

# **UCLA**

## **UCLA Electronic Theses and Dissertations**

### **Title**

Robust Human Activity Classification and Motion Monitoring Systems Using Inertial Sensors

### **Permalink**

<https://escholarship.org/uc/item/6qn2247s>

### **Author**

Wu, Xiaoxu

### **Publication Date**

2016

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA  
Los Angeles

**Robust Human Activity Classification and  
Motion Monitoring Systems Using Inertial  
Sensors**

A dissertation submitted in partial satisfaction  
of the requirements for the degree  
Doctor of Philosophy in Electrical Engineering

by

**Xiaoxu Wu**

2016

© Copyright by

Xiaoxu Wu

2016

ABSTRACT OF THE DISSERTATION

**Robust Human Activity Classification and  
Motion Monitoring Systems Using Inertial  
Sensors**

by

**Xiaoxu Wu**

Doctor of Philosophy in Electrical Engineering

University of California, Los Angeles, 2016

Professor Gregory J. Pottie, Chair

The proliferation of powerful microcomputers and the development of modern machine learning tools have enabled human daily activity monitoring systems using wearable inertial sensor like accelerometers and gyroscopes. These systems fulfilled the urgent need in health and wellness industries in helping doctors and clinicians during diagnosis, treatments and rehabilitation processes for neurological diseases like strokes and Parkinson's.

For most current activity monitoring systems, there exists an assumption that the sensors are always securely and correctly mounted by the users. Unfortunately, such assumptions do not hold as the scale of studies increase. And it is especially challenging for subjects with neurological diseases to follow instructions about how to mount the sensors everyday, because some of the elderly tend to be technophobic and neurological diseases are often accompanied with cognitive difficulties. Errors in sensor mounting pose can cause large amount of data loss and distortion and will affect the robustness of the systems severely.

In observance of these issues, a series of solutions for sensor orientation and position errors in human motion monitoring and activity classification will be pre-

sented. Opportunistic calibration methods to find the true sensor orientation and position will be discussed. In addition, systems that provide robust monitoring regardless of the exact sensor pose will be proposed.

The dissertation of Xiaoxu Wu is approved.

Lara Dolecek

Mario Gerla

William J. Kaiser

Gregory J. Pottie, Committee Chair

University of California, Los Angeles

2016

*To my mother . . .  
who—among so many other things—  
saw to it that I learned to touch-type  
while I was still in elementary school*

## TABLE OF CONTENTS

<b>1</b>	<b>Introduction . . . . .</b>	<b>1</b>
1.1	Motivations . . . . .	1
1.2	Scope of This Thesis . . . . .	2
1.3	Prior Works . . . . .	3
1.3.1	Sensor Misplacement Problems . . . . .	3
1.3.2	Lower Body Motion Tracking . . . . .	4
1.4	Background Knowledge . . . . .	5
1.4.1	Inertial Sensors . . . . .	5
1.4.2	Rigid Body 3-dimensional Orientation Representations . .	6
1.4.3	Possibilities of Sensor Misplacement . . . . .	8
1.5	Thesis Organization . . . . .	9
<b>2</b>	<b>Sensor Orientation Error Correction And the Application to Mo-</b>	
<b>tion Tracking . . . . .</b>		<b>11</b>
2.1	Problem Statement . . . . .	11
2.2	Methodology . . . . .	12
2.2.1	Sensor Misorientation . . . . .	12
2.2.2	Rotational Displacement . . . . .	13
2.3	Experiments and Results . . . . .	15
2.3.1	Data Collection . . . . .	16
2.3.2	Misorientation . . . . .	17
2.3.3	Rotational Displacement . . . . .	19
2.4	Discussions . . . . .	22

<b>3 Sensor Placement Error Correction And the Application to Motion Tracking . . . . .</b>	<b>24</b>
3.1 Problem Statement . . . . .	24
3.2 Methodology . . . . .	25
3.2.1 Experimental Instrumentation . . . . .	25
3.2.2 Data Preprocessing . . . . .	25
3.2.3 Foot-sensor Gait Reconstruction Using ZUPT . . . . .	27
3.2.4 Ankle-sensor Gait Reconstruction Using Non-ZUPT . . . . .	28
3.3 Experiments and Results . . . . .	30
3.3.1 Experimental Procedure . . . . .	30
3.3.2 Results and Analysis . . . . .	32
3.4 Discussions . . . . .	36
<b>4 A Robust Motion Tracking Algorithm regardless of Sensor Placement and Orientation . . . . .</b>	<b>37</b>
4.1 Problem Statement . . . . .	37
4.2 Methodology . . . . .	38
4.2.1 Inverted Pendulum Model . . . . .	38
4.2.2 Step Length Estimation - the testing phase . . . . .	39
4.2.3 Leg Length Estimation - the training phase . . . . .	43
4.3 Experiments and results . . . . .	44
4.3.1 System Set-up . . . . .	44
4.3.2 Experiment Design . . . . .	45
4.3.3 Results and Analysis . . . . .	46
4.4 Discussions . . . . .	48

<b>5 Orientation Self-correcting Daily Activity Classification . . . . .</b>	<b>50</b>
5.1 Problem Statement . . . . .	50
5.1.1 Motion Sensing System Setup . . . . .	51
5.1.2 Trial Implementation . . . . .	53
5.1.3 Current SIRRACt Performance . . . . .	53
5.2 Methodology . . . . .	54
5.2.1 Layer 1: a conservative walking detection classifier . . . . .	55
5.2.2 Orientation Correction . . . . .	56
5.2.3 Layer 2: a orientation dependent binary classifier . . . . .	58
5.3 Experiments and Results . . . . .	59
5.3.1 Data Collection and Subject Description . . . . .	59
5.3.2 Ground Truth Labeling . . . . .	60
5.3.3 Experiment Design . . . . .	61
5.3.4 Results . . . . .	61
5.4 Discussions . . . . .	63
<b>6 Conclusion . . . . .</b>	<b>65</b>
6.1 Contributions . . . . .	65
6.2 Future Work . . . . .	67
<b>References . . . . .</b>	<b>69</b>

## LIST OF FIGURES

1.1	Axis-Angle Representation of a 3D Rotation . . . . .	6
1.2	Definition of all sensor misplacement possibilities . . . . .	8
2.1	System block diagram for misorientation . . . . .	13
2.2	System block diagram for rotational displacement . . . . .	15
2.3	Data collection setup . . . . .	17
2.4	Signal recovery for misorientation . . . . .	18
2.5	Step-length-measurement accuracy for misorientation before and after calibration . . . . .	19
2.6	Signal recovery for rotational displacement . . . . .	20
2.7	Key Walking Phases of Motion Reconstruction . . . . .	21
3.1	Experimental Setup . . . . .	26
3.2	Gait Segmentation . . . . .	27
3.3	System Block Diagram For Trajectory Reconstruction . . . . .	29
3.4	System Block Diagram For Sensor Position Estimation . . . . .	31
3.5	Velocity Update Before and After Using ZUPT And Non-ZUPT Method . . . . .	33
4.1	The Inverted Pendulum Model for Human Walk (Figure adapted by author from [Kuo10] and [SR06]) . . . . .	39
4.2	The Gait Segmentation Adopted in this Method (Figure adapted by author from [LHW07]) . . . . .	40
4.3	Gait segmentation method comparison . . . . .	41
4.4	Heel-strike Detection Results for Left and Right Ankle Sensors . .	42

4.5	The IMU sensors used in this chapter. (a) the sensor size, (b) the front view of sensor mounting, (c) the side view of sensor mounting	44
4.6	The KINECT2 sensor . . . . .	45
4.7	The KINECT joints tracking tool . . . . .	45
5.1	Subject Kit Devices . . . . .	52
5.2	Correct Sensor Placement . . . . .	54
5.3	System high level description . . . . .	55
5.4	Detected Walking Beacons . . . . .	57
5.5	DTW orientation segment matching . . . . .	58
5.6	Matlab Labeling Tool GUI . . . . .	60

## LIST OF TABLES

1.1	Summary of three possibilities of sensor misplacement . . . . .	9
2.1	Step-Length-Measurement Accuracy . . . . .	23
3.1	Motion Tracking Error for Flat Floor Walking . . . . .	34
3.2	Motion Tracking Error For Stairs Walking . . . . .	35
4.1	Absolute error rate of the total walking distance estimation . . . .	47
4.2	Performance comparison of the two methods with Trial 1 (same sensor position with training section) and Trial 2 (different sensor position with training section). The Pose Invariant (PI) method is more robust with either trial while non-ZUPT only works well under Trial 1 . . . . .	48
5.1	subject informations . . . . .	59
5.2	Algorithm Design (Details of F1 and F2 listed in Table 5.3) . . . .	62
5.3	Details of the Two Feature Set . . . . .	63
5.4	Classification Results . . . . .	64

## ACKNOWLEDGMENTS

First of all, I would like to express my sincere gratitude to my advisor Prof. Greg Pottie for the continuous support during my Ph.D study. With endless patience and immense knowledge, he guided me through all the time of my research and writing this thesis. The advice from him have always been sources of inspiration and encouragement that carry my through the hard and easy time of my Ph.D study.

Besides, I would like to thank the rest of my thesis committee: Dr. William Kaiser, Dr. Mario Gerla and Dr. Lara Dolecek for their insightful comments and encouragement, as well as the questions that inspired me to explore my research projects deeper and wider.

My sincere thanks also goes to Dr. Andrew Dorsch, Mr. Seth Thomas, Dr. Xiaoyu Xu, Mr. Yeung Lam and Mr. Eric Yuen for coordinating with patient subjects for data collection and giving me access to the sensing devices and computing platform. Without their support, it would not be possible for me to conduct this research.

I thank my fellow labmates Dr. Yan Wang, Dr. Chieh Chien, Dr. Hua-I Chang and Dr. James Xu for the stimulating discussions, the hardworking time and all the fun we have had in the past four years.

Last but not the least, I would like to thank my family: my husband Mr. Yufei Mao for the inspirational discussions and my parents Mr. Zinan Wu and Ms. Jiahui Wang for supporting me spiritually throughout my Ph.D study.

## VITA

2010	B.E. (Communications Engineering), Fudan University, Shanghai, China
2012	M.S. (Electrical Engineering), UCLA, Los Angeles, California.
2014, 2015	Interim Engineer, Qualcomm. Inc., UCLA.
2014–2015	Teaching Fellow, Electrical Engineering Department, UCLA.
2012–2016	Graduate Student Researcher, Wireless Health Institute, UCLA.

## PUBLICATIONS

- Xiaoxu Wu, X. Xu, G. Pottie, W. Kaiser, A double layer orientation correction method for human activity recognition, Body Sensor Networks 2016, San Francisco CA, June 14-17, 2016 (accepted)
- Xiaoxu Wu, Y. Wang, G. Pottie, A Robust Step Length Estimation System for Human Gaits, Wireless Health 2015, Bethasta MD, Oct 14-16, 2015
- Xiaoxu Wu, Y. Wang, G. Pottie, A Robust Step Length Estimation System for Human Gaits, Wireless Health 2015, Bethasta MD, Oct 14-16, 2015  
Xiaoxu Wu, Y. Wang, G. Pottie, A Non-ZUPT Gait Reconstruction Method for Ankle Sensors, 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Chicago IL, Aug 26-30, 2014

- Y. Wang, J. Xu, Xiaoxu Wu, G. Pottie, W. Kaiser, A simple calibration for upper limb motion tracking and reconstruction, 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Chicago IL, Aug 26-30, 2014
- Xiaoxu Wu, Y. Wang, C. Chien, G. Pottie, Self-Calibration of sensor misplacement based on motion signatures, Proc. 2013 IEEE International Conference on Body Sensor Networks, Cambridge MA, May 6-9, 2013.
- H. Chang, Xiaoxu Wu, C. Huang, Y. Wang, L. Dolecek, G. Pottie. Virtual Inertial Measurements for Motion Inference in Wireless Healthaccepted by Asilomar Conference on Signals, Systems, and Computers , 2013
- Y. Wang, J. Xu, X. Xu, Xiaoxu Wu, G. Pottie, W. Kaiser. Inertial Sensor Based Motion Trajectory Visualization and Quantitative Quality Assessment of Hemiparetic Gait, BodyNets, 2013

# CHAPTER 1

## Introduction

### 1.1 Motivations

The proliferation of powerful mobile computers and the development of modern machine learning tools have made it possible for human daily activity monitoring using inertial sensors with light weight, small size and cheap price. Such daily activity monitoring has been proven to be a key indicator of human health status [DC15] [DTD15]. With these advances, many of the most urgent problems in health and wellness promotion, diagnostics, and the treatment of neurological conditions can be addressed.

Similar to most machine learning systems, missing and erroneous data issues emerge as the scales of applications increase. Current state-of-the-art machine learning systems can be made to be adaptive to such problems with examples like EM algorithms [DLR77] as long as a sufficient amount of data is available. But such approaches can be hard to apply in the wireless health context, because sufficiently large datasets with ground truth labeling are generally very expensive and usually not available.

Therefore, we pursue the solution of defining common possibilities of data loss or distortion and try to solve each problem with a model. We found that one of the most commonly seen issues in most human activity recognition systems are sensor placement errors. Unfortunately, such errors are especially common among patient populations since elderly patients often tend to be technophobic

and many neurological diseases are accompanied with cognitive issues. All these factors make it harder for neurological disease patient with mobility impairments to follow instructions about the sensor placement everyday.

Since most current state-of-the-art activity monitoring systems assume good sensor placement, it can be very valuable to find general methods of signal correction for sensor misplacement. Such misplacement correction algorithms will make human activity monitoring systems more powerful and adaptive to real world applications.

## 1.2 Scope of This Thesis

The scope of this thesis focuses on sensor misplacement issues and how solving these issues can make activity recognition as well as motion tracking systems more reliable.

Activity recognition systems enable monitoring of quantity of daily activities, which means the amount of time that a subject spend walking, sitting or sleeping etc. These metrics provides valuable information towards the health condition of a subject and can be easily used as daily exercise prescriptions.

In addition, motion trajectory reconstruction problems that enable monitoring of detailed quality of daily activities are also of great interest. Metrics like step length, walking speed, duration of bouts etc. can be extracted given detailed trajectories. These metrics serves as important references for clinicians and doctors during diagnosis and rehabilitation processes.

We limit our discussions to lower body activities because they are the most frequently prescribed activities during rehabilitation processes for patients with mobility difficulties and they are proven to be strongly related to people's health status. [Pub09]

## 1.3 Prior Works

### 1.3.1 Sensor Misplacement Problems

One of the most widely used approaches for sensor misplacement issues is to identify placement invariant features. [KL08] studied how accelerometer and gyroscope signals are affected by sensor displacement. They showed that the acceleration component due to rotation is especially sensitive to sensor displacement. And thus, by avoiding segments of signals where rotation dominates, the effect of sensor displacement can be significantly reduced. [FBR09] proposed a feature extraction method based on genetic programming. Significant performance improvement for gesture recognition systems were reported using sensor position invariant features. [ZD14] propose a novel gait representation for accelerometer and gyroscope data which is sensor-orientation-invariant. A series of high-performance gait biometrics were explored and used for classification purposes. These methods achieve robustness of the recognition systems by using features that are not affected by misplacement of sensors. But the main issue for these approaches is that we often have to compromise on the overall performance. Such compromises can be intolerable in cases where we want to discriminate between activities with directions (e.g. walking upstairs versus walking downstairs). In addition, such approaches are only applicable for classification problems and do not help when exact motion trajectories are of interest.

Another popular solution for sensor misplacement issues is to calibrate sensor signals with predefined activities. [GBN11] uses the distribution of gravity on 3 axies during static activities (e.g. standing, sitting) to transform the accelerometer signals from the device coordinate system to the body reference coordinate system, making the signals not sensitive to sensor orientation. [FHC12] propose a very similar algorithm. During several pre-defined static activities, the true mounting orientations of the sensors were estimated. Then sensor signals can be corrected

using the calibration information. These methods all require pre-defined calibration activities performed by the users. Such requirements are usually not feasible for large scale applications, because assuming all users to follow the instructions about quantity as well as quality of calibration activities is not realistic.

### 1.3.2 Lower Body Motion Tracking

In addition to the widely used activity recognition systems that provide quantities of daily activities, wearable inertial sensors have found their power in more detailed motion tracking systems. By reconstructing the detailed trajectory of certain activities, more metrics can be extracted making quality of daily activities also available.

[WCX13a] proposed tracking lower body motion by using a double pendulum model. But this model needs two sensors on each side (one on the thigh and one on the ankle) for accurate estimation. Moreover, the requirement of correct orientation and placement of the sensors limits its practical application.

The other approach is double integration of the motion acceleration. A zero-velocity-update (ZUPT) method is applied to compensate the cumulative error [BBS08][SMS05][WXX13]. The ZUPT method needs only one sensor and is extensively employed because of its robustness and easy implementation. This ZUPT method assumes that there exists a stable zero-velocity interval within each stride and this critical assumption only holds when the sensor is mounted on the foot. However, in most medical applications, clinicians prefer ankle-mounted positions for lower body motion tracking[DXB11][XBK11][DD11]. This is because, not all patients wear shoes with laces everyday. It happens frequently that they don't even wear shoes when it is important to capture activity. In contrast, ankle-mounted sensors allow clinicians to deploy the sensors without considering these constraints. Hence, a new velocity update method for ankle-mounted sensors is

needed.

## 1.4 Background Knowledge

In this section, background knowledge regarding the sensors used, possibilities of sensor misplacement and 3D rotation representation are presented with the hope of ease of reading for the rest of the thesis.

### 1.4.1 Inertial Sensors

The most widely used inertial sensors are 3-axes accelerometers, gyroscopes. We will introduce some basic knowledge about these two types of devices.

MEMS accelerometers can be modeled by a spring-mass system where the acceleration of a mass is measured by the displacement of the connected spring. The output measurements are the summations of gravity and motion acceleration. The gravity component has a magnitude equal to the gravity constant of  $9.8m/s^2$ , and this magnitude is distributed among the 3 axes based on the current sensor orientation. The motion acceleration component is dependent on the traveling acceleration. When the sensor stays still, the accelerometer outputs only the gravity component and has a constant magnitude. But when the sensor is moving, the two components of gravity and motion acceleration add up to the total measurements and can be hard to separate.

MEMS gyroscopes measure angular velocity using the fact that a vibrating object tends to continue vibrating in the same plane even if its support rotates. And the Coriolis effect causes a force from the object applied to the support, and by measuring this force the angular velocity can be determined. Therefore, the exact position of the sensor will not affect the gyroscope measurements. But an orientation error will affect the distribution of total angular velocity among the 3 axes.

### 1.4.2 Rigid Body 3-dimensional Orientation Representations

Four ways of describing a rotation in 3-dimensional space are introduced in this section. They are very useful tools as we proceed to the rest of this thesis about detailed algorithms.

#### 1. Euler Angle.

Euler angle is the most intuitive way of describing a 3D rotation. Euler angles represent a sequence of three elemental rotations, i.e. rotations about the axes of a coordinate system. For instance, a first rotation about z by an angle  $\alpha$ , a second rotation about x by an angle  $\beta$ , and a last rotation again about y, by an angle  $\gamma$ . These rotations start from a known standard orientation [Wik16a]. This representation is a straight forward way of describing a rotation. But the Euler Angle representation suffers from gimbal lock issues and introduces complexity in tracking. Therefore, this representation is not used in this thesis.

#### 2. Axis-angle

The axis-angle representation is a second way to describe a rotation. It consists of a unit vector  $e$  indicating the direction of an axis of rotation, and an angle  $\theta$  describing the magnitude of the rotation about the axis  $e$  (Figure 1.1).

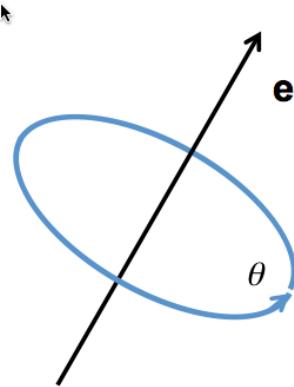


Figure 1.1: Axis-Angle Representation of a 3D Rotation

### 3. Rotation Matrix

We can also represent a 3D rotation in a 3 by 3 orthogonal rotation matrix with determinant of 1. The rotation matrix is the most widely used representation in this thesis because of the clear mathematical form. A 3D vector representing a sensor measurement can be directly projected onto a rotated coordinate system by simply applying the right hand rule in Equation 5.3 where  $\mathbf{a}_{rot}$  and  $\mathbf{a}_{orig}$  are sensor measurements in the rotated coordinate and original coordinate respectively and the  $\mathbf{R}$  is the rotation matrix.

$$\mathbf{a}_{rot} = \mathbf{a}_{orig}\mathbf{R} \quad (1.1)$$

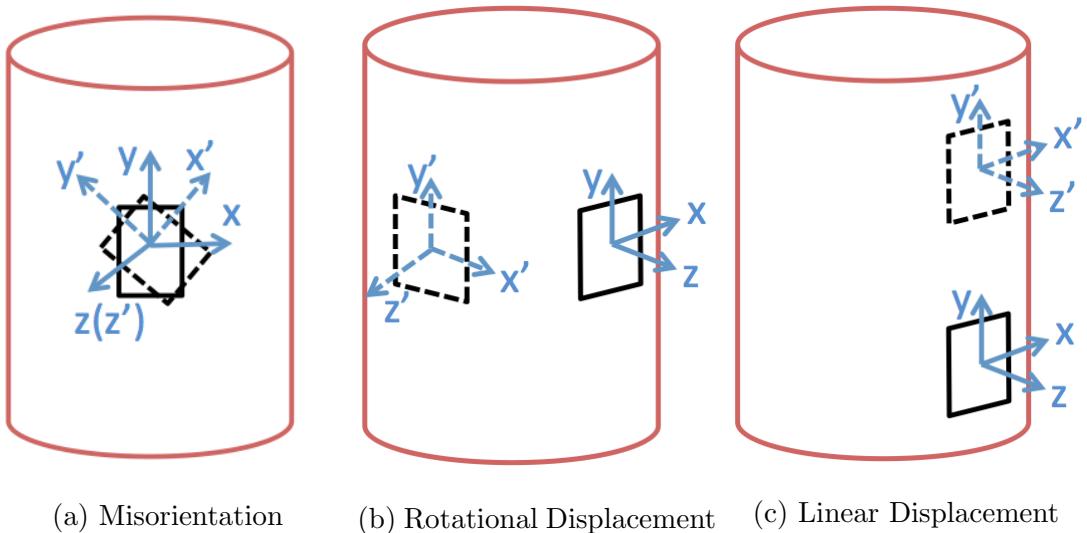
### 4. Orientation Quaternion

Another important rotation representation is the quaternion. A quaternion is a simple four-number encoding of the axis-angle representation of a rotation. A rotation of  $\theta$  around axis  $\mathbf{e}$  can be represented as a quaternion in the form of Equation 1.2 where  $\mathbf{i}$ ,  $\mathbf{j}$  and  $\mathbf{k}$  are unit vectors representing the three Cartesian axes.

$$\cos \frac{\theta}{2} + (\mathbf{e}_x i + \mathbf{e}_y j + \mathbf{e}_z k) \sin \frac{\theta}{2} \quad (1.2)$$

Compared to Euler angles they are simpler to compose and avoid the problem of gimbal lock. Compared to rotation matrices they are more numerically stable and may be more efficient [Wik16b]. If we wish to rotate a sensor measurement in 3D space  $\mathbf{a} = (\mathbf{a}_x, \mathbf{a}_y, \mathbf{a}_z)$  by a rotation in quaternion form  $\mathbf{q}$ , the sensor measurement in the rotated coordinate system  $\mathbf{a}' = (\mathbf{a}'_x, \mathbf{a}'_y, \mathbf{a}'_z)$  can be calculated in Equation 1.3 using the Hamiltonian product.

$$\mathbf{a}' = \mathbf{q}\mathbf{a}\mathbf{q}^{-1} \quad (1.3)$$



(a) Misorientation      (b) Rotational Displacement      (c) Linear Displacement

Figure 1.2: Definition of all sensor misplacement possibilities

#### 1.4.3 Possibilities of Sensor Misplacement

Sensor misplacement can vary from case to case. But assuming that a sensor is always placed such that the x-y plane is firmly attached to the limb, all misplacement within a limb can be decomposed into the three cases shown in Figure 1.2.

*Misorientation* means a sensor is placed at the correct position but the sensor is rotated around the z axis (shown in Figure 1.2a). In this case, z-axis sensor signals are not distorted. As a result, the distortion can be modeled by a rotation around the z axis. Both accelerometer and gyroscope measurements distortion can be modeled using a rotation matrix in Equation 1.4.

$$R_{misorientation} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1.4)$$

In a *rotational displacement* case (shown in Figure 1.2b), we assume that the distance between the correctly placed sensor and the incorrectly placed sensor

	misorientation	rotational displacement	linear displacement
Accelerometer	$R_{misorientation}$	$R_{rotational}$	gravity not affected motion acceleration affected
Gyroscope	$R_{misorientation}$	$R_{rotational}$	not affected

Table 1.1: Summary of three possibilities of sensor misplacement

is negligible. Both accelerometer and gyroscope measurements distortion can be represented by a rotation matrix described in Equation 1.5

$$R_{rotational} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad (1.5)$$

In a *linear displacement* case (shown in Figure 1.2c), the orientation of the sensor is not corrupted. Therefore, the gyroscope measurements are not changed. For accelerometer measurements, the gravity component is also not changed due to the fact that the orientation is correct. However, the motion acceleration can be affected due to the fact that position of the sensor is changed.

In all three cases, the impact on accelerometer and gyroscope measurements are summarized in Table 1.1

## 1.5 Thesis Organization

The rest of this thesis is organized as follows.

In Chapter 2, an opportunistic calibration method for finding sensor mounting orientation will be described and discussed. Two-dimensional rotation matrix models will be built and evaluated for cases of misorientation and rotational displacement.

Following that, in Chapter 3, a method for estimating the sensor mounting position will be discussed. In addition, knowing the sensor position will lead up to the possibility of lower body motion tracking using only one sensor on the ankle.

After the discussion of calibration methods in finding the orientation and position of the sensors and their applications to the lower body motion tracking in these two chapters, we are ready to move on to more advanced models that are robust to sensor mounting positions and orientations without requiring necessary calibration process.

In Chapter 4, a step-length estimation method will be discussed by using the quaternion change within each stride. This novel differential model requires no knowledge of the sensor mounting orientation or position, and is therefore, very robust to all types of sensor misplacement.

In Chapter 5, automatic orientation correction was achieved by using a double layer model. In the first layer, a conservative orientation invariant model was used to detect walking activities with very high confidence. Then these walking activity signals are compared with the training template to calculate the sensor orientation information. With this sensor orientation information, the sensor signal can be corrected and fed into the second layer orientation variant classifier with higher accuracy.

Finally, this thesis will be concluded with Chapter 6.

Centered around inertial sensor misplacement problems, this thesis will propose several methods with the theme of opportunistic identification of certain segments of the data and using them to serve as calibration points or anchors for the rest.

## CHAPTER 2

# Sensor Orientation Error Correction And the Application to Motion Tracking

### 2.1 Problem Statement

As is discussed in Chapter 1, one of the most commonly seen problems in human activity recognition systems is sensor placement errors. The frequent occurrence of signal distortion due to sensor placement errors can cause serious degradation of the system performance.

In reaction to the problem of sensor placement errors, we propose a novel calibration process based on repetitive motion signatures available in daily life activities in this chapter. We choose walking as one example of such motion signatures and show that it contains meaningful information that can be applied to calibration for lower limb sensors.

The main contributions for this chapter are as follows. A novel calibration method based on motion signatures was developed. To demonstrate the effectiveness of the proposed approach, we examine the sensor signal recovery. Specifically, we study the application to lower body motion tracking and step-length measurement.

## 2.2 Methodology

In this section, we will study two common sensor misplacement cases: misorientation (Figure 1.2a) and rotational displacement (Figure 1.2b). The model and the system architecture for each one are discussed in detail.

### 2.2.1 Sensor Misorientation

In sensor misorientation, the z axis for the incorrectly placed sensor is exactly the same as the correctly placed sensor (shown in Figure 1.2a). Only the x-y plane is rotated.

#### 2.2.1.1 Model

The relationship between the measurement in two frames can be represented by a 2x2 transformation matrix [KL08]. Since the transformation matrix has a determinant of 1, the unknowns can be reduced from four to one (shown in Equation 2.1).  $\mathbf{x}$  and  $\mathbf{y}$  are vectors that represent the measurement of the correctly placed sensor on the x and y axes over time.  $\mathbf{x}'$  and  $\mathbf{y}'$  are vectors that represent the measurement of the incorrectly placed sensor on x and y axes over time.  $\theta$  is the rotation angle of the x-y plane between the correct frame and the incorrect frame; this angle needs to be estimated during the calibration process.

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} * \begin{bmatrix} \mathbf{x}' \\ \mathbf{y}' \end{bmatrix} \quad (2.1)$$

#### 2.2.1.2 System Architecture

The system block diagram is shown in Figure 2.1. Since during misorientation the sensor x-y plane is rotated, the measurement in a correct frame and the measurement in an incorrect frame are related by Equation 2.1.

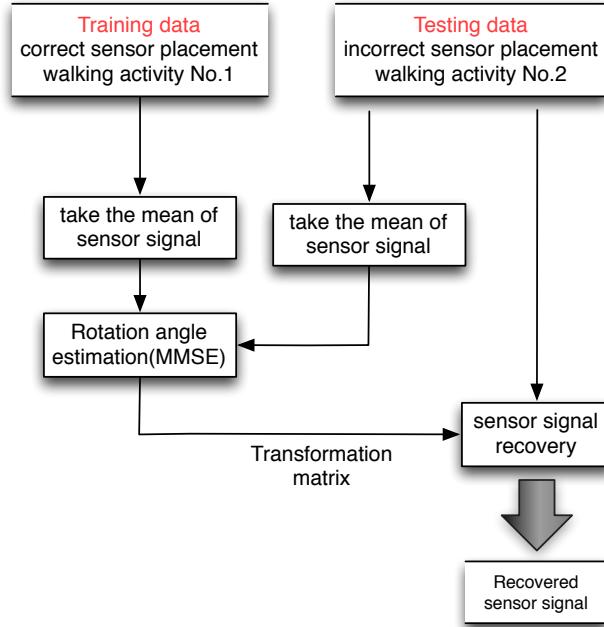


Figure 2.1: System block diagram for misorientation

We take the mean of both training data and testing data on the x and y axes, and use them for the MMSE estimation of the rotation angle in Equation 2.2 where x and y represent the mean of the training data (correct frame measurement) on the x axis and y axis and x' and y' represent the mean of the testing data (incorrect frame measurement) on the x axis and y axis.

Then, the transformation matrix can be decided and applied to the testing data so that the sensor misorientation can be compensated.

$$\hat{\theta} = \arg \min_{\theta} \left\{ \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} * \begin{bmatrix} x' \\ y' \end{bmatrix} \right\} \quad (2.2)$$

## 2.2.2 Rotational Displacement

Assuming that the distance between the correct and incorrect sensor is negligible, the y-axis signal is not distorted and the x-axis and z-axis signals are rotated as

shown in Figure 1.2b.

In the presence of additional assumptions and uncertainties, we now need to use information from gait cycles in a more sophisticated manner. Despite its complexity, this method is applicable to both misorientation and rotational displacement.

### 2.2.2.1 Model

The relationship between the measurement of the correctly placed sensor and the incorrectly placed sensor on the x and z axes can be represented by Equation 2.3. This is valid for both accelerometer and gyroscope signals.  $\mathbf{x}$  and  $\mathbf{z}$  are vectors that represent the measurement of the correctly placed sensor on the x and z axes over time.  $\mathbf{x}'$  and  $\mathbf{z}'$  are vectors that represent the measurement of the incorrectly placed sensor on the x and z axes over time.  $\theta$  is the rotation angle of the x-z plane between the correct frame and the incorrect frame.

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} * \begin{bmatrix} \mathbf{x}' \\ \mathbf{z}' \end{bmatrix} \quad (2.3)$$

### 2.2.2.2 System Architecture

The system block diagram is shown in Figure 2.2. We observe that the energy signal (the square sum of the accelerometer measurement on all three axes) is invariant in sensor misplacement. Thus, we can chop the training and testing data into gait cycles based on the energy signal.

Each gait cycle in the training set was compared to all the other gait cycles in the training set based on a dynamic time warping (DTW) algorithm [RJ93] using energy features. The gait cycle with the minimum energy DTW distance to all the other gait cycles is chosen as the training template, as being in some sense the most typical gait cycle. For the selection of the testing template, we similarly

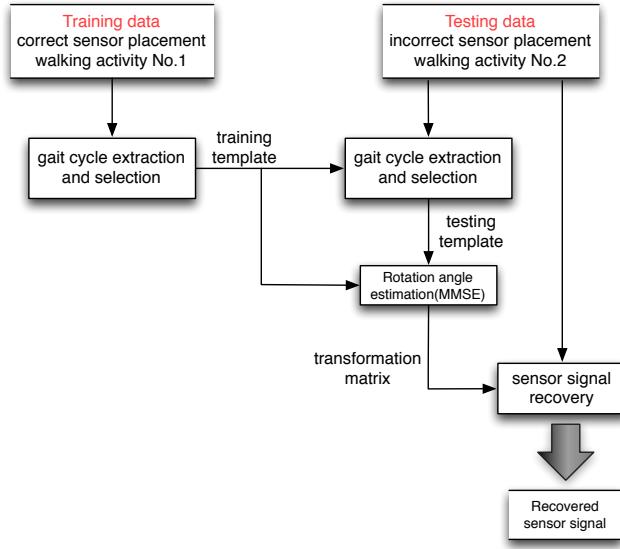


Figure 2.2: System block diagram for rotational displacement

choose the one with the minimum energy DTW distance to the training template.

The training template and the testing template are then synchronized by using the optimal path computed by DTW. Following that, they are used for MMSE estimation of the rotation angle based on Equation 2.4.  $\mathbf{x}$  and  $\mathbf{z}$  are training templates on the x and z axes.  $\mathbf{x}'$  and  $\mathbf{z}'$  are testing templates on the x and z axes.

$$\hat{\theta} = \arg \min_{\theta} \left\{ \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix} - \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} * \begin{bmatrix} \mathbf{x}' \\ \mathbf{z}' \end{bmatrix} \right\} \quad (2.4)$$

## 2.3 Experiments and Results

Sensor placement error has a significant impact on human activity motion tracking. We examine the improvement that sensor placement correction can give us for lower body motion tracking and step-length measurement.

For the two sensor misplacement cases (Figure 1.2a and Figure 1.2b), the data

collection setup, the signal recovery result, and the motion tracking step-length-measurement results are discussed in detail below.

### 2.3.1 Data Collection

Five 9DOF Razor IMUs were used to acquire motion data and the measurements were sent wirelessly through a Bluetooth modem to a tablet. Four sensors were attached to the subjects' waist, thigh, and calf with the x-y plane aligned with the sagittal plane and the y axis with gravity (shown in Figure 2.3). All the sensors were sampled at 50Hz. Signature motions were put at the start and end of the data sequence for synchronization. Stride length was measured directly using a ruled floor surface as well as a means of marking shoe contact by application of marking liquid to the shoe. Using these sensor signals, lower body motion tracking was constructed by the kinematic chain model described in [WCX13b].

Besides these four sensors, the fifth sensor with incorrect sensor placement was attached to the right thigh. The subjects wore both the correctly placed sensors and the incorrectly placed sensor simultaneously throughout all the step-length-measurement experiments.

We here introduce two terms: *intra-instance* and *inter-instance* calibration. In intra-instance calibration, training data and testing data are extracted from the same walking instance. We start with intra-instance calibration to verify the model and algorithm. However, in real life, training data and testing data come from different walking instances, which makes it necessary to study the inter-instance calibration. During an inter-instance calibration, training data come from the correct sensor placement in one walking instance and they are compared to testing data in a different walking instance. They are used to estimate the rotation matrix. For inter-instance calibration, DTW is used to synchronize the training and testing templates.

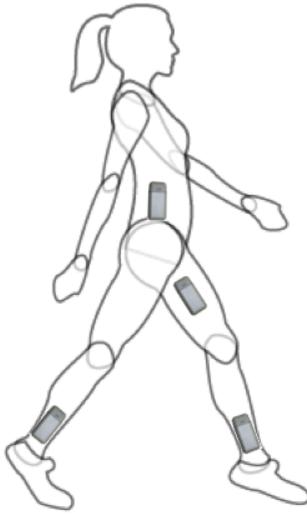


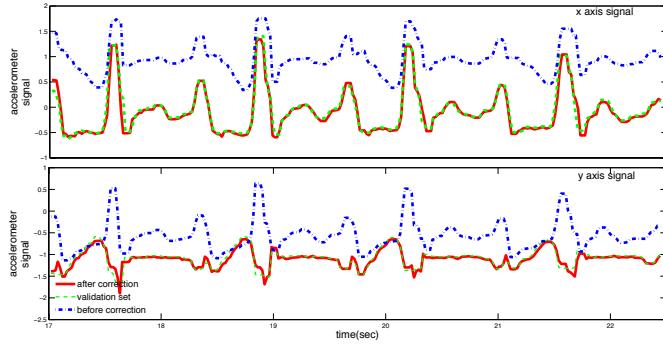
Figure 2.3: Data collection setup

Extracting training and testing data from different walking instances makes it difficult to evaluate the signal recovery performance. So we introduce a validation set for this purpose. In the first walking instance, signals from correctly placed sensors are used as the training set. In the second walking instance, signals from incorrectly placed sensors are used as the testing set, and signals from correctly placed sensors are used as the validation set. By comparing the training set and the testing set, we extract the rotation angle and hence determine the transformation matrix. Following that, the transformation matrix is applied to the testing set so that we can compare the outcome with the validation set.

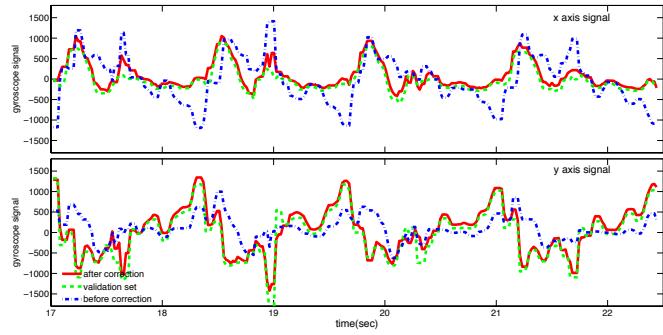
### 2.3.2 Misorientation

#### 2.3.2.1 Signal Recovery Result

During misorientation, the position of the sensor is correct and there is only an angle distortion from the correctly placed sensor. Thus, the 1D rotation model for the x-y plane describes the problem accurately. Figure 2.4 shows that the testing



(a) accelerometer signal



(b) gyroscope signal

Figure 2.4: Signal recovery for misorientation

signal after calibration matches the validation set very well. Both gyroscope signals and accelerometer signals follow similar behavior.

### 2.3.2.2 Motion Tracking and Step-Length Measurement

The step-length-measurement algorithm mainly relies on the z-axis gyroscope signals. The accelerometer signal on the x axis also has a minor impact on the motion tracking. On the other hand, sensor misorientation does not affect the sensor signal on the z axis. But the small degradation of step-length-measurement accuracy from sensor misorientation can still be corrected by this algorithm. The result was evaluated over a dataset of 17 traces from five different subjects in total and is shown in Figure 2.5. On average, the correctly placed sensor gives us an accuracy of 97.38%, while the incorrectly placed sensor gives us an accuracy of 95.50%.

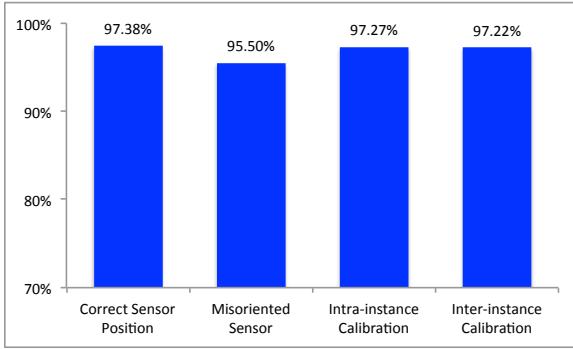


Figure 2.5: Step-length-measurement accuracy for misorientation before and after calibration

With intra-instance calibration, the accuracy can be brought up to 97.27% while inter-instance calibration gives us an accuracy of 97.22%.

### 2.3.3 Rotational Displacement

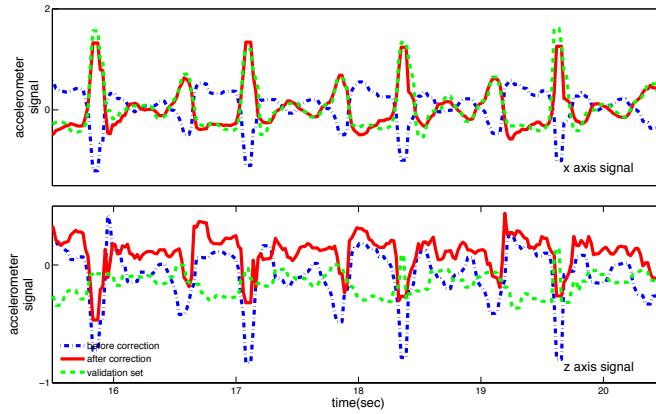
#### 2.3.3.1 Signal Recovery Result

In rotational displacements, the main distortion comes from the rotation of the x-z plane. In this work, we ignore position difference. As is shown in Figure 2.6, the x-axis accelerometer and z-axis gyroscope signals match the validation set very well. z-axis accelerometer and x-axis gyroscope signals, however, still have room for improvement. The main reason is in MMSE estimation of the rotation angle, the total absolute error was minimized, so it has a bias against components with smaller amplitude.

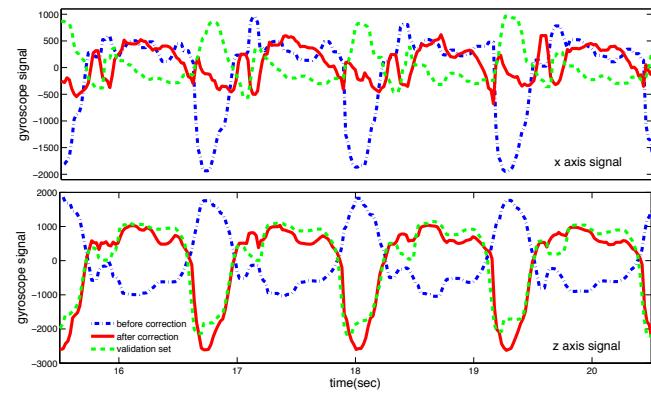
#### 2.3.3.2 Motion Tracking and Step-Length Measurement

In the rotational displacement case, both the x-axis and z-axis signals are distorted. Thus, the motion tracking and the step-length-measurement performance are strongly affected by this type of sensor misplacement.

Figure 2.7 shows three of the most important phases in one gait cycle of the

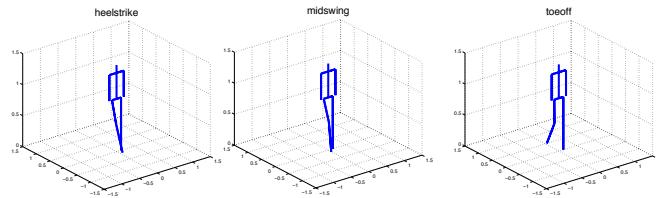


(a) accelerometer signal

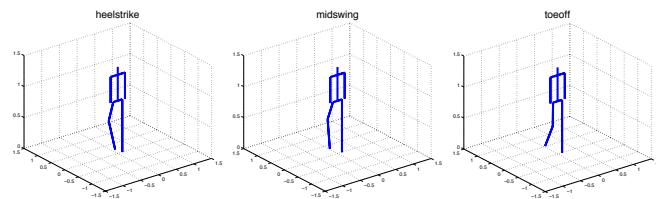


(b) gyroscope signal

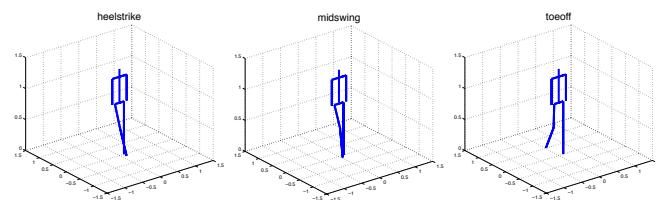
Figure 2.6: Signal recovery for rotational displacement



(a) Correct Sensor Placement



(b) Incorrect Sensor Placement



(c) Incorrect Sensor Placement After Calibration

Figure 2.7: Key Walking Phases of Motion Reconstruction

lower body motion tracking generated by the kinematic chain model [WCX13b]. The phases of interest are toe-off, mid-stance, and heel-strike; toe-off and heel-strike are used to estimate step length. Figure 2.7a shows the result of a motion reconstruction based on the four correctly placed sensors. If the sensor on the thigh suffers from rotational displacement, the motion tracking can be strongly affected (Figure 2.7b). After the calibration process discussed above, however, the motion tracking can be repaired (Figure 2.7c).

Instead of misplacing the sensor randomly, we intentionally placed the sensors such that the x-z plane is rotated  $45^\circ$ ,  $90^\circ$ , or  $180^\circ$ . This makes it possible for us to examine how rotational displacement degrades the motion tracking and how well our algorithm recovers the results.

In order to study how rotational displacement affects step-length-measurement performance, 40 walking traces of seven different subjects were collected. Table 1 summarizes the measurement accuracy before and after calibration. When a sensor suffers from a minor rotational displacement (e.g.  $45^\circ$ ), the sensor signals are not distorted severely. Not surprisingly, the calibration does not improve the accuracy. However, as the rotational displacement increases (e.g. in the  $90^\circ$  or  $180^\circ$  case), it strongly degrades the step-length-measurement performance. Such degradation can be effectively corrected by both the intra-instance and inter-instance calibration methods. Notice that the inter-instance calibration method has a slightly weaker performance than intra-instance calibration since choosing templates and synchronizing them is challenging and invites more errors.

## 2.4 Discussions

In this chapter, a new calibration process for sensor misplacement based on walking was developed. A rotation matrix model was implemented. We mainly focused on two sensor misplacement cases: misorientation and rotational displace-

	Correct Sensor Position	Incorrect Sensor Position	Intra-instance Calibration	Inter-instance Calibration
45°	96.22%	95.97%	96.03%	92.37%
90°	95.93%	83.90%	95.23%	94.37%
180°	96.76%	45.84%	95.39%	94.51%

Table 2.1: Step-Length-Measurement Accuracy

ment. Their signal-recovery results and step-length-measurement accuracy were studied. For the case of misorientation, signal recovery works perfectly since the rotation matrix model describes the problem accurately. Since the step-length-measurement algorithm mainly relies on gyroscope signals on the z axis, the performance was not strongly degraded(from 97.38% to 95.50%) with misorientation. After the calibration process, the accuracy is 97.22%. In the rotational displacement case, however, signal recovery does not match the correct sensor measurements exactly. Nonetheless, the strong degradation of step-length-measurement accuracy due to rotational displacement (from 96.76% to 45.84%) could be brought up to 94.51%.

In the next chapter, the third case, sensor linear misplacement, will be discussed in more detail. We will examine how to estimate the sensor location on the ankle by combining the usage of both accelerometers and gyroscopes and how to use the sensor location information to achieve lower body motion tracking with only one sensor on the ankle.

## CHAPTER 3

# Sensor Placement Error Correction And the Application to Motion Tracking

### 3.1 Problem Statement

One of the most widely used current state-of-the-art lower body motion tracking algorithm is the Zero-Velocity-UPdate (ZUPT) [PS10] [CNH12]. In the ZUPT algorithm, motion accelerations are integrated twice to calculate the traveling distance. In order to compensate the cumulative errors due to double integration, stance phase of each gait was detected and the velocity during the stance phase period was reset to zero. Integrating the compensated velocity will produce good traveling distance estimation.

This method has been widely used for pedestrian localization because of its robustness. But mounting sensors on the foot is considered not feasible for the medical context due to the fact that patient might not wear shoes. Therefore, in this chapter, a novel non-ZUPT method is proposed. In this non-ZUPT method, the velocity signals are reset to estimated values instead of zeros at the stance-phase of each stride. This stance-phase velocity estimation was calculated using gyroscope measurements and sensor position information. A training process with short walking was deployed to avoid the manual measurement of sensor position each time the sensor is mounted.

The rest of this chapter is structured as follows. After a quick review of the widely used ZUPT method for foot-mounted sensor trajectory estimation,

detailed discussion of the novel non-ZUPT method will be described in section 3.2. In section 3.3, experimental design and results are reported. In section 3.4, conclusions and suggestions for future work are presented.

## 3.2 Methodology

### 3.2.1 Experimental Instrumentation

Invensense Motion SDK sensors produce 3D accelerometer measurements ( ${}^s\mathbf{a}_t$ ), 3D gyroscope measurements ( ${}^s\boldsymbol{\omega}_t$ ), and filtered orientation information in quaternion representation ( ${}^i\mathbf{q}_t$ ) with 200Hz sampling rate. The right subscript t represents a sample at time t; the left superscript s represents the measurement in the sensor frame; the left superscript i of the quaternion represents the orientation of the sensor with respect to the initial frame when the sensor was powered on. Data were transmitted through the on-board Bluetooth to a local PC.

Two sensors were mounted on the left ankle and the left foot (Figure 3.1) of the subject. For the ankle-mounted sensor, performance of the traditional ZUPT and newly-proposed non-ZUPT methods will be compared. In addition, a performance reference is provided by a foot-mounted sensor using the ZUPT method.

### 3.2.2 Data Preprocessing

All data collected are preprocessed by the method described in this section. Before the motion starts, a short stable period was required. The sensor frame during this period was considered to be the global reference frame. The average of the accelerometer measurements and quaternions in this period are denoted as  ${}^g\mathbf{a}_0$  and  ${}^g\mathbf{q}_0$ . The quaternions at any time t can be projected onto the global reference



Figure 3.1: Experimental Setup

frame with Equation 3.1.

$${}^g\mathbf{q}_t = ({}^g\mathbf{q}_0)^{-1} \times ({}^i\mathbf{q}_t) \quad (3.1)$$

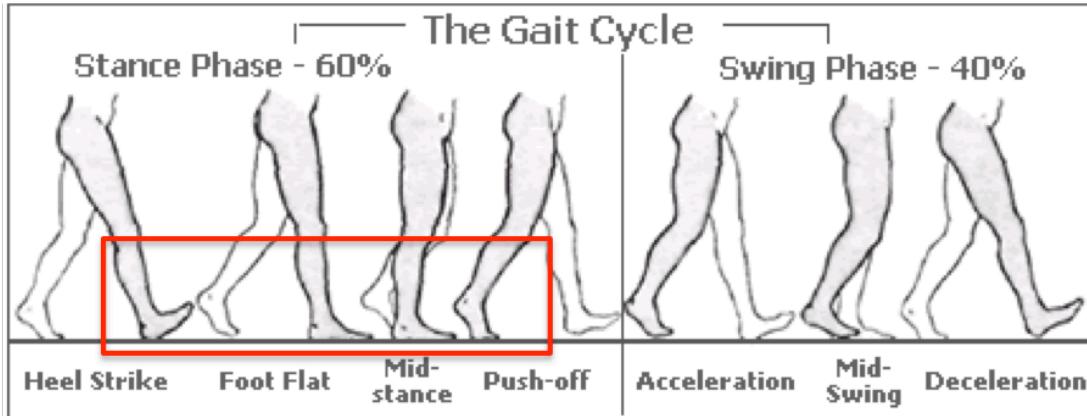
Using quaternions in the global reference frame, the accelerometer data can be projected onto the global frame with Equation 3.2, where  $\overline{{}^g\mathbf{q}_t}$  is the quaternion conjugate of  ${}^g\mathbf{q}_t$ .

$$\begin{bmatrix} 0 \\ {}^g\mathbf{a}_t \end{bmatrix}^T = {}^g\mathbf{q}_t \times \begin{bmatrix} 0 \\ {}^s\mathbf{a}_t \end{bmatrix}^T \times \overline{{}^g\mathbf{q}_t} \quad (3.2)$$

Since  ${}^g\mathbf{a}_0$  is a good estimate of gravity in the global frame  ${}^g\mathbf{G}$ , pure motion acceleration in the global frame  ${}^g\mathbf{a}_t^{Motion}$  can be calculated by subtracting  ${}^g\mathbf{a}_0$  from  ${}^g\mathbf{a}_t$  (Equation 3.3).

$$\begin{aligned} {}^g\mathbf{a}_t^{Motion} &= {}^g\mathbf{a}_t - {}^g\mathbf{G} \\ &= {}^g\mathbf{a}_t - {}^g\mathbf{a}_0 \end{aligned} \quad (3.3)$$

This preprocessing procedure gives motion acceleration  ${}^g\mathbf{a}_t^{Motion}$  and sensor orientation with respect to the global frame in quaternion  ${}^g\mathbf{q}_t$ .



[www.aaronswansonpt.com](http://www.aaronswansonpt.com)

Figure 3.2: Gait Segmentation

### 3.2.3 Foot-sensor Gait Reconstruction Using ZUPT

#### 3.2.3.1 stance phase detection

Figure 3.2 shows the complete gait cycle of a healthy adult. At the middle of each stance-phase, the foot is stable and flat on the ground. These time stamps should be detected before velocity update can be implemented.

A detection method similar to the Acceleration Magnitude Detector in [WXX13] is applied. The average of the motion acceleration energy  $\|{}^g\mathbf{a}_t^{Motion}\|^2$  in a sliding window (length 0.1s) is evaluated. By selecting the proper threshold ( $0.025m^2/s^4$ ) windows of stance phase are detected. The mid-points of these stance-phase windows are denoted as  $ST_i$ , where index i means the stance-phase of the  $i^{th}$  stride in the whole walking process.

#### 3.2.3.2 velocity update and trajectory estimation

The raw velocity signal ( ${}^g\mathbf{v}_t^{raw}$ ) calculated by integrating  ${}^g\mathbf{a}_t^{Motion}$  (Equation 3.4) has cumulative error because of the noise in the accelerometer. This cumulative error is corrected by resetting  ${}^g\mathbf{v}_t^{raw}$  to zero at  $ST_i$ . Then, a reliable trajectory can be calculated by integrating this corrected velocity ( ${}^g\mathbf{v}_t^{cor}$ ).

$${}^g\mathbf{v}_t^{raw} = \int_0^t {}^g\mathbf{a}_{\tau}^{Motion} d\tau \quad (3.4)$$

### 3.2.4 Ankle-sensor Gait Reconstruction Using Non-ZUPT

For ankle-mounted sensors, the velocity at the middle of the stance phase in each stride is not zero. This velocity can be estimated, and a similar non-ZUPT correction can be applied.

#### 3.2.4.1 Stance Phase Detection

The average of the motion acceleration energy  $\|{}^g\boldsymbol{\omega}_t^{Motion}\|^2$  in a sliding window (length 0.1s) is evaluated. By selecting the proper threshold ( $2 \text{ rad}^2/\text{s}^2$ ) , coarse windows of stance phase are detected. Within each of these coarse windows, a finer window of length 0.05s with the smallest gyroscope energy variance is selected. The starting and ending of this finer window are denoted as  $ST_{i1}$  and  $ST_{i2}$ , and the center of these fine windows are denoted as  $ST_i$  ( $i = 1\dots n$ ,  $n$  is the total number of strides).

#### 3.2.4.2 Velocity Update and Trajectory Estimation

During the stance-phase of each gait cycle, the calf is rotating around the heel (Figure 3.2). Hence, for an ankle-mounted sensor, the angular velocity  ${}^s\boldsymbol{\omega}_t$  from gyroscope signals, velocity  ${}^s\mathbf{v}_t$  and the rotation moment arm  $\mathbf{r}$  (determined by the sensor position) are related by Equation 3.5. The velocity is solved in the sensor frame. This is because, for rotation motion, velocity in the global frame is constantly changing in direction while velocity in the sensor frame is relatively constant.

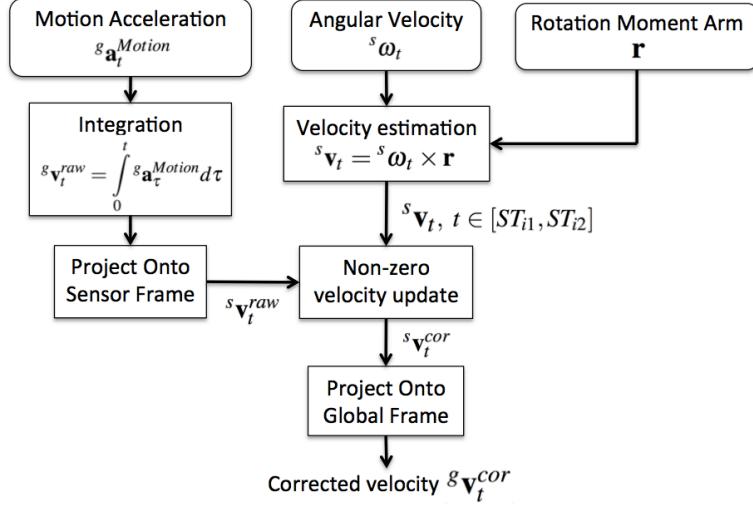


Figure 3.3: System Block Diagram For Trajectory Reconstruction

$${}^s \mathbf{v}_t = {}^s \boldsymbol{\omega}_t \times \mathbf{r}, \quad , t \in [ST_{i1}, ST_{i2}] \quad (3.5)$$

Figure 3.3 shows the system block diagram for the trajectory reconstruction. We start with integrating the pure motion acceleration  ${}^g \mathbf{a}_t^{Motion}$  and calculating the raw velocity signal  ${}^g \mathbf{v}_t^{raw}$  (Equation 3.4). The velocity is then projected into the sensor frame using quaternion information (Equation 3.6).

$$\begin{bmatrix} 0 \\ {}^s \mathbf{v}_t^{raw} \end{bmatrix}^T = \overline{{}^g \mathbf{q}_t} \times \begin{bmatrix} 0 \\ {}^g \mathbf{v}_t^{raw} \end{bmatrix}^T \times {}^g \mathbf{q}_t \quad (3.6)$$

The raw velocity signals  ${}^s \mathbf{v}_t^{raw}$  are reset to  ${}^s \mathbf{v}_{ST_i}$  at the resetting point  $ST_i$  in each stride. This corrected velocity  ${}^s \mathbf{v}_t^{cor}$  will be projected back into the global frame (Equation 3.7). Further integration of  ${}^g \mathbf{v}_t^{cor}$  will give us the trajectory estimation.

$$\begin{bmatrix} 0 \\ {}^g \mathbf{v}_t^{cor} \end{bmatrix}^T = {}^g \mathbf{q}_t \times \begin{bmatrix} 0 \\ {}^s \mathbf{v}_t^{cor} \end{bmatrix}^T \times \overline{{}^g \mathbf{q}_t} \quad (3.7)$$

### 3.2.4.3 Sensor Placement Estimation

This non-ZUPT algorithm works only if we know the rotation moment arm  $\mathbf{r}$  in Equation 3.5. A detailed method of estimating  $\mathbf{r}$  in a training process will be described below.

In a training process, the subject was asked to do a 3-meter-level short walking on a flat floor. The raw data was preprocessed with the same method described in section II-B. The system block diagram for sensor position estimation is shown in Figure 3.4.

Similarly, motion acceleration  ${}^g\mathbf{a}_t^{Motion}$  was first integrated to calculate velocity  ${}^g\mathbf{v}_t^{raw}$  (Equation 3.4). Then,  ${}^g\mathbf{v}_t^{raw}$  was projected onto sensor frame  ${}^s\mathbf{v}_t^{raw}$  with Equation 3.6. Because we keep the training process short (3-meter-level walking), resetting the velocity to zero at the starting and ending will correct the cumulative error, and  ${}^g\mathbf{v}_t^{cor}$  was calculated. The stance-phase window was detected with the method described in section II-D-(1). Finally, the rotation moment arm can be estimated by MMSE using the sampling points over stance-phase windows in Equation 3.8.

$$\mathbf{r} = \arg \min_{\mathbf{r}} \sum_t \| {}^s\mathbf{v}_t^{cor} - {}^s\boldsymbol{\omega}_t \times \mathbf{r} \|, \quad t \in [ST_{i1}, ST_{i2}] \quad (3.8)$$

Since the cross product between two three-element column vectors can always be reformulated as product of a matrix and a vector, Equation 3.8 can be easily solved in normal equation form.

## 3.3 Experiments and Results

### 3.3.1 Experimental Procedure

10 subjects were recruited for applying the non-ZUPT algorithm to flat-floor walking. First, in the training process, the subject was asked to perform a 3-meter-level

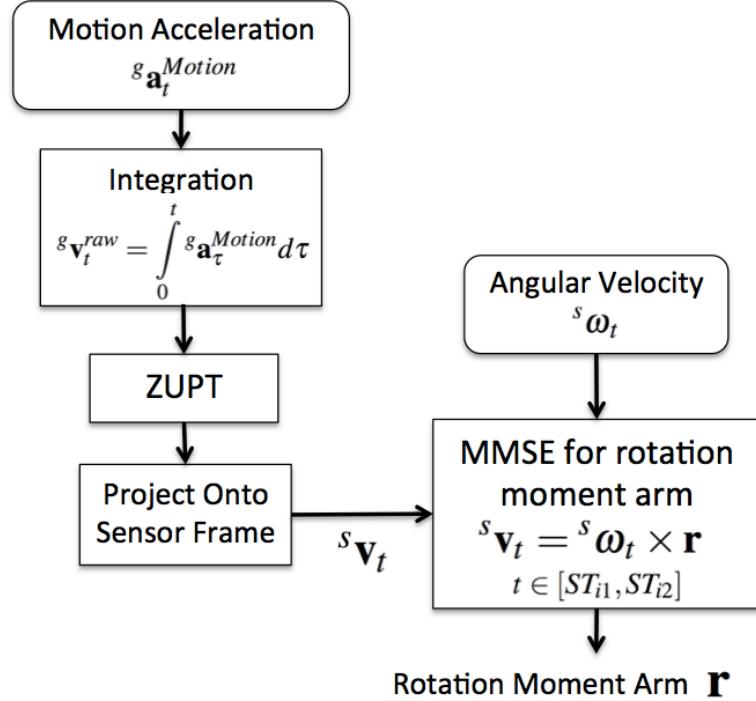


Figure 3.4: System Block Diagram For Sensor Position Estimation

walking (the walking length does not need to be exact). Sensor position information was estimated with the method described in section II-D-(3). Second, in the testing process, the subject performs two sets of 40-meter-level walking on the ruled floor.

In addition, 4 subjects were recruited for applying the non-ZUPT algorithm to stairs walking. The training process is performed on flat floor as well and is consistent with the above. In the testing process, however, the subjects were asked to walk on stairs.

Total walking distance estimations ( $l^{est}$ ) from both the ZUPT and non-ZUPT trajectory reconstruction algorithms were compared to the ground truth ( $l^{gt}$ ) collected during the experiments. Estimation errors were calculated in Equation 3.9 and reported in Table 3.1.

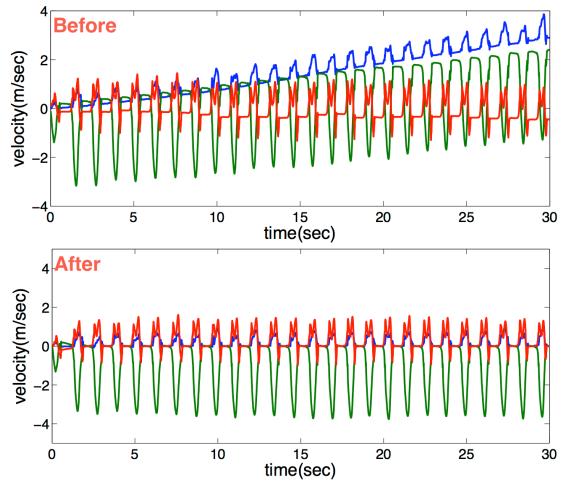
$$err = \frac{|l^{est} - l^{gt}|}{l^{gt}} \quad (3.9)$$

### 3.3.2 Results and Analysis

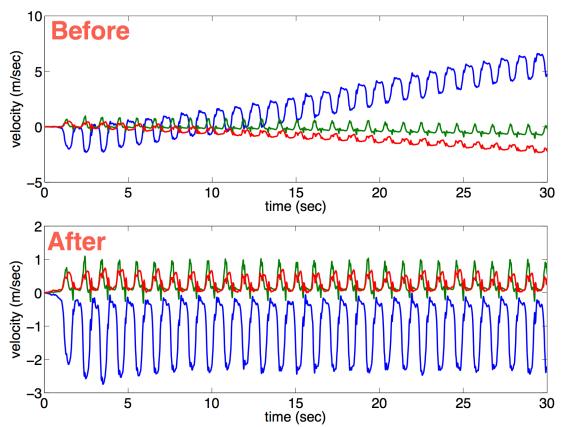
Figure 3.5 shows the velocity estimation before and after drift correction under the ZUPT algorithm and non-ZUPT algorithm. Note that only a short interval within a long walking was shown in the figure for clear visualization. Before any velocity correction algorithm, integrating the accelerations produces drifted velocity signals. The ZUPT algorithm detects stance-phases in each gait cycle and resets velocity there to be zero (Figure 3.5a). For the new non-ZUPT algorithm, however, we would reset the velocity at each stance-phase to an estimated value based on the estimated sensor position and gyroscope signals (Figure 3.5b).

Table 3.1 shows the distance reconstruction error for different algorithms over ten subjects. Not surprisingly, the traditional ZUPT algorithm gives satisfactory trajectory estimation with 2.33% error. On the other hand, applying the ZUPT algorithm on the ankle-mounted sensor (Table 3.1 column 1) produces poor results with 13.22% error. Note that applying ZUPT to ankle-mounted sensors always under-estimates the total walking distance. This is because at each stance-phase, the sensor on the ankle has positive velocity along the walking direction. Forcing them to zero with the ZUPT algorithm makes the velocity smaller than the actual values. Integrating the under-estimated velocity signals will have a cumulative effect on the distance estimation results. This systematic error can be corrected by the newly proposed non-ZUPT algorithm. The non-ZUPT algorithm estimates the actual velocity value at each stance-phase, making the trajectory estimation more accurate. Table 3.1 column 2 shows that distance estimation error was reduced to 3.58%.

Table 3.2 shows the distance estimation error for walking upstairs and downstairs. For upstairs walking, distance estimation error was reduced to 3.61% using



(a) ZUPT for a foot-mounted sensor



(b) non-ZUPT for ankle-mounted sensor

Figure 3.5: Velocity Update Before and After Using ZUPT And Non-ZUPT Method

Subject number	Ankle sensor (ZUPT)	Ankle sensor (non-ZUPT)	Foot sensor (ZUPT)
1	30.41%	3.63%	2.88%
2	4.24%	3.53%	0.32%
3	17.71%	6.04%	0.84%
4	16.89%	1.70%	4.00%
5	5.17%	5.44%	2.57%
6	9.17%	5.43%	0.37%
7	11.53%	4.76%	1.83%
8	16.01%	1.24%	2.54%
9	9.71%	2.22%	4.61%
10	13.55%	1.32%	3.36%
Avg.	<b>13.44%</b>	<b>3.58%</b>	<b>2.33%</b>

Table 3.1: Motion Tracking Error for Flat Floor Walking

the non-ZUPT method from 24.65% using the ZUPT method. In addition, the ZUPT method is always over-estimating the walking distance. This is because, the ankle has a negative velocity at the stance-phase when walking upstairs. Forcing them to zero over-estimates the velocity, hence, over-estimates the walking distance. For downstairs walking, the non-ZUPT method reduces estimation error significantly only for subject No.4. For 3 other subjects, because there exists a short zero-velocity interval for each stance-phase even for the ankle-mounted sensor, performance of the ZUPT method is also acceptable. Overall, the non-ZUPT method, without the zero-velocity interval assumption is more robust over different walking styles.

Subject number		Ankle sensor (ZUPT)	Ankle sensor (non-ZUPT)	Foot sensor (ZUPT)
Upstairs	1	25.57%	7.00%	8.29%
	2	31.04%	1.75%	3.34%
	3	23.18%	2.44%	8.07%
	4	18.81%	3.23%	2.94%
<b>Avg.</b>		<b>24.65%</b>	<b>3.61%</b>	<b>5.66%</b>
Downstairs	1	4.60%	1.06%	4.44%
	2	4.50%	4.53%	4.32%
	3	2.96%	2.80%	14.14%
	4	20.51%	7.83%	14.82%
<b>Avg.</b>		<b>8.14%</b>	<b>4.06%</b>	<b>9.43%</b>

Table 3.2: Motion Tracking Error For Stairs Walking

### 3.4 Discussions

In this chapter, a new non-ZUPT method was developed. This method makes single-sensor lower-body motion tracking possible for the medically more preferable ankle-mounted sensors. We use the fact that during the stance-phase, the ankle-mounted sensor is rotating around the heel. Thus, the velocity of the ankle-mounted sensor in this period can be estimated using angular velocity and the estimated rotation moment arm. The rotation moment arm is acquired through a training process with a short walking period. With this easily available training activity, we avoid the need of measuring the sensor position manually each time the sensor is mounted. Resetting velocity at the stance-phase to the estimated velocity compensates the cumulative error and estimates the final trajectory correctly.

With this method, ankle-mounted distance estimation error was reduced to 3.58% on average, compared to 13.22% on average using the conventional ZUPT method for flat floor walking. For upstairs walking, estimation error was reduced to 3.61% from 24.65%, while for downstairs walking, estimation error was reduced to 4.06% from 8.14%.

In chapters 2 and 3, we discussed the opportunistic calibration method for both sensor orientation and position information using walking activities (the most common daily activity) to achieve accurate lower-body motion tracking and step length estimation.

These methods find the true position and orientation of the sensors to enable good estimation of lower body movement. In the following chapter, however, we will focus on methods that work with no knowledge of the true sensor mounting placement. These conveniences can make our motion tracking systems more robust to real world conditions.

## CHAPTER 4

# A Robust Motion Tracking Algorithm regardless of Sensor Placement and Orientation

### 4.1 Problem Statement

The Non-ZUPT method makes the single sensor lower body motion tracking possible for the medically preferred ankle mounted sensor location. But this method relies on a training process to determine the exact location of the sensor on one's ankle, and can be vulnerable to location change of the sensor throughout the day.

Therefore, in this chapter, we propose a new step length estimation method called the Pose Invariant (PI) method that uses the quaternion output of the motion sensor and estimates the traveling distance within each step by multiplying the leg length and the sine of the leg's orientation change within each step. This method requires only one training session per clinical trial and maintains robust performance regardless of sensor mounting position change. In addition, we are only interested in the orientation differences between the beginning and end of each stance. This makes it robust against the shaky movement in the middle of the motion, since the nature of this model is differential instead of cumulative.

This chapter provides the following contributions: 1) a more accurate gait segmentation method using the sensor orientation signal instead of detecting the peak in the accelerometer norm and 2) an inverted pendulum model based step-length estimation method. Experiments showing its accuracy and robustness will be described.

## 4.2 Methodology

In this section, we will start from human gait modeling and segmentation. Then the step length estimation algorithm using this model will be discussed in subsection 4.2.2. Since this algorithm takes subject leg length as an input, we will discuss how this can be estimated by a once-per-clinical-trial training process in subsection 4.2.3.

### 4.2.1 Inverted Pendulum Model

Human walking is usually segmented into strides that start with a heel-strike and end with the next heel-strike on the same foot. A stride can then be divided into a stance phase when one foot is in contact with the ground, and a swing phase when the same foot is not in contact with the ground. During the stance phase, the total body center-of-mass trajectory can be approximately modeled as an inverted pendulum (Figure 4.1).

Observing that each swing phase of the left foot corresponds to a stance phase of the right foot and each swing phase of the right foot corresponds to a stance phase of the left foot, the following segmentation was adopted in this chapter. We segment human gaits into left inverted pendulum (denoted as  $InvP^L$ ) and right inverted pendulum (denoted as  $InvP^R$ ) periods. An  $InvP^L$  starts with a left-foot heel-strike and ends with the next right-foot heel-strike. Similarly, an  $InvP^R$  starts with a right-foot heel-strike and ends with the next left-foot heel-strike. A walking motion moves on with the  $InvP^L$ 's and  $InvP^R$ 's repeating themselves alternately (Figure 4.2).

Thus, within each  $InvP^L$  or  $InvP^R$ , the center-of-mass travel distance can be estimated given lower limb length and the angle change during this period (Figure 4.1, Equation 4.1).

$$SL = 2L \sin(\alpha/2) \quad (4.1)$$

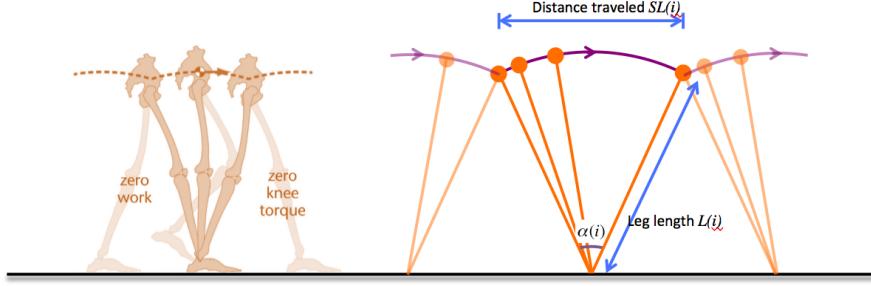


Figure 4.1: The Inverted Pendulum Model for Human Walk (Figure adapted by author from [Kuo10] and [SR06])

Note that the center-of-mass travel distance within each *InvP* stays the same regardless of the exact trajectory in between. Equation 4.1 holds as long as the legs are close to straight at heel-strokes.

#### 4.2.2 Step Length Estimation - the testing phase

Traditionally, human walking segmentation starts with heel-strike detection by applying peak detection algorithms to the accelerometer norm. However, in practice, such methods suffer accuracy issues, especially when the subject's stepping frequency or heel-strike intensity keeps changing. Figure 4.3a demonstrates one example of heel-strike detection error near the 14 second timestamp.

In this chapter, we adopted a more robust gait segmentation method by applying peak detection to orientation information instead. Since the InvenSense motion sensors provide sensor orientation information in quaternions, the angle between the sensor's y-axis and gravity can be calculated. Applying a peak detection algorithm to this angle signal yields robust toe-off timestamps (Figure 4.3b). Then, the heel-strike moments can be detected by finding one peak within each period divided by those toe-off timestamps (Figure 4.3c). Note that the three subfigures of Figure 4.3 come from the same data collection event; the detection result in Figure 4.3c is accurate, and the results in Figure 4.3a suffer several errors

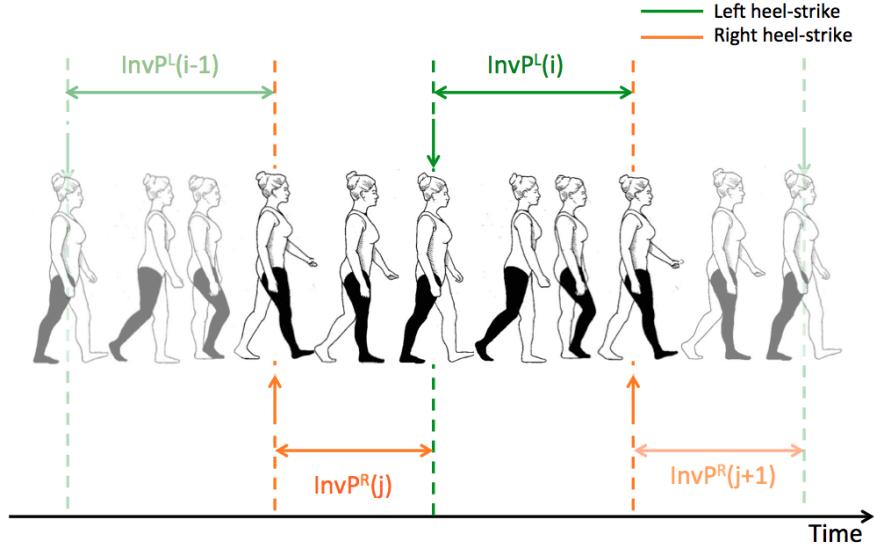
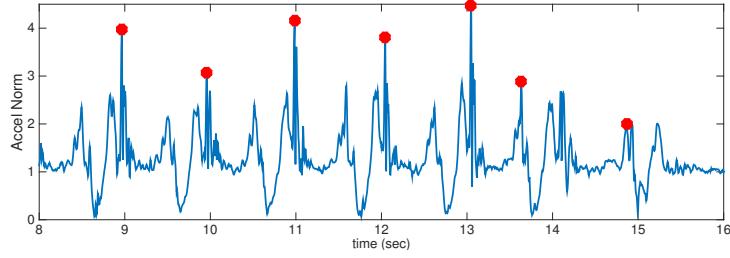


Figure 4.2: The Gait Segmentation Adopted in this Method (Figure adapted by author from [LHW07])

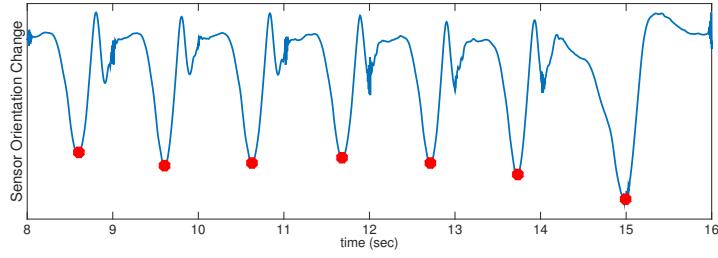
near the 14 second timestamp.

The detected heel-strike moments are denoted as  $HS^L(i)$  and  $HS^R(j)$ , where  $i$  and  $j$  represent the  $i^{th}$  and  $j^{th}$  heel-strike for the left ankle sensor and the right ankle sensor respectively. Figure 4.4 shows one example of the heel-strike detection results for both left and right ankle sensors for the same data collection event. Then, for each left-side heel-strike timestamp  $HS^L(i)$ , we get the smallest right-side heel-strike timestamp larger than  $HS^L(i)$ , and denote it as  $HS^R(i')$ . The period between  $HS^L(i)$  and  $HS^R(i')$  is denoted as  $InvP^L(i)$ . Similarly, for each right-side heel-strike timestamp  $HS^R(j)$ , we get the smallest left-side heel-strike timestamp larger than  $HS^R(j)$ , and denote it as  $HS^L(j')$ . The period between  $HS^R(j)$  and  $HS^L(j')$  is denoted as  $InvP^R(j)$ . Figure 4.2 illustrates this segmentation.

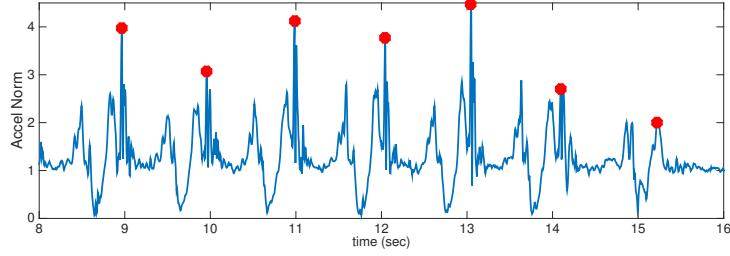
The InvenSense IMU sensors provide filtered sensor orientation information in the form of quaternions. Thus, the orientation change within each  $InvP^L(i)$  and  $InvP^R(j)$ , can be calculated with Equation 4.2 where  $q^L$  and  $q^R$  represents the



(a) Peak Detection for Accelerometer Measurement Norm



(b) Peak Detection for angle between the sensor y-axis with gravity



(c) Heel-strike detected using the results from Figure 4.3b

Figure 4.3: Gait segmentation method comparison

quaternion measurement from the left ankle sensor and the right ankle sensor.

$$\begin{aligned} q_{rot}^L(i) &= q^L(HS^L(i)) \times (q^L(HS^R(i')))^{-1} \\ q_{rot}^R(j) &= q^R(HS^R(j)) \times (q^R(HS^L(j')))^{-1} \end{aligned} \quad (4.2)$$

Because the quaternion has an Euler representation shown in Equation 4.3, the rotation angle can be determined by Equation 4.4, where  $q_{rot}^L(i)[1]$  refers to the first element of the quaternion  $q_{rot}^L(i)$ .

$$q = \cos(\alpha/2) + (u_x\mathbf{i} + u_y\mathbf{j} + u_z\mathbf{k}) \sin(\alpha/2) \quad (4.3)$$

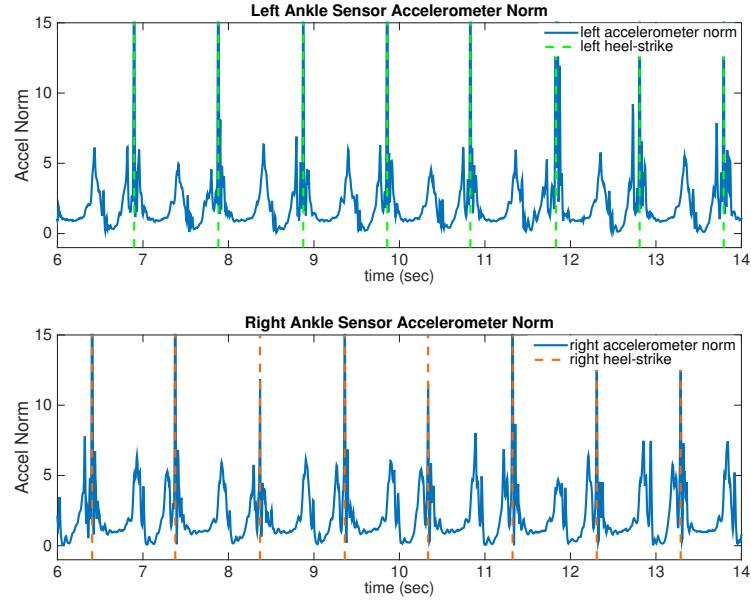


Figure 4.4: Heel-strike Detection Results for Left and Right Ankle Sensors

$$\begin{aligned}\alpha^L(i) &= 2 \cos^{-1}(q_{rot}^L(i)[1]) \\ \alpha^R(j) &= 2 \cos^{-1}(q_{rot}^R(j)[1])\end{aligned}\tag{4.4}$$

Then, the traveling distance within each  $InvP^L$  and  $InvP^R$  can be calculated using Equation 4.5 where L represents the subject's leg length, and  $SL^L(j)$  and  $SL^R(j)$  the traveling distance within each  $InvP^L$  and  $InvP^R$ . Taking the difference of sensor orientations within each step makes this method preferable for accommodating sensor mounting errors. Moreover, it avoids the possible cumulative errors in previous ZUPT or non-ZUPT approaches.

$$\begin{aligned}SL^L(i) &= 2L \sin(\alpha^L(i)/2) \\ SL^R(j) &= 2L \sin(\alpha^R(j)/2)\end{aligned}\tag{4.5}$$

Finally, the total walking distance can be calculated by the summation of all  $SL^L$ 's ad  $SL^R$ 's as is shown in equation 4.6 where n is the total number of  $InvP^L$ 's

and  $m$  is the total number of  $InvP^R$ 's.

$$SL_{total} = \sum_{i=1}^n SL^L(i) + \sum_{j=1}^m SL^R(j) \quad (4.6)$$

One important assumption for this method is that the leg length of the subject is known. We discuss the training phase algorithm for estimating subject leg length in the following section.

#### 4.2.3 Leg Length Estimation - the training phase

In order to estimate leg length, we ask the subject to walk in front of a KINECT2 sensor. The KINECT2 sensor provides 3-dimensional joint position in a timely manner. Given the movement of the “SpineBase” joint during each step and angle change information from the motion sensor, we can estimate leg length using Equation 4.7, where  $SL$ 's are given by KINECT2 sensors and  $\alpha$ 's are derived from IMU quaternion measurements using Equation 4.2 and Equation 4.4. The best leg length estimation was determined as the median of the candidates from the steps captured by the KINECT2 sensor.

$$\begin{aligned} L^L(i) &= SL(i)/(2\sin(\alpha^L(i)/2)) \\ L^R(j) &= SL(j)/(2\sin(\alpha^R(j)/2)) \end{aligned} \quad (4.7)$$

Since only the leg length parameter is estimated during this training phase, it is valid regardless of the motion sensor mounting position and needs to be done only once per clinical trial.

In some clinical trials, sensor data for patients walking for a given length was recorded as a training template. Under such circumstances, since the exact walking distance and the IMU sensor data are available, the leg length can be estimated without the KINECT2 sensor, as we can work back from the total distance and number of strides.

## 4.3 Experiments and results

In this section, the devices used and system set-up will be discussed in detail. Following that, experiment design for data collection is reported. Then we will move on to the results and discussion.

### 4.3.1 System Set-up

Two InvenSense Motion SDK sensors were mounted on the two ankles by Velcro straps (Figure 4.5). The sensors were sampled at 200Hz and the output data consists of raw accelerometer measurements, gyroscope measurements and filtered sensor orientation (in quaternions). Data were transmitted through the on-board Bluetooth to a local PC and synchronized according to the receiving time.

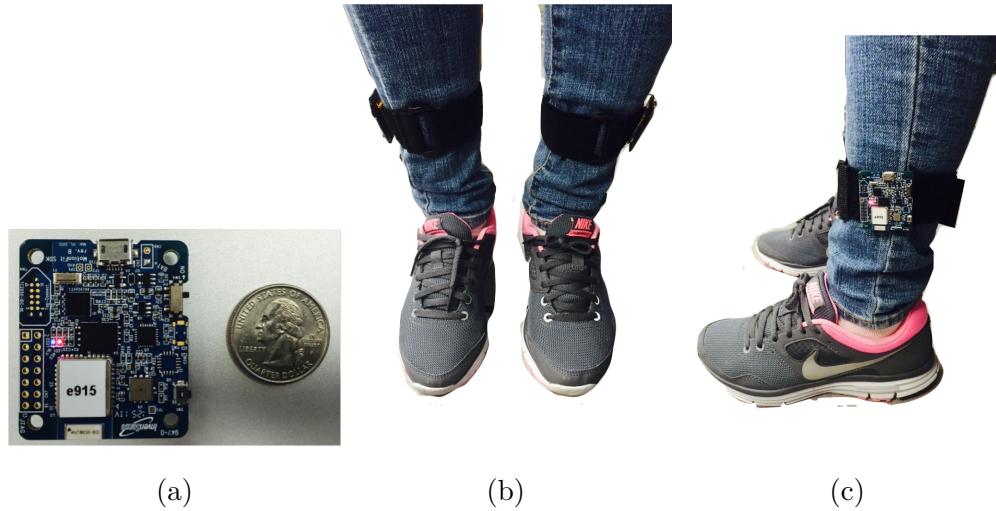


Figure 4.5: The IMU sensors used in this chapter. (a) the sensor size, (b) the front view of sensor mounting, (c) the side view of sensor mounting

In addition, a KINECT2 sensor was used in the training phase for detection of a subject's position. The KINECT2 sensor uses video processing techniques and merges information from a regular RGB camera and an infra-red depth camera to provide 3-dimensional spatial-temporal human skeleton information (Figure 4.6).

We developed a KINECT skeleton data recorder software tool using KINECT SDK v2.0. As is shown in Figure 4.7, users can select the joints to be recorded with the checkboxes. After pressing the “Start Data Collection” button, it will start writing the three dimensional position data as well as the corresponding timestamps to ‘.txt’ files. One ‘.txt’ file will be generated for each selected joint.

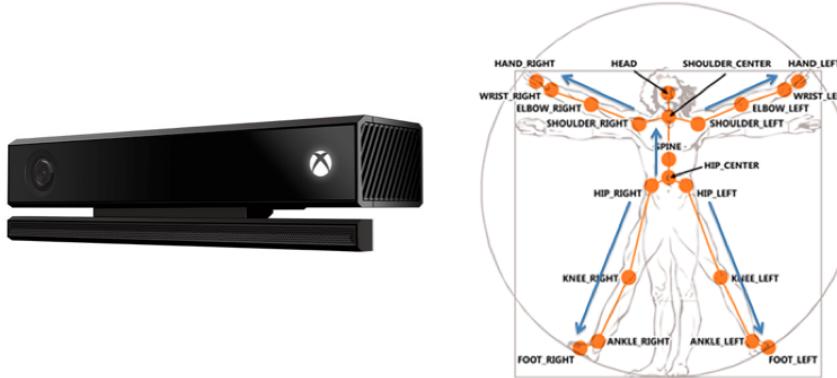


Figure 4.6: The KINECT2 sensor

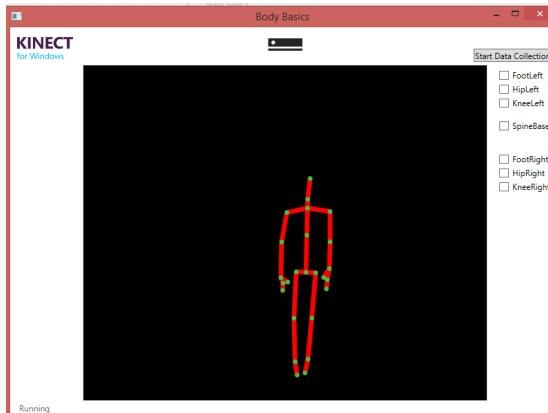


Figure 4.7: The KINECT joints tracking tool

### 4.3.2 Experiment Design

In order to evaluate the performance in estimating the step lengths, we recruited 9 adult subjects for the following experiment. The subjects were asked to mount two IMU sensors, one on each ankle, with velcro straps (Figure 4.5).

In the training phase, the subject was instructed to walk in front of a KINECT2 sensor, during which period, the position data of the “SpineBase” joint from the KINECT2 sensor and motion data from the IMU sensors were recorded. The subject’s leg length was estimated using this information. In the testing phase, we kept the IMU sensors on the ankles and asked the subject to walk for roughly 50 feet on a flat floor. The exact walking distance was measured as the ground truth on a ruled floor. The total walking distance estimated using this algorithm was compared to the ground truth, and the error rate was recorded.

In order to prove the robustness of this method, an additional experiment was designed to compare with the non-ZUPT method. The training phase was exactly the same as described above. The testing phase consisted of 2 trials. The first trial was the same as the testing phase above. In the second trial, we moved the sensor mounting position on the ankle. The subject was asked to walk for 50 feet on a flat floor for both trials. The data from both trials were processed with the two methods respectively.

#### 4.3.3 Results and Analysis

The subject leg length estimated during the training phase as well as the walking distance estimation error rate was reported in Table 4.1. There is a good span of subject leg length. An average error of 3.69% in step length was achieved.

The results for the second experiment are shown in Table 4.2. Trial 1 was conducted right after the training phase, and the sensor position was not changed. In this trial, both methods for all three subjects produced good performance. In trial two, however, the sensor position was intentionally changed, and the non-ZUPT method had a very large error while this new pose-invariant (PI) method estimation error remains low. This is because, for the non-ZUPT method, the sensor position was estimated during the training phase. Since the reset velocities

Subject #	Estimated Leg Length	Absolute Error Rate
1	69.3	3.51%
2	95.7	3.36%
3	79.0	2.26%
4	83.3	6.23%
5	83.9	1.61%
6	91.7	4.54%
7	80.5	5.88%
8	88.9	2.12%
9	69.8	3.70%

Table 4.1: Absolute error rate of the total walking distance estimation

are the cross product of sensor position and the gyroscope signal, changing the sensor position will cause wrong reset velocities, thus making the final trajectory estimation to drift. For this new PI method, we only estimated the leg length during the training phase, and this information remains valid even after changing the sensor position on the ankle. The step lengths are only related to the change of angles within each step, and this change of angles is invariant with the sensor position.

Prior research has resulted in several methods for step length estimation, one of the most interesting human gait metrics for medical applications. The ZUPT method integrates acceleration and resets the velocity to zero at the stance phase to compensate for the cumulative error. This method requires the sensor to be placed on shoes and that a clear stance phase, where one's foot is stable on the ground, can be detected. The modified non-ZUPT resets the velocity to a nonzero value derived from gyroscope signals. This method allows a medically more preferable ankle-mounted position, but requires an extra training session each time the sensor's position changes.

Subject #		1	2	3
Trial 1	non-ZUPT	2.00%	2.54%	4.57%
	PI method	6.00%	0.11%	2.23%
Trial 2	non-ZUPT	9.27%	7.35%	16.10%
	PI method	4.68%	0.46%	1.80%

Table 4.2: Performance comparison of the two methods with Trial 1 (same sensor position with training section) and Trial 2 (different sensor position with training section). The Pose Invariant (PI) method is more robust with either trial while non-ZUPT only works well under Trial 1

## 4.4 Discussions

In this chapter, a more robust step length estimation method was proposed and evaluated. Using the inverted pendulum model, the traveling distance within each step can be calculated by multiplying the leg length and the sine of the leg's orientation change within each step. The orientation change was obtained by taking the differences of the quaternion measurements from the IMU sensor within each step. Using this differential method instead of the cumulative methods offers several benefits: 1) the orientation change is invariant to the exact sensor position; and 2) the shaky movement within each step commonly seen in impaired gaits won't affect the estimation result. In addition, the training session is needed only once per clinical trial for each subject due to the fact that only the leg length parameter is estimated.

Experiments with 9 subjects walking for 50 feet each shows that on average a 3.69% error rate can be achieved by this method. An additional experiment shows that this method provides more robust measurement against sensor placement change compared to the previous non-ZUPT method.

Now that the robustness to sensor position are discussed and resolved in the

chapter, we will introduce a double-layer classification model which is robust to sensor orientation change throughout the day in the next chapter.

# CHAPTER 5

## Orientation Self-correcting Daily Activity Classification

In this chapter, we propose a double layer walking activity classification model that resolves the sensor orientation robustness issue. The first layer consists of a highly conservative walking detection model using only the accelerometer magnitude which is invariant to sensor orientation. The detected walking beacons from this layer are compared with the training templates to determine the rotation matrix to fix the data due to the erroneous sensor mounting. Then the corrected data are fed into the second layer orientation variant model with more accurate results.

The remainder of this chapter is structured as follows. We start with introducing the hardware devices and system set-up in Section 5.1. Then our newly proposed double layer algorithm design is discussed in Section 5.2 with more details. Following that, in Section 5.3, experimental design as well as classification results are reported and discussed. Finally, we further discuss the results and conclude the chapter with Section 5.4

### 5.1 Problem Statement

In this section, we are going to talk about the current activity recognition system utilization, including the wearable motion sensors, uploading cellphone device and server database.

In this section, the devices used and system set-up will be discussed in detail.

Following that, experiment design for data collection are reported. Then we will move on to the results and discussion.

### 5.1.1 Motion Sensing System Setup

In order to monitor human daily activity, we utilized the motion sensing kit developed by the Wireless Health Institute at UCLA [XBK11] [DTX15]. A motion sensing subject kit consists of two soft Velcro straps (Fig. 5.1a), two WHIMS sensors (Fig. 5.1b), an Android smart-phone (Fig. 5.1c) and a wireless charging station (Fig. 5.1d).

The WHIMS sensors shown in Figure 5.1b integrate the latest micro-mechanical inertial chips within a 31 gram, 1" x 1.5" x 0.5" packages and can operate continuously for over 24 hours. Chipsets on the WHIMS sensor include: an Invensense MPU- 6050 motion sensor, an altimeter, Bluetooth radio, real time clock (RTC), microSD card storage, and wireless recharge. The motion sensor provides 3-axis of accelerometer and 3-axis of gyroscope measurements at 40Hz. In order to use two WHIMS sensors simultaneously, initial time synchronization between the sensors and a hub is performed via Bluetooth and is maintained through the RTC. The RTC drifts only 0.17 seconds per day. The inclusion of the real-time clock and on-board storage ensures that data is still captured and synchronization maintained in the event that Bluetooth is disconnected.

An Android smart phone with pre-installed apps is included in the kit for downloading sensor data from the onboard SD card via Bluetooth connection and then uploading them to a designated server via WiFi or 4G connection.

Two soft Velcro straps are also provided so that the sensors are mounted securely while at the same time comfortably for daily usage. Finally, the wireless charging station makes the charging process during the night easy.



(a) Two Soft Velcro Straps



(b) Two WHIMS Sensors



(c) Android Smartphone



(d) Charging Station

Figure 5.1: Subject Kit Devices

### **5.1.2 Trial Implementation**

In this section, we will describe the current trial process that are undertaking with UCLA Neurology Department.

In recognition of the variations in gait speed and stand and swing symmetry that might occur among individuals, we invite the subjects to the clinics for a short training session before distributing the motion sensor kit. During a training session, the subject performs two 30-feet walks at self-selected slow, casual and fast walking speeds. These walking bouts are collected and labeled using a Bluetooth connected Android smartphone and further uploaded to designated server as training templates.

For the days we wish to monitor one's daily activity, subjects placed one sensor on each ankle in the morning (Fig. 5.2). The soft Velcro strap (Fig. 5.1a) secured each sensor (Fig. 5.1b) proximal to the medial malleolus, flush against the bony tibia. At the end of the day, before the subject went into bed, the sensors were removed and placed on the wireless charging station to recharge. Following that, at 12am midnight, the accelerometer data were transmitted to the smart-phone via Bluetooth connection and then uploaded to the central server at UCLA for secure storage and processing. The data transmission and uploading process are triggered by a smartphone with pre-installed apps with no human interventions needed.

### **5.1.3 Current SIRRACt Performance**

[XBK11] claims to have achieved on average 83.61% precision and 84.38% recall for daily walking activity classifications using the SIRRACt trial platform. However, such performances only hold under the assumption of correct sensor mounting orientations. We have noticed the serious problem of subjects not following instructions when the scale of our study keeps increasing. And such problems are



Figure 5.2: Correct Sensor Placement

especially serious when dealing with patient subjects. This is because elderly patients can be technophobic and neurological diseases are often accompanied with cognitive difficulties. Thus, enabling our system to be adaptive to sensor mounting errors is a very important topic. We will be introducing a newly proposed double layer automatic orientation correction algorithm in the next section.

## 5.2 Methodology

The methodology proposed is to pursue robust classification against sensor orientation error without making a concession on the performance by using only orientation invariant features. This requires a first layer hierarchy that consists of a highly conservative orientation-invariant walking detection classifier. Comparing these walking sections with the training template walking, a rotation matrix representing the sensor orientation error can be calculated. Then, this rotation

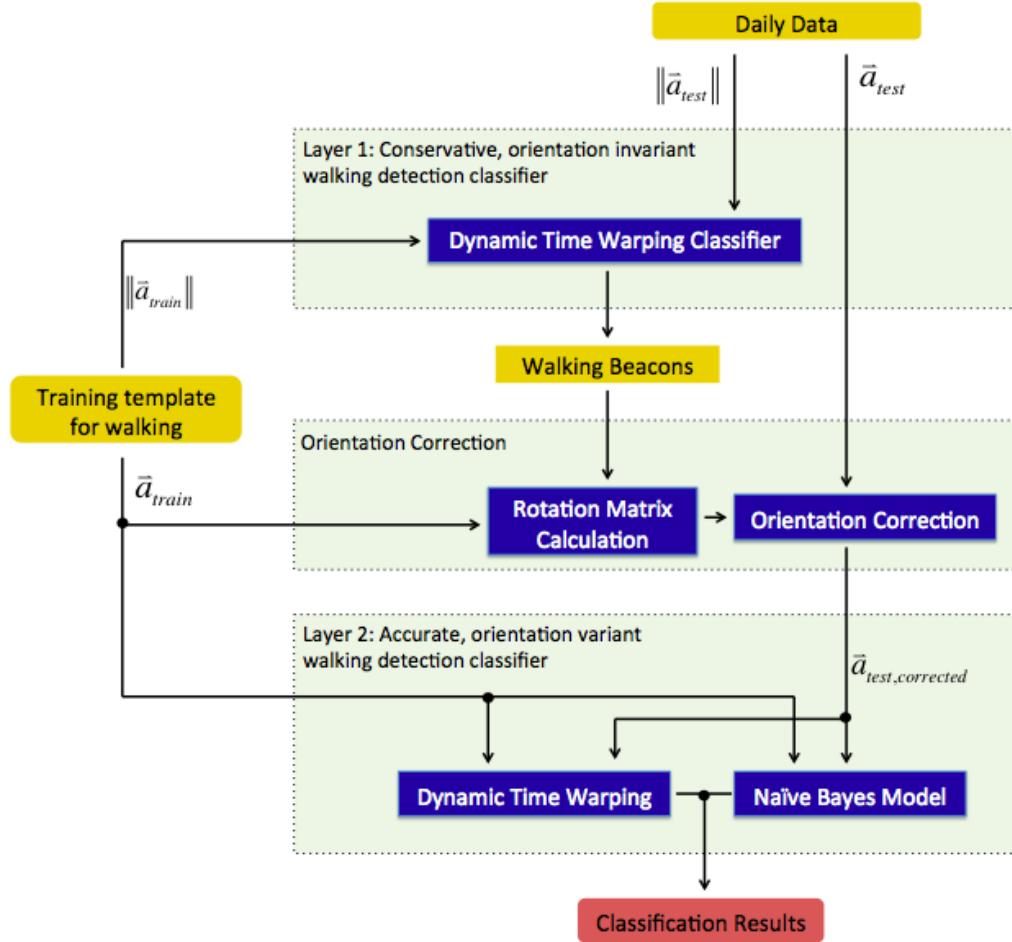


Figure 5.3: System high level description

matrix can be applied to consecutive data assuming that the sensor orientation does not change too frequently. Next, the corrected sensor signal can be used for the more accurate second layer classifier. System high level architecture is shown in Fig. 5.3. We will discuss each of these parts in more detail.

### 5.2.1 Layer 1: a conservative walking detection classifier

The time sequence daily data are divided into overlapping 5 seconds window with 1 second increment. The accelerometer energy signals are calculated. In the first layer hierarchy, accelerometer magnitude of both the training template and testing

data are calculated by Equation 5.1

$$\|\mathbf{a}\| = \sqrt{\mathbf{a}_x^2 + \mathbf{a}_y^2 + \mathbf{a}_z^2} \quad (5.1)$$

Dynamic Time Warping (DTW) distances are calculated between the training walking template accelerometer magnitude and the whole day testing data accelerometer magnitude. Choosing a threshold for DTW distances for walking beacons detection is critical for the following two reasons. First, setting the threshold too low will produce false positive walking beacons and these false positive walking beacons will result in a wrong orientation correction rotation matrix; such errors can damage consecutive sensor signals. Second, setting the threshold too high will leave us with very few walking beacons. In view of the possibility for sensor orientation to change, observation of the sensor orientation using walking beacons from time to time is important. Therefore, we use relative instead of definite thresholding in this chapter. Comparing with the walking template, each sliding window has a DTW distance score, and the lower 1% smallest DTW distance through out the day will be chosen as the threshold. All sliding windows with DTW distance smaller than the threshold will be marked as walking beacon candidates. If there were more than one walking candidate within 5 minutes, we will reserve only the best matching candidate (meaning smallest DTW distance) among the neighbor candidates, and mark it as the walking beacon. These walking beacons will be used in the next step for orientation correction.

The detected walking beacons are marked as stars in Fig. 5.4. Note that only the first 4 hours of the data are shown for the purpose of clear presentation.

### 5.2.2 Orientation Correction

The next step comes with estimating the rotation matrix (i.e. the orientation placement error) comparing the training template and the walking beacons. The

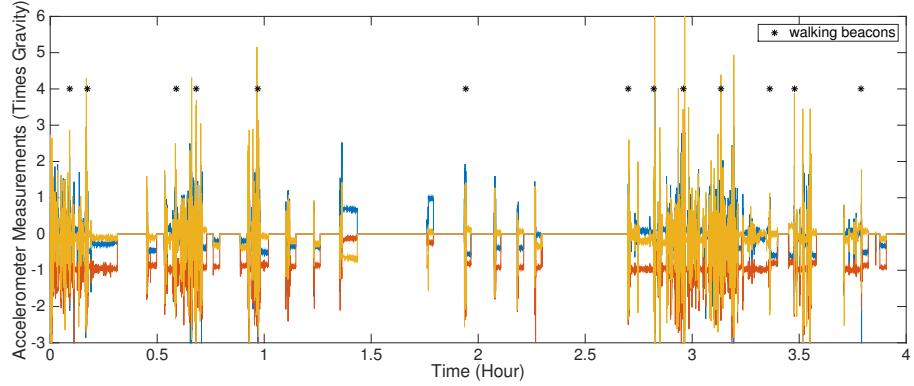


Figure 5.4: Detected Walking Beacons

rotation matrix was estimated by comparing the accelerometer measurement during the stance phase for the training templates and the testing beacons. The reason for choosing the stance-phase section of gait cycle is because the gravity component dominates the accelerometer measurements during this period. The gravity distribution on three axes indicates the sensor orientation.

For the training template, the stance phase was marked manually. This manual process is needed only once for each subject. Stance phase in the testing walking beacons are marked by following the DTW trace, i.e. the matching segment with the training stance phase was marked as stance phase in the testing walking beacon (Fig. 5.5).

We denote the mean of accelerometer measurement in the training template stance phase as  $\mathbf{a}_{train,stance}$ , and the mean of accelerometer measurement in the testing template stance phase as  $\mathbf{a}_{test,stance}$ . Then, the rotation between vectors  $\mathbf{a}_{train,stance}$  and  $\mathbf{a}_{test,stance}$  can be calculated with Equation 5.2 in axis-angle representation.

$$\begin{aligned} \mathbf{r} &= \mathbf{a}_{train,stance} \times \mathbf{a}_{test,stance} \\ \theta &= \mathbf{a}_{train,stance} \cdot \mathbf{a}_{test,stance} \end{aligned} \tag{5.2}$$

For ease of usage, the axis-angle representation of rotation are further transformed into the rotation matrix representation with Equation 5.3, where  $x = \mathbf{r}_x$ ,  $y = \mathbf{r}_y$ ,

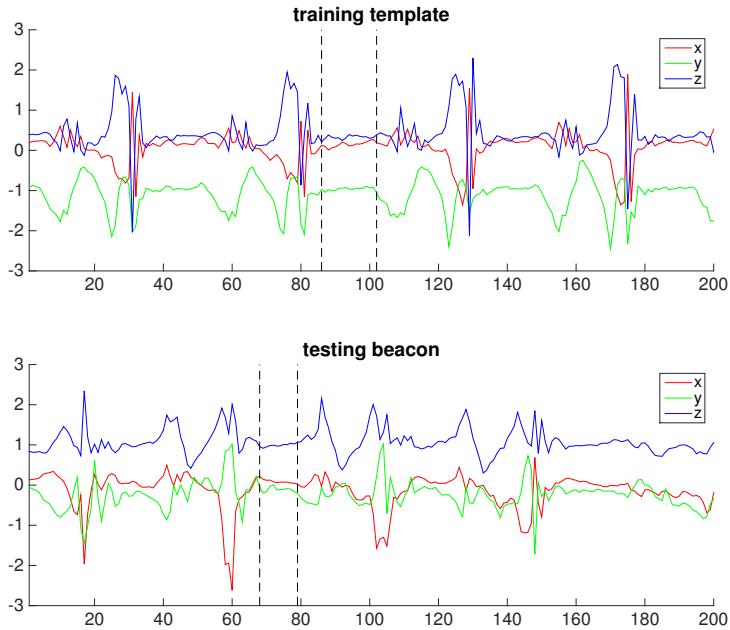


Figure 5.5: DTW orientation segment matching

$$z = \mathbf{r}_z, s = \sin\theta, c = \cos\theta, t = 1 - c.$$

$$\mathbf{R} = \begin{bmatrix} txx + c & txy - zs & txz + ys \\ txy + zs & tyy + c & tyz - xs \\ txz - ys & tyz + xs & tzz + c \end{bmatrix} \quad (5.3)$$

One rotation matrix was calculated using Equation 5.3 for each individual walking beacon. All sensor signals following the current walking beacon would be corrected using this rotation matrix with Equation 5.4 until the next walking beacon was detected.

$$\mathbf{a}_{cor} = \mathbf{a}_{org} \mathbf{R} \quad (5.4)$$

### 5.2.3 Layer 2: a orientation dependent binary classifier

We inherit the classification method described in [XBK11] where Dynamic Time Warping and Naive Bayes classifier outputs were combined to produce binary clas-

Subject ID	Gender	Condition
98	Female	RA patient
117	Female	RA patient
150	Male	healthy
151	Female	healthy
152	Male	healthy
153	Male	healthy
154	Female	healthy
185	Male	healthy
192	Female	healthy

Table 5.1: subject informations

sification results. The Naive Bayes model uses feature set F1 listed in Table 5.3, while the DTW matches 3-axis accelerometer measurements of the testing data with the training template. The results of both models are combined to leverage the robustness of the DTW algorithm against spatiotemporal variations and limited training data with high confidence in the classified results of the NB model. Detailed algorithm description can be found in [XBK11].

## 5.3 Experiments and Results

### 5.3.1 Data Collection and Subject Description

In order to show the validity of this method, we hired 7 healthy subjects and 2 patient subjects with Rheumatoid Arthritis. Table 5.1 shows the subject informations.

In the training data collection phase, each subject was asked to walk through a marked and observed 30-feet section with three different safe speeds selected by

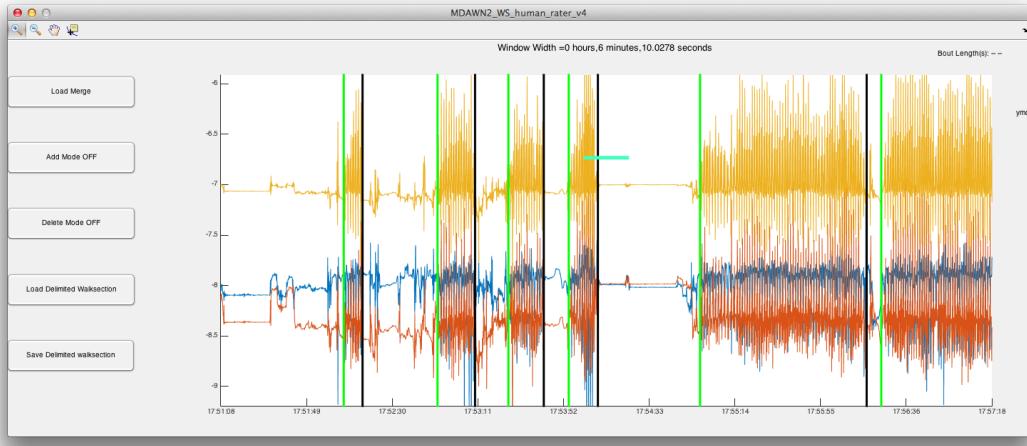


Figure 5.6: Matlab Labeling Tool GUI

subjects (slow, normal and fast). Sensor data were collected and labeled using the Andriod smartphone and uploaded onto a designated server.

Following that, in the testing phase, the subjects were asked to take the kit home, and charge the sensors. From the next morning, the subject can start wearing the sensors during the day. No special instruction was given with respect to sensor mounting orientation. Each subject was asked to carry the sensors for at least two days with at least 6 hours each day.

### 5.3.2 Ground Truth Labeling

Ground truth labeling has always been difficult when dealing with large data set verification for the case of human activity classification. In order to tackle this problem, we designed a Matlab GUI tool that makes manual labeling easier (Fig. 5.6).

First, the accelerometer data was loaded and shown in the window by clicking the “Load Merge” button and selecting the proper data file. Then, users can start marking each walking bout by clicking “Add Mode On” button and sliding the bouts delimiter. Each bout is marked by a green vertical line as the starting and

a black vertical line as the ending. After all the bouts are correctly labeled, the delimiter locations can be saved to local files by clicking “Save Delimited Walk Section”. These walking bouts can then be transformed into classification results with the following criteria. We are using 1 second increment, 5 second window length for the classifiers, therefore, we transform the walking bout such that, if more than half of the window lies within one bout, this window is marked as positive.

The whole day labeling result was collected and compared with the classification result. Since we are building a binary classifier in this paper, precision and recall are recorded for each day when the data are collected.

### 5.3.3 Experiment Design

In order to show the validity of this model, three types of algorithms were evaluated and compared, precision, recall for each of which were recorded.

Algorithm A directly inherit the classifier in [XBK11] where orientation-variant features are used for NB model and 3-axis accelerometer measurements are used for DTW template matching. Algorithm B keeps everything the same from algorithm A except for adding an orientation correction layer described in section III. Algorithm C modifies algorithm A to make it orientation invariant. The modifications involves (1) choosing corresponding orientation invariant features for Naive Bayes classifier (detailed features listed in Table 5.3) and (2) using accelerometer magnitude for DTW instead of 3-axis accelerometer measurements. The differences between the 3 algorithms are summarized in Table 5.2

### 5.3.4 Results

Two days data from each subject was processed with all three algorithms and the classification results are reported in the Table 5.4. Since we are evaluating a

Algorithm	Orientation Correction	NB Features	DTW
Algo. A	Not Included	F1 (orientation variant)	3-axis Accel.
Algo. B	Included	F1 (orientation variant)	3-axis Accel.
Algo. C	Not Included	F2 (orientation invariant)	Accel. Magnitude

Table 5.2: Algorithm Design (Details of F1 and F2 listed in Table 5.3)

binary classifier, precisions and recalls defined in Equation 5.5 are calculated and reported for each day the data was collected.

$$\begin{aligned} precision &= \frac{tp}{tp + fp} \\ recall &= \frac{tp}{tp + tf} \end{aligned} \quad (5.5)$$

With algorithm A, the model works well when the sensor mounting orientations are correct (e.g. 151 day2); but if the sensor mounting orientation was not correct, the system would fail by not being able to capture walking bouts (e.g. 152 day2). On average,  $34.97\% \pm 41.38\%$  precision and  $22.66\% \pm 36.44\%$  recall was achieved. Note that for the days when no walking bouts were detected, NaN's were recorded for recall since there was no positive output from the classifier. But when calculating the mean and standard deviation, we substitute the NaN's with 0's since ignoring these days will make the overall performance better than reality.

With algorithm B, however, by adding an orientation correction layer, sensor data was corrected successfully, and the system outputs an accurate classification results. On average,  $85.20\% \pm 7.33\%$  precision and  $93.50\% \pm 7.83\%$  recall was achieved.

Algorithm C also works well for this problem by using only orientation invariant features. On average,  $85.66\% \pm 8.91\%$  precision and  $96.27\% \pm 4.48\%$  recall was achieved. This is because we are interested in a binary classifier that deals with walking activity only. If we are interested in differentiating activities with

Feature Set	Included Features
F1	standard deviation of y axis, maximum of y axis, mean of y axis
F2	standard deviation of magnitude, maximum of magnitude, mean of magnitude

Table 5.3: Details of the Two Feature Set

directions (e.g. walking upstairs and downstairs), then Algorithm C would not be able to provide reliable performance because only orientation invariant features are used.

## 5.4 Discussions

Current state-of-the-art machine learning techniques allow the wide application of using motion sensors to produce human daily activity classification. One example of such systems is SIRRACt developed by UCLA Wireless Health Institute. [XBK11] claims that on average 83.61% precision and 84.38% recall can be achieved. But one important assumption lies in the correct sensor mounting orientation which can be very hard to insure especially when the scale of the study increases.

In this chapter, a new double-layer automatic orientation correction classifier was proposed and evaluated. On average, 85.20% precision and 93.45% recall was achieved over 7 healthy subjects and 2 stroke patient subject with no assumption on the sensor orientation. In the mean time, the original SIRRACt classifier yields on average 34.97% precision and 22.66% recall over the same set of data. We can conclude that the newly proposed method achieves robust classification performance by relaxing the assumption correct sensor mounting orientation.

	Subject No.	Date No.	Algo. A		Algo. B		Algo. C	
			Precision	Recall	Precision	Recall	Precision	Recal
healthy	151	day1	72.64%	4.66%	94.03%	98.06%	91.20%	99.61%
		day2	95.93%	99.49%	95.93%	99.49%	92.71%	99.68%
	151a	day1	88.74%	82.96%	78.05%	83.88%	88.21%	83.79%
		day2	NaN	0%	86.25%	99.05%	86.18%	99.54%
	152	day1	14.55%	0.14%	89.47%	97.39%	94.41%	97.53%
		day2	NaN	0%	93.83%	98.57%	94.76%	98.20%
	153	day1	NaN	0%	85.00%	96.68%	83.64%	95.95%
		day2	NaN	0%	81.29%	98.05%	87.48%	99.02%
patient	154	day1	41.82%	29.77%	85.82%	100%	80.87%	100%
		day2	NaN	0%	73.65%	99.47%	88.37%	97.88%
	185	day1	NaN	0%	95.04%	97.43%	94.62%	99.55%
		day2	NaN	0%	81.28%	91.47%	83.52%	99.16%
	192	day1	89.80%	97.05%	91.30%	98.13%	93.97%	98.13%
		day2	NaN	0%	87.63%	89.19%	83.18%	96.84%
	98	day1	41.67%	4.99%	70.27%	88.68%	77.02%	89.60%
		day2	NaN	0%	80.24%	83.09%	75.57%	94.26%
	117	day1	89.14%	63.57%	80.66%	93.76%	57.84%	99.12%
		day2	95.14%	25.32%	83.89%	70.65%	72.28%	97.39%
	overall	mean	34.97%	22.66%	85.20%	93.45%	84.77%	96.96%
		std	41.38%	36.44%	7.33%	7.91%	9.58%	4.14%

Table 5.4: Classification Results

# CHAPTER 6

## Conclusion

### 6.1 Contributions

This thesis focuses on solutions for sensor placement errors in human motion tracking and activity classification problems.

We start with proposing an opportunistic sensor orientation calibration method for an existing lower body motion tracking system in Chapter 2. Sensor misorientation and rotational displacement cases are considered individually, each of which are modeled by a 2D rotation around a fixed axis. Rotation angles are estimated by comparing the testing data with the training template. We prove that the sensor signals can be recovered and that accurate step length estimation can be achieved.

In Chapter 3, we then propose an algorithm to estimate sensor linear displacement and then use this information to modify the current ZUPT algorithm to achieve single sensor lower body motion tracking using only one sensor mounted on the ankle. Integrating motion accelerations twice generates traveling distance with very large cumulative errors. The current ZUPT method resets the velocity to be zero during the stance phase to compensate for this error and achieve high accuracy in motion tracking. Observing the fact that the calf is rotating around the ankle during the stance phase, the Non-ZUPT method proposed in this thesis uses sensor position and angular velocity (output of the gyroscope) to estimate the velocity value during each stance phase. Resetting velocity to that non-zero

value will compensate the cumulative error and then generate traveling distance with high accuracy. We prove the validity of this method with flat-floor walking as well as stairs walking. High accuracies were achieved in both cases.

Following that, in Chapter 4, we propose a differential algorithm that can estimate step length with high robustness regardless of the exact sensor mounting position or orientation. In an inverted pendulum human walking model, the traveling distance within each gait cycle can be modeled by multiplying the leg length and the sine of the orientation change within this period. In this case, the exact sensor position or orientation have no effect on the system performance as long as it is firmly attached. This is because, we are only interested in the orientation change instead of the absolute value.

Finally, in Chapter 5, a double layer walking activity classification model was proposed. The first layer consists of a highly conservative, orientation invariant classifier where only the walking beacons with high confidence were selected. These walking beacons were compared with the training template to estimate the rotation matrix, which was used to correct the whole day data. The whole day corrected sensor signals were then fed into the second layer, more accurate, orientation variant classifier.

In conclusion, this thesis presented four different solutions for sensor misplacement issues in motion tracking systems and activity classifiers. Validity of these methods are proved via real world experiments with healthy subjects as well as patients. The opportunistic calibration method proposed in Chapter 2 can be applied to systems where sensor mounting orientation errors exists. This method works as long as training templates of walking activities with correct sensor mounting are available. The trajectory reconstruction problems that we used require four sensors, but only accelerometers are needed. The non-ZUPT algorithm that we proposed in Chapter 3 can be used for lower body trajectory reconstruction and sensor linear displacement estimation. It can recover full walking trajectory with

only one sensor mounted on the ankle of a subject. Both accelerometers and gyroscope devices are required for this method to work. The PI method discussed in Chapter 4 works well even if the location of the sensors change. Even though the full trajectory can not be recovered, the most important metric, the step length, can be calculated. It requires two sensors mounted on both ankles of a subject and both accelerometers and gyroscopes are needed. The opportunistic calibration method for incorrect sensor orientation was applied to a real world activity classification problem. This method can be used when whole day data are available. For classification problems, only accelerometer sensors are needed.

## 6.2 Future Work

Future works also include applying similar methods to the upper body activities. Upper body activities are usually more challenging problems. The first reason is that people have less constraints in upper body movements. Therefore, more variety can appear when people are doing the same activity which makes it more challenging to model precisely. Also, the vast majority of upper body activities is not periodic. In addition, the existence of lower body movement will interfere with the sensor signals in the upper body. These situations are widely seen in people's daily life (e.g. drinking coffee while walking on the street), and make upper body activity signal processing tasks more challenging. Extensive data collection combined with careful modeling will be required to make progress.

One of the innovations of this thesis is that one can opportunistically identify certain segments of the data that serve as calibration points or anchors for the rest. Examples of such anchors are stance phase in a gait in Chapter 3 and clear walking instances of high confidence in daily activities in Chapter 5. Beyond this, for future research, we are interested in a question of how common are these kinds of “anchor points” in data streams for a broader range of problems. There

is also a theoretical question of how often these must occur in order to 1) be able to calibrate and track time varying stochastic processes (e.g. movement of sensor positions or changes in environment, etc.) and 2) have confidence in the accuracy of the inferences. This almost surely relates to how well we can model the underlying stochastic process and the observation noise.

This thesis serves as a good example of such opportunistic calibration systems. Here we have good models for the accelerometer noise and gyro drift and some very strong models for walking that make our methods possible. We also have intense variability in walking among subjects that makes things challenging. Modeling and noise specifications of stochastic processes are important factors to consider for the applications of similar methods to a broader range of problems.

## REFERENCES

- [BBS08] S.J.M. Bamberg, A.Y. Benbasat, D.M. Scarborough, D.E. Krebs, and J.A. Paradiso. “Gait Analysis Using a Shoe-Integrated Wireless Sensor System.” *Information Technology in Biomedicine, IEEE Transactions on*, **12**(4):413–423, July 2008.
- [CNH12] D. Simn Colomar, J. O. Nilsson, and P. Hndel. “Smoothing for ZUPT-aided INSs.” In *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*, pp. 1–5, Nov 2012.
- [DC15] Bruce H. Dobkin and S. Thomas Carmichael. “The Specific Requirements of Neural Repair Trials for Stroke.” *Neurorehabilitation and Neural Repair*, 2015.
- [DD11] B. Dobkin and A. Dorsch. “The promise of mHealth: daily activity monitoring and outcome assessments by wearable sensors.” *Neurorehabil Neural Repair*, **25**(9):788–98, 2011.
- [DLR77] A.P. Dempster, N.M. Laird, and D.B. Rubin. “Maximum Likelihood from Incomplete Data via the EM Algorithm.” *Journal of the Royal Statistical Society. Series B*, **39**(1), 1977.
- [DTD15] Andrew Dorsch, Seth Thomas, and Bruce Dobkin. “Walking Quality During Inpatient Stroke Rehabilitation Assessed by Wireless Sensing (P5.176).” *Neurology*, **84**(14 Supplement), 2015.
- [DTX15] Andrew Dorsch, Seth Thomas, Xiaoyu Xu, William Kaiser, and Bruce Dobkin. “SIRRACt: An International Randomized Clinical Trial of Activity Feedback During Inpatient Stroke Rehabilitation Enabled by Wireless Sensing.” *Neurorehabilitation and Neural Repair*, **29**(5):407–415, 2015.
- [DXB11] BH Dobkin, X Xu, M Batalin, S Thomas, and W. Kaiser. “Reliability and validity of bilateral ankle accelerometer algorithms for activity recognition and walking speed after stroke.” *Neurorehabil Neural Repair*, **42**(8):2246–50, 2011.
- [FBR09] K. Frster, P. Brem, D. Roggen, and G. Trster. “Evolving discriminative features robust to sensor displacement for activity recognition in body area sensor networks.” In *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on*, pp. 43–48, Dec 2009.
- [FHC12] Ascher Friedman, Nabil Hajj Chehade, Chieh Chien, and Greg Pottie. “Estimation of accelerometer orientation for activity recognition.” In

- Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pp. 2076 –2079, 28 2012-sept. 1 2012.
- [GBN11] L. Gao, A. K. Bourke, and J. Nelson. “A system for activity recognition using multi-sensor fusion.” In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, pp. 7869–7872, Aug 2011.
  - [KL08] Kai Kunze and Paul Lukowicz. “Dealing with sensor displacement in motion-based onbody activity recognition systems.” In *Proceedings of the 10th international conference on Ubiquitous computing*, UbiComp ’08, pp. 20–29, New York, NY, USA, 2008. ACM.
  - [Kuo10] Donelan J. M. Kuo A. D. “Dynamic principles of gait and their clinical implications.” *Physical Therapy*, **90**(2):157–174, Feb. 2010.
  - [LHW07] Moe R. Lim, Russel C. Huang, Anita Wu, Federico P. Girardi, and Jr Frank P. Cammisa. “Evaluation of the Elderly Patient With an Abnormal Gait.” *J Am Acad Orthop Surg*, **15**(2):107–117, Feb. 2007.
  - [PS10] Sang Kyeong Park and Young Soo Suh. “A Zero Velocity Detection Algorithm Using Inertial Sensors for Pedestrian Navigation Systems.” *Sensors*, **10**(10):9163, 2010.
  - [Pub09] Harvard Health Publications. “Walking: Your steps to health.” [http://www.health.harvard.edu/newsletter\\_article/Walking-Your-steps-to-health](http://www.health.harvard.edu/newsletter_article/Walking-Your-steps-to-health), Aug 2009.
  - [RJ93] L. R. Rabiner and B. Juang. “Chapter 4 pattern comparison techniques.” In *Fundamentals of speech recognition.*, 1993.
  - [SMS05] A.M. Sabatini, C. Martelloni, S. Scapellato, and F. Cavallo. “Assessment of walking features from foot inertial sensing.” *Biomedical Engineering, IEEE Transactions on*, **52**(3):486–494, March 2005.
  - [SR06] Manoj Srinivasan and Andy Ruina. “Computer optimization of a minimal biped model discovers walking and running.” *Nature*, **439**(7072):72–75, 2006.
  - [WCX13a] Yan Wang, Chieh Chien, J. Xu, G. Pottie, and W. Kaiser. “Gait analysis using 3D motion reconstruction with an activity-specific tracking protocol.” In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 1041–1045, May 2013.
  - [WCX13b] Yan Wang, Chieh Chien, James Xu, Greg Pottie, and William Kaiser. “gait analysis using 3D motion reconstruction with an activity-specific

- tracking protocol.” In *The 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2013.
- [Wik16a] WikiPedia. “Euler angles.” [https://en.wikipedia.org/wiki/Euler\\_angles](https://en.wikipedia.org/wiki/Euler_angles), Feb 2016.
- [Wik16b] WikiPedia. “Quaternion and spatial rotation.” [https://en.wikipedia.org/wiki/Quaternions\\_and\\_spatial\\_rotation](https://en.wikipedia.org/wiki/Quaternions_and_spatial_rotation), Jan 2016.
- [WXX13] Yan Wang, James Xu, Xiaoyu Xu, Xiaoxu Wu, Gregory Pottie, and William Kasier. “Inertial Sensor Based Motion Trajectory Visualization and Quantitative Quality Assessment of Hemiparetic Gait.” In *Proceedings of the 8th International Conference on Body Area Networks*, BodyNets ’13, pp. 169–172, 2013.
- [XBK11] Xiaoyu Xu, Maxim A. Batalin, William J. Kaiser, and Bruce Dobkin. “Robust Hierarchical System for Classification of Complex Human Mobility Characteristics in the Presence of Neurological Disorders.” In *Body Sensor Networks (BSN), 2011 International Conference on*, pp. 65–70, May 2011.
- [ZD14] Y. Zhong and Y. Deng. “Sensor orientation invariant mobile gait biometrics.” In *Biometrics (IJCB), 2014 IEEE International Joint Conference on*, pp. 1–8, Sept 2014.