer data gathered from a subject's feet as he or she walks and the marked regions indicate the time during which his/her foot is either on the ground or in the air.

**Knee Extension** A subject's knee extension was calculated using the hinge joint techniques explained by Yost [1]. The equations used to calculate the knee extension are shown on



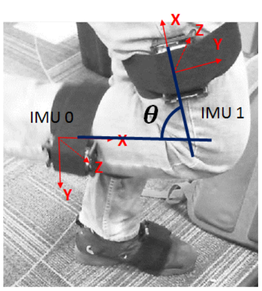

(a)




(b)

Fig. 5. **Step detection graph and IMU references**

the right of Figure 6. The picture on the left shows the angles that are reported by the final calculation and the orientations reported by each of the IMUs used in the calculation. Each quaternion from the IMUs is converted to a local coordinate using rotation matrix respectively. The angle is calculated by dot product of two quaternions projected to the x-axis in the IMUs the local coordinate system.




$$q_0 = q\omega_0 + qx_0 + qy_0 + qz_0$$

$$q_1 = q\omega_1 + qx_1 + qy_1 + qz_1$$

$$q_x = [1\ 0\ 0]$$

$$q_0' = H_0\, q_x \qquad H_0 : Rotation\ Matrix\ of\ q_0$$

$$q_1' = H_1\, q_x \qquad H_1 : Rotation\ Matrix\ of\ q_1$$

$$\theta = \arccos\left(\frac{q_0' \cdot q_1'}{\|q_0'\|\,\|q_1'\|}\right)$$

Fig. 6. **Knee extension Calculation**

## C. User Interface

The user interface is aimed at a technical professional rather that a healthcare provider, but it is built so that certain features are easy to access for non-technical personnel and data storage for future processing is streamlined. Our design uses the popular Robot Operating System (ROS) platform to control data flow into the system software from the single stream of data coming from the master node. The control buttons for data from the individual nodes are simple to access and can be used to change the system's configuration. The interface also contains a slave node health status indicator that reports whether or not the master node has successfully received data from a given slave within a recent time-frame. Figure 7 shows the configuration interface and the health status indicator. In a healthcare environment, these tools would be used by technical personnel to configure the system software before it is used by a healthcare provider.

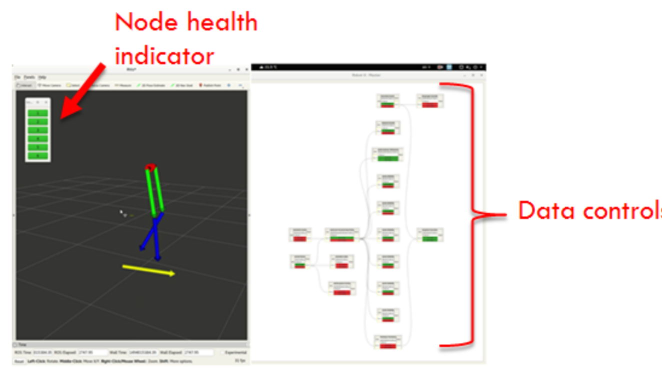
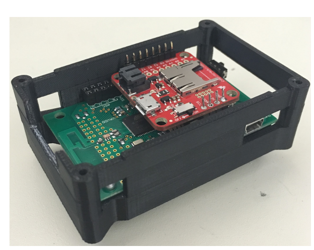



Fig. 7. **ROS user interface**

The user interface also provides a convenient video playback feature than can replay all of the data visualizations produced in real time during active data capture as well as an accompanying video of the subject. Finally, the data are logged automatically for future use or offline analyses.

## D. Enclosure

To ensure that the IMU remains in a fixed position on the body, and to protect the connection between the Firefly and the IMU, we designed a 3D printed enclosure for the slave nodes. Figure 8 shows the enclosure with and without the specially designed lid. The enclosure allows for easy access to all programming ports, power switches and batteries without time consuming disassembly. Further, the design allows a strap to easily pass through the upper and lower pieces of the enclosure so that the devices can be comfortably affixed to subjects using velcro straps. During testing, we found that the straps held the slave devices securely in place and they did not restrict the subject's movements. Figure 9 shows the devices attached to a subject; Figure 9(b) provides a close up of how the device connects to the patient's shoe.



(a)

(b)

Fig. 8. **Enclosure for sensor nodes**

(a) (b)

Fig. 9. **Sensors attached to subject**

## IV. Results

Our results are best demonstrated through a series of the screenshots acquired during multiple demos. They capture the objective of our project and demonstrate the value of the current prototype as well as the points where improvements could be made.

### A. Visualization

Our final data visualization included a simple avatar and representations of the orientation of all six slave nodes. Figure 10 shows the final visualization layout. Both elements of the visualization moved in real time as data passed in from the master. For the orientations, each node's axes move independent of the other nodes, so the movements provide the user with a sense of the raw noise included in the orientation measurements. If any of the node's IMUs reports some kind of spurious, outlying data point, the user will observe a sudden change in that IMU's axis representation. Future work would filter out sudden changes to prevent this phenomena.

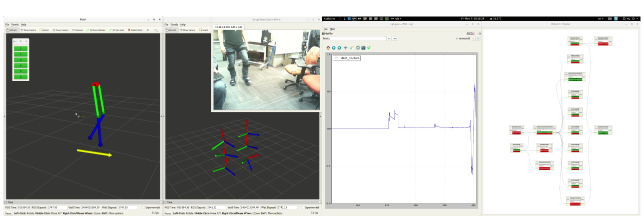


Fig. 10. **Final visualization layout**

### B. Gait parameter graphs

In addition to visualizing a subject's movements, our system can generate graphs of a subject's knee extension and stride length in real time. Figure 11 shows the graph of knee extension on the right as our subject bends his knee. The graph contains separate extension angles for the left and right knee, so it is simple to notice if the range of motion in one knee is more restricted than the other.

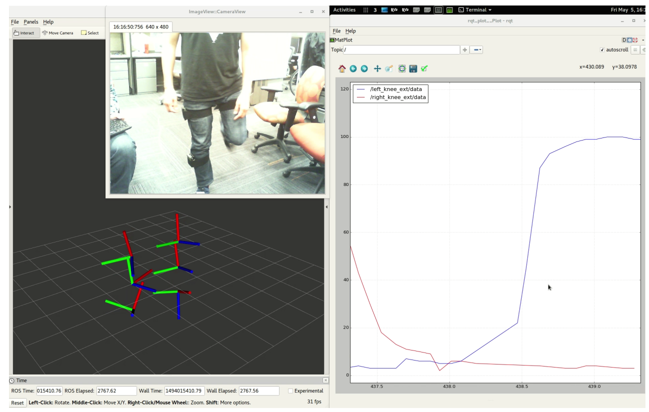



Fig. 11. **Real time knee extension reporting**

The stride length graph appears in nearly real time, but requires additional processing of the segments of multiple data points to find where steps occur, thus delaying it slightly behind real time. However, it still provides useful, prompt feedback on the length of a subject's stride as shown in Figure 12. The graph shows the distance between the right and left foot, so the rise and fall pattern observed in the graph in this figure indicates the feet separating as a subject's leg moves forward and coming back together as the subject swings one foot past the other on the next step.

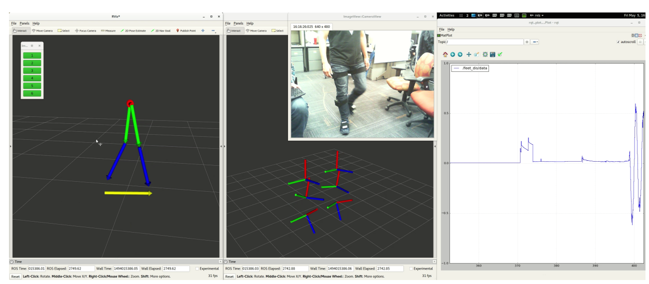


Fig. 12. **Stride length reporting**

### C. Fall Detection

In testing our gait monitoring system, we found that the orientation reported by the nodes strapped to a patient's thighs provided a reliable fall indicator for our subject. When the subject is walking, the measurement is stable and indicates that the node is perpendicular to the ground plane. If a subject falls, the orientation data indicates a rapid change in the angle between the node's central axis and the ground as demonstrated in Figure 13. Though our system relies primarily on the obvious change in the graph that a system operator can visually observe, an automatic mechanism based on a set threshold for the angle between the nodes and the ground would suffice to warn caregivers that a patient has fallen.

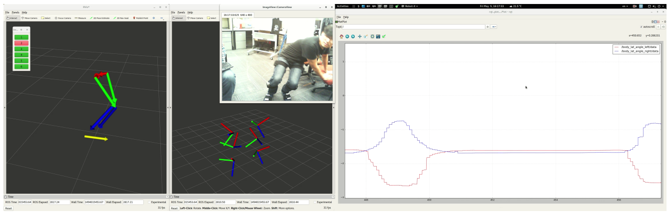


Fig. 13. **Angle reporting for fall detection**

## V. Conclusion

Safewalk is a proof-of-concept implementation of gait monitoring using low cost IMUs and a small number of wireless sensors. We were able to measure relevant parameters of our subject's gaits and visualize the data in an accessible fashion despite the relatively low data transfer rate in our network. Safewalk represents an example of a low cost monitoring solution for patients with fall risks, and it could provide additional data for healthcare providers seeking to customize a plan of care to a patient's behavior outside of the office. We worked through several major challenges including timing mismatch between the IMU and the Firefly, packet loss in the network, IMU drift, and signal processing subtleties.

## VI. Milestones and Task Division

We achieved most of the milestones that we set out to accomplish in our original project proposal, though some of the deadlines for the different milestones shifted. Figure 14 is a timeline of our original deliverable schedule and an explanation of when each deliverable was finished or how it changed.

| Date | Deliverable | Status |
|---|---|---|
| 3/21 | Attach IMU to Firefly and enclosure | MET |
| | Transfer IMU data through Firefly to PC | Moved to 4/4 |
| | Calibrate single IMU for orientation capture | MET |
| 4/4 | Visualize data from single sensor on PC | MET |
| | Synchronize master and slave clocks | Moved to 4/18 with new MAC protocol |
| | Intermediate Demo | MET |
| 4/18 | Gather data from all slave nodes | MET |
| | Measure packet loss | Never finalized, approximated for 5/4 |
| | Complete multiple IMU calibration | Deemed infeasible given IMU drift characteristics |
| 5/4 | Extract gait measurements from sensors | MET |
| | Finalize visualization of all gait data | MET |
| | Final demo | Met |

Fig. 14. **Timeline of deliverables**

The table below reports the segments of the project each team member completed.

**Alex:**
- IMU calibration feasibility assessment
- ROS system development and management
- Motion capture and visualization

**Emily:**
- Data transfer from IMU to Firefly
- Inter-Firefly communication management
- Final report organization

**Iljoo:**
- Knee extension calculation
- Step detection research and implementation
- Enclosure fabrication


## Acknowledgements

We thank the Arjit Arora, Robotics Engineer, 4moms for assistance with design and fabrication of enclosures for sensor nodes. We would also like to show our gratitude to the Dr. Seungmoon Song and Dr. Hanbyul Joo, Robotics Institute, Carnegie Mellon University for sharing their pearls of wisdom with us during the course of this project. We are also immensely grateful to Prof. Rajkumar, Shunsuke Aoki, and Sandeep Dsouza for their support.


## References


[1] Calculating Angles Between Two Yost Labs 3-Space Sensor Devices on a Human Body. Technical report, Yost Labs.

[2] F. Dadashi, B. Mariani, S. Rochat, C. J. Bla, B. Santos-Eggimann, and K. Aminian. Gait and foot clearance parameters obtained using shoe-worn inertial sensors in a large-population sample of older adults. *Sensors*, 14(1):443–457, 2014.

[3] D. Graurock, T. Schauer, and T. Seel. Automatic pairing of inertial sensors to lower limb segments  a plug-and-play approach. *Current Directions in Biomedical Engineering*, 2(1):220 – 231, 2016.

[4] M. Jerrell. Walter reeds state-of-the-art biomechanics lab, 2008.

[5] 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. *Journal of NeuroEngineering and Rehabilitation*, 2(1):6, 2005.

[6] K. Karathanasis, A. Kiourti, and J. L. Volakis. A wireless body area network for carefree medical sensing. In *2014 IEEE Antennas and Propagation Society International Symposium (APSURSI)*, pages 317–317, July 2014.

[7] C. Kirtley, M. W. Whittle, and R. J. Jefferson. Influence of walking speed on gait parameters. *Journal of Biomedical Engineering*, 7(4):282–288, 2017/05/10.

[8] M. e. a. Martins. Assessment of walker-assisted gait based on Principal Component Analysis and wireless inertial sensors. *Revista Brasileira de Engenharia Biom*, 30:220 – 231, 09 2014.

[9] R. Rajkumar. Medium access control (mac) protocols. 2017.

[10] A. Rowe. Point coordination function tdma, August 2010.

[11] W. Tao, T. Liu, R. Zheng, and H. Feng. Gait analysis using wearable sensors. *Sensors*, 12(2):2255–2283, 2012.

[12] X. Technologies. Gait tracking with x-imu, 2013.