BODY MOTION CAPTURE USING MULTIPLE INERTIAL SENSORS

A Thesis Submitted to the College of
Graduate Studies and Research
in Partial Fulfillment of the Requirements
for the Degree of Master of Science
in the Department of Electrical and Computer Engineering
University of Saskatchewan
Saskatoon

By

XIAOYE XIA

PERMISSION TO USE

In presenting this thesis in partial fulfilment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Electrical and Computer Engineering
57 Campus Drive
University of Saskatchewan
Saskatoon, Saskatchewan
Canada
S7N 5A9

ABSTRACT

Near-fall detection is important for medical research since it can help doctors diagnose fall-related diseases and also help alert both doctors and patients of possible falls. However, in people's daily life, there are lots of similarities between near-falls and other Activities of Daily Living (ADLs), which makes near-falls particularly difficult to detect. In order to find the subtle difference between ADLs and near-fall and accurately identify the latter, the movement of whole human body needs to be captured and displayed by a computer generated avatar.

In this thesis, a wireless inertial motion capture system consisting of a central control host and ten sensor nodes is used to capture human body movements. Each of the ten sensor nodes in the system has a tri-axis accelerometer and a tri-axis gyroscope. They are attached to separate locations of a human body to record both angular and acceleration data with which body movements can be captured by applying Euler angle based algorithms, specifically, single rotation order algorithm and the optimal rotation order algorithm.

According to the experiment results of capturing ten ADLs, both the single rotation order algorithm and the optimal rotation order algorithm can track normal human body movements without significantly distortion and the latter shows higher accuracy and lower data shifting. Compared to previous inertial systems with magnetometers, this system reduces hardware complexity and software computation while ensures a reasonable accuracy in capturing human body movements.

Keywords — *Motion capture; Activities of Daily Living (ADLs); Inertial sensors; Euler angles*

ACKNOWLEDGEMENTS

It is difficult to overstate my gratitude and appreciation to my supervisors, Dr. Daniel Teng, Dr. Jenny Basran and Dr. Anh van Dinh, for giving me the confidence to explore my research interests and the guidance to avoid getting lost in my exploration. From beginning to end, their patience, tremendous support, as well as sheer genius made possible any progress that was made.

I wish to express my appreciation to Dr. Jenny Basran. As my co-supervisor, she gave me lots of professional advices and new ideas from medical and kinesiology areas and supported me all the time. Meanwhile, I am extremely grateful to Dr. Vanina Dal Bello-Haas, Flo Wagner and my friend, Zheng Qian, for assisting me to complete the whole experiments. Furthermore, thanks to my partner Joseph Schneider whom I have been worked with for the past two years. His open suggestion, support and assistance encouraged me to conquer all the difficulties.

I would like to thank all the classmates and friends; working with them made my time during my study and research a wonderful experience. Also, thanks to my boyfriend Kuande Wang for his continuous support and encouragement throughout my studies.

Countless and sincere thanks also goes to my parents. They are always supportive during my whole life. To them I dedicate this thesis.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ADL	Activities of Daily Living
ADC	Analogue-to-Digital Converter
DAC	Digital-to-Analogue Converter
DMA	Direct Memory Access
DOF	Degree Of Freedom
ECG	Electrocardiogram
EMG	Electromyogram
FANFARE	Falls and Near Falls Assessment Research and Evaluation
FOCSVM	Fuzzy One Class Support Vector Machine
GSR	Galvanic Skin Response
I^2C	Inter-Integrated Circuit
LDO	Low Drop Out
LR-WPANs	Low-Rate Wireless Personal Area Networks
MEMS	Micro Electro Mechanical Systems
PANs	Personal Area Networks
PC	Personal Computer
RAM	Random-Access Memory
SD	Secure Digital
SHIMMER	Sensing Health with Intelligence Modularity, Mobility and Experimental Reusability
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver/Transmitter
USART	Universal Synchronous/Asynchronous Receiver/Transmitter

1 Introduction

This thesis is based on the ongoing project called "Falls and Near-Falls Assessment Research and Evaluation" (FANFARE), the objectives of which are combining different technologies for fall and near-fall researches and protecting elderly people from falling down.

In this chapter, the motivation of developing the motion capture system and the reason why inertial sensors were chosen is presented first. Then the objectives of this work and the main contribution will be given, followed by the thesis outline.

1.1 Motivation

As a result of improved standards of living and advances in medical care, a growing number of elderly Canadians are living alone at home with little supervision. However, the majority of these elderly citizens still require some degree of monitoring in case of emergencies. It is impossible for medical professionals to continuously monitor these people in the case of such emergencies. One major concern is the possibility of these at-home elderly citizens falling and injuring themselves - a common and unexpected event which can occur in everyday life. More than one third of people over the age of 65 will fall at least once per year and about ten to fifteen percent of these falls will cause serious injuries [1,2]. In order to alarm and protect elderly from falling down in advance, near-fall detection should be taken into account.

Near-fall detection is important for medical research to help doctors diagnose fall-related diseases and also to help alert both the doctor and the patient (at-home) of near-falls to protect the elderly people in advance to prevent fall-related injuries. Near-fall is a state between fall and normal activity, which can be considered as the potential fall. It is

difficult to give a definition of near-fall, since different people may have different feelings of near-fall. Basically, a near-fall can be considered as two-step movements: first step is that a subject loses balance, and the second step is that he/she gets his/her balance back immediately. Near-fall may happen anytime and anywhere of people's daily life. If people cannot control their balance, then a near-fall can easily transform into a real fall. Therefore, it is very important and necessary to study and detect near-falls.

Existing researches are mainly focused on fall detection, however, there has been very little research done on near-fall detection. For example, a fall detection system using one-belt kinematic sensor is presented in [3], which can detect four types of falls (forward, backward, tilt left and right) among other pre-selected normal activities with the accuracy of 97%. In [4], C. Dinh et al. presented real-time fall detection approach using fuzzy logic and a neural network. By attaching a single accelerometer on the human body, it can detect four kinds of falls (forward, backward, sideward and collapse) with the accuracy of 94%. A fall detection system using FOCSVM (Fuzzy One Class Support Vector Machine) is proposed in [5]. In this system, two video cameras are used to capture the motions, and the results show that 99% of falls can be correctly detected. Although these systems can capture different falls with very high accuracy, when using them in near-fall detection, the accuracy will be decreasing significantly. Current fall detection systems detect falls usually based on the different features between falls and normal activities. However, in near-falls, the changes in people's body location or orientation, as well as heart rate and blood pressure are not as significant as them during falls. Also, there are lots of similarities between near-falls and other Activities of Daily Living (ADLs), which makes near-falls difficult to be identified and recognized. Meanwhile, the above systems can only tell doctors there is a fall happened, but they cannot show more information about how that fall happened and what the subject did before it happened. Since people may have different physical features, such as walking styles and sitting postures, only the detected falls could not help doctors to diagnose the reason of falling down for individual subject. Furthermore, researchers could not find a good way to generate "real" near-falls in a laboratory environment, which makes the near-fall researches very difficult.

From the clinical aspect, in order to study and detect near-falls, the most efficient and straightforward way is to monitor the motions of a human body when a near-fall is happening, such as videos. However, in reality, it is impossible to install cameras in people's houses and continuously capture the movements based on the consideration of protecting people's privacy. To avoid this problem and still record the useful motion information, motion capture technologies is involved in this research. Near-fall detection is a very challenging subject and it is the goal of the whole FANFARE project. As a part of the FANFARE project, the objective of this thesis is using motion capture technologies to continuously record subjects' movements of normal activities, and show these movements by a computer avatar. By watching the avatar playing those ADL movements, doctors can intuitively study and analyze the related features of normal activities, which can be used later in near-fall detection and recognition among normal activities. The motion capture technologies can also be combined with fall detection systems to improve the accuracy. Usually fall detection systems are not 100% accurate. By combining a motion capture system, once a fall is detected, the doctor can use the collected motion information to generate an avatar to repeat that movement and to further detect whether it is a real fall or normal activities.

Motion capture is a process which records real-time information about the motion of a human body or other systems and display motions using recorded information. It is used in robotics [6,7], synthetic environments as well as a tool for education [8], training [9,10] and entertainment [11,12]. It also shows great advantages in near-fall detection applications. With motion capture systems, the constant monitoring of subject's heart rate, muscle strength or blood pressure can be avoided. Also, motion capture provides the movement information of each body segment to facilitate near-fall detection algorithm.

Recently, a great number of motion capture systems have been developed for human body motion capturing in virtual reality, biomechanical and biomedical applications. These systems include mechanical sensing systems, optical sensing systems, acoustic sensing systems, magnetic sensing systems and inertial motion capture systems. Mechanical motion capture systems such as Gypsy [13] and Measurand ShapeWrap [14] use exoskeleton that is attached to the articulated body segment to measure joint angles. However, wearing exoskeletons not only restraint performers' actual movements but also

cause people uncomfortable during extended time periods. Optical motion capture systems are usually based on measurements of light emitted by a light source or reflected by an object. Typical systems are Vicon [15] and Qualisys [16]. It is favored in the computer-animation community and the film industry. But the disadvantages are also obvious: it depends on complex devices such as special light source generators, markers or cameras which restrict the range of motion to a room space. Acoustic/ultrasonic tracking systems such as Cricket location System detect the location by calculating the distance between the generators and the markers [17]. It requires one or more sound generation devices to be placed at fixed points in a testing environment and several sound markers to be attached on segments of a human body. Acoustic sensors typically offer a larger detection range than mechanical sensors. However, their accuracy can be affected by different factors like wind speed, temperature and air currents in the environment since they all affect the sound speed and consequently affect the accuracy of distance detection. Position and orientation information can also be obtained by using small magnetic sensors attached on body segments to sense the earth magnetic field or artificial generated magnetic fields. This has been a popular means of motion tracking for many years. However, unfortunately, these magnetic systems often have high power consumption and they are extremely sensitive to the ferromagnets in the environment.

Ideal motion capture technology must meet several requirements. It should be low cost and low power consumption, be tolerant to environmental interference such as optical, acoustic or electromagnetic noise, be able to track multiple users and maintain reasonable accuracy and can be used in different environment without distance restrictions [18]. The primary reason that above motion capture systems fail to meet the requirements described is the dependence on artificially generated “sources” and markers to determine orientation and location information [18,19]. Those “sources” and markers can be easily seen in optical and acoustic systems, in which light sources (such as lasers, infrared and light bulbs) or sound generators should be firstly placed at fixed points in the lab environment and markers should be attached on tester to receive the signals from the “sources”.

The development of microelectromechanical inertial sensors over the last few years makes it possible to determine orientation based on the passive measurement of physical

quantities which are directly related to the location and orientation of a rigid body on which they are attached. Inertial sensors avoid the problems associated with current technologies and ultimately allow capturing movements over a virtually unlimited area. Inertial sensor motion capture systems usually use accelerometers to measure gravity related vectors and gyroscopes to measure the change in angular rate. There are currently a variety of different inertial sensing systems that have been proposed and developed to solve specific problems. For example, a ubiquitous human upper limb motion capture system has been proposed using wearable micro-inertial-sensors (accelerometers and gyroscopes) in [20]. In [21], a tri-axial accelerometer was used to monitor certain human movements and postures. Also, a fall detection system using both accelerometers and gyroscopes is presented in [22] to recognize four kinds of static postures: standing, bending, sitting and lying. Existing inertial motion capture systems usually detect certain parts of human body or only certain types of movements. In this thesis, a wireless motion capture system using inertial sensors is presented, which can capture the whole body motions and almost all the basic normal activities during our daily life. At the meantime, the recorded motion information can be further used for near-fall detection in medical research.

1.2 Objectives and Contribution

In the thesis, a novel wireless motion capture system using inertial sensors is presented to record relevant motion information and capture ten selected ADLs. To accommodate the portability and wide out-door range requirements of this human motion capture system – so that those being monitored will still be able to perform daily activities – the sensor unit contains a tri-axis accelerometer and a tri-axis gyroscope which do not require any extra light sources or sound generators. The main objectives of this inertial sensor based motion capture project include:

- Using a low-cost and low-power wearable motion capture system to capture the ADLs for near-fall research.
- Develop a low-complexity Euler-angle based algorithm for ADL motion capture.
- Build the visual computer avatar model to display the subject's movements.

The main contributions of the proposed motion capture system in this project can be summarized as follows:

- This motion capture system can capture the complete body movements of a human while some previous researches can detect only part of human body movements. Ten ADLs selected in the experiment almost covered all the basic activities in an elderly person's daily life while previous researches only focus on some of these movements.
- Hardware cost can be reduced by using only accelerometers and gyroscopes instead of combining magnetometers such as Quaternion-based motion capture systems [23,24]. The former can reduce the hardware complexity and cost.
- Compared to the conventional Euler angle-based algorithm [25], an optimal rotation order algorithm is proposed to obtain high accuracy.
- The created avatar animation can greatly help doctors on studying and analyzing the features of normal activities, based on which doctors can further studying the features of near-falls. Meanwhile, the animation can give doctors intuitive pictures of how a fall happened and help them to diagnose those fall-related diseases.

As shown above, this inertial sensor motion capture system shows its great potential in not only medical research for near-fall detection, but also other applications, such as computer animation, video gaming and virtual reality.

1.3 Thesis Outline

The rest of the thesis is structured as follows: Chapter 2 reviews different methods of representing the orientation of a rigid body. Some existing algorithms of modeling human body motion applications are introduced. Chapter 3 presents a description of Euler angle-based body motion capture algorithms. Both single rotation order algorithm and optimal rotation order algorithm are discussed in detail. Chapter 4 discusses the hardware architecture of the SHIMMER sensor system which is applied in the body motion capture experiment. The TinyOS environment and NESC language used in the SHIMMER platform are introduced in this chapter as well. Chapter 5 introduces the environment of

the experiment and the experiment steps. Some preparations of the experiment are completed before the implementation, including calibration of the sensors, overnight drifting test and the measurements. Chapter 6 presents the implementation of Euler angle-based body motion capture algorithms and discusses about the experiment results, data processing and simulation analysis. The comparison with previous inertial motion capture systems is discussed as well. The final chapter presents a short summary along with the conclusions. It also provides recommendations for future enhancements of this research.

2 Motion Captures

Motion capture (or motion tracking) started as a photogrammetric analysis tool in biomechanics research in the 1970s and 1980s. This later extended into education, training, sports and recently into computer animation for television, cinema and video games as the technology matured [26]. The major task involved in motion capture is to record relevant movement data of the system (in this case, a human body) in time for immediate or delayed analysis and playback. Besides capturing the orientation and location information of body's motion, this technology has also been applied to detect deformation of the face and muscle masses; for example to distinguish hand gestures [27].

This section introduces the background of body motion capture, including the notion of coordinate systems in body motion capture and rigid body orientation representation methods. Based on this basis, some existing body motion capture algorithms are discussed as well as several previous motion capture systems.

2.1 Coordinate Systems in Body Motion Capture

In order to represent the orientation of a rigid body, it is conventional to choose coordinate systems attached to an appropriate inertial frame and then describe all the vectors within these coordinates. Typically, there are two kinds of coordinate systems in body motion capture system, reference coordinates and body coordinate. Reference coordinates usually point to some fixed directions which are unchanged during the motion. Therefore, all the rotation and displacement of other coordinates can be depicted in the reference coordinate system as some dynamic motion sequences. While, body coordinate system is defined according to the mounted sensors, which provide a measurement basis for motion analysis.

Before discussing the human motion capture methods, it is necessary to define several symbols used in the thesis. Here, earth is chosen as the reference, let ${}^{e}X$-, ${}^{e}Y$- and ${}^{e}Z$-axis be the orthogonal reference coordinate system oriented earth south, earth east and radially inward to the earth up respectively. Let ${}^{b}X$-, ${}^{b}Y$- and ${}^{b}Z$-axis are the orthogonal tri-axes of the body coordinate align to the X, Y, Z-axis of the sensor circuit board. Since the sensor board is attached on the body segment and any deformation of the muscle in motion is ignored, the body segment can be represented as an unchanged vector in the body coordinate system. The earth-body coordinate system pair composes a frame to describe the human body motion.

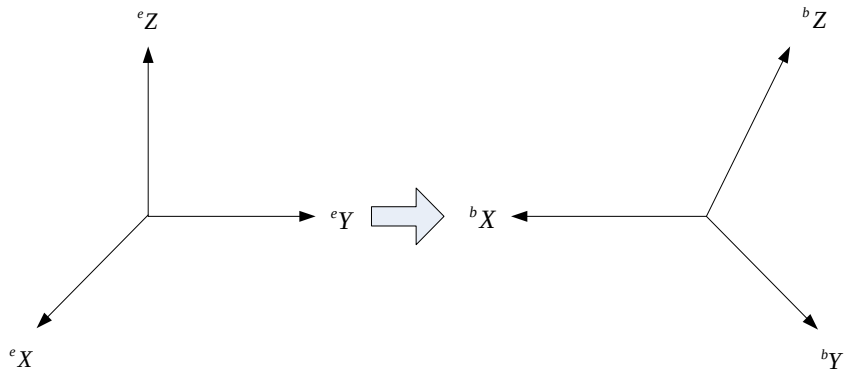


Figure 2.1 Earth coordinate system and body coordinate system

Figure 2.1 shows the earth coordinate system and the body coordinate system used in human body motion capture. By combining the gravity direction and the angular information measured by the accelerometers and the gyroscopes, body movements can be considered as the rotation of body coordinate in the earth coordinate. The main task of motion capture is using inertial information to find the relationship between the two coordinate systems.

2.2 Rigid Body Orientation Representation Methods

A human body can be modeled as a set of related rigid bodies. A rigid body is an idealization of a solid body with finite size in which the deformation is neglected. Without deformation, links, joints, limb segments and a body's trunk can be simplified as

fixed shape segments and arranged in a tree-like structure. Specifying the posture or attitude of a body involves describing the position and orientation of each individual segment. The natural relationships between the links in the structure will determine whether the positions and orientations of the segments should be described individually or relatively to one another. Specifically, the muscles and skeleton structure restrict the range of individual segments' movement. Hence, to properly model human body motion, a number of factors must be taken into account. These factors include the ability to represent all orientations, computational efficiency, storage capacity, and algorithms.

Rigid bodies are solid and completely inelastic, which provides a convenient simplified model to a deformable body and places fewer restrictions on motion modeling. There are numerous methods used to express the orientation of rigid bodies but each has some caveats. Possible methods include direction cosines, Euler angles, Homogeneous matrices and quaternion and vector-angle pairs.

2.2.1 Direction Cosines

The direction cosines of a vector are the cosines of the angles between the vector and the three coordinate axes [28].

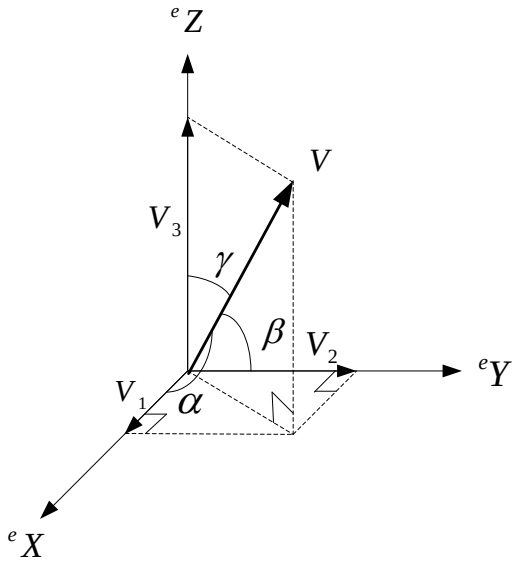


Figure 2.2 Direction cosines

As showed in Figure 2.2, if V is a vector defined in a 3-D space and the reference coordinate is (${}^{e}X$, ${}^{e}Y$, ${}^{e}Z$), then,

$$V = v_1 x + v_2 y + v_3 z \tag{2.1}$$

where x, y, z represent the unit vectors align to the correspond axis. Then, the direction cosines are

$$\cos\alpha = \frac{V \bullet \hat{x}}{\|V\|} = \frac{v_1}{\sqrt{v_1^2+v_2^2+v_3^2}} \tag{2.2}$$

$$\cos\beta = \frac{V \bullet \hat{y}}{\|V\|} = \frac{v_2}{\sqrt{v_1^2+v_2^2+v_3^2}} \tag{2.3}$$

$$\cos\gamma = \frac{V \bullet \hat{z}}{\|V\|} = \frac{v_3}{\sqrt{v_1^2+v_2^2+v_3^2}} \tag{2.4}$$

and

$$\cos^2\alpha + \cos^2\beta + \cos^2\gamma = 1 \tag{2.5}$$

where, the direction angles α, β, γ are the angles between the vector and the positive ${}^{e}X$-, ${}^{e}Y$- and ${}^{e}Z$-axis.

The direction angles representation is simple and intuitive. However, it is not commonly used in body motion capture system since it cannot describe the dynamic change of a rigid body, especially when the coordinate is changing with time.

2.2.2 Euler Angle Rotations

One of the traditional methods for orientation representation is called Euler angles which describes the orientation of a rigid body using three rotations corresponding to specified axes. Using Euler angles, any coordinate set can be considered as a set of continuous rotations of a reference coordinate [29].

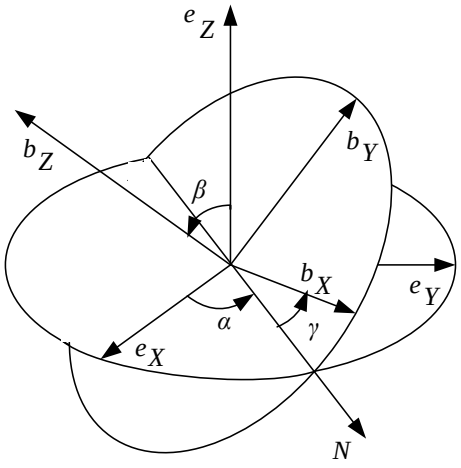


Figure 2.3 Euler angles

As shown in Figure 2.3, the Euler angles are three rotation angles corresponding to the ^eX, ^eY and ^eZ axes. The Euler angles are generated by considering a rotation which consists of shifting the ^eX-axis by γ, then the ^eY-axis by β and finally the ^eZ-axis by α, the shifting angles aforementioned can be denoted as roll (or bank), pitch (or elevation) and yaw (or azimuth) respectively.

Using Euler angles to represent rotations is straightforward. A vector in new coordinates can be represented using the same vector in old coordinates multiplied by a corresponding rotation matrix. However, singularity of the trigonometric functions is the main disadvantage of Euler angle rotations.

2.2.3 Conversion to Homogenous Matrix

A Homogeneous matrix is a 4×4 matrix, which includes both position and orientation information. In robotics, a position in reference coordinate is defined as a 3×1 position vector [30]

$$^eP = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \tag{2.6}$$

A rotation of the body coordinate with respect to the reference coordinate is described by a 3×3 rotation matrix ${}^{e}_{b}\boldsymbol{R}$, the elements of which are the trigonometric functions of Euler angles.

$$ {}^{e}_{b}R = \begin{bmatrix} {}^{e}\hat{X}_b & {}^{e}\hat{Y}_b & {}^{e}\hat{Z}_b \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{2.7} $$

Then, a frame is a set of four vectors giving position and orientation information. For example, as depicted in Figure 2.4, frame $\{body\}$ is described by ${}^{e}_{b}\boldsymbol{R}$ and ${}^{e}P_b$ with respect to frame $\{earth\}$, where ${}^{e}P_b$ is the vector which locates the origin of the frame $\{body\}$:

$$ \{body\} = \left\{ {}^{e}_{b}R,\ {}^{e}P_b \right\} \tag{2.8} $$

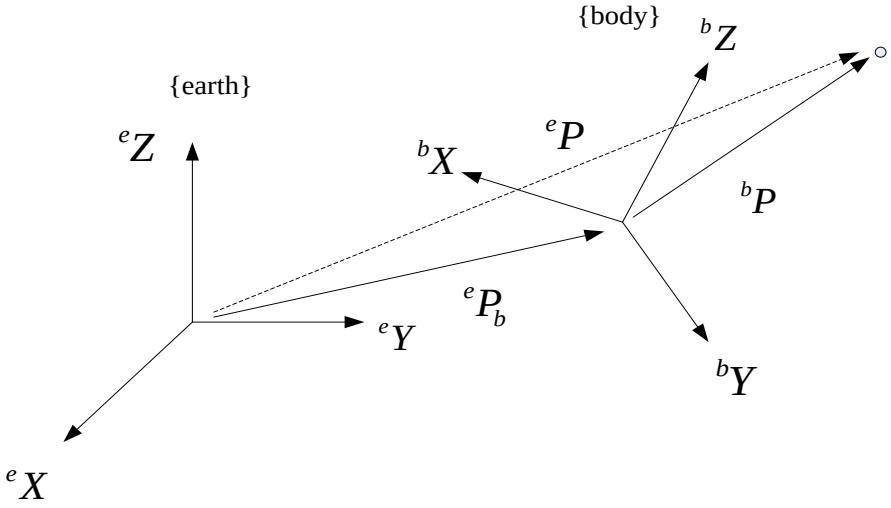


Figure 2.4 General transform of a vector [30]

Then,

$$ {}^{e}P = {}^{e}_{b}R\ {}^{b}P + {}^{e}P_b \tag{2.9} $$

The above equation has a structure as

$$ \begin{bmatrix} {}^{e}P \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{e}_{b}R & {}^{e}P_b \\ 0,0,0 & 1 \end{bmatrix} \begin{bmatrix} {}^{b}P \\ 1 \end{bmatrix} \tag{2.10} $$

where ${}^{e}_{b}\boldsymbol{R}$ is a 3×3 matrix, which describes the rotation. ${}^{e}P_b$ is a 3×1 matrix, which represents the displacement. The middle 4×4 matrix is the Homogeneous transform,

often denoted by ${}^{e}_{b}T$. Using this method can depict the movement of a rigid body more directly and roundly because it includes both position and orientation information. However, Homogeneous matrix also has singularity problems.

2.2.4 Quaternions and Vector-angle Pairs

The quaternion is a number system that extends the complex numbers. They were firstly described by Irish mathematician Sir William Rowan Hamilton in 1843 and applied to mechanics in 3-D space [31].

Quaternion uses three "imaginary" parts and one "real" part. The imaginary portion is often considered as a vector. The real part is called scalar. A quaternion can be given by

$$q = w + xi + yj + zk \tag{2.11}$$

or

$$\begin{aligned} q &= (w, x, y, z) \\ &= (w, v) \end{aligned} \tag{2.12}$$

where i, j and k denote the standard orthogonal basis for 3-D space, v is a vector of imaginary part and w is the real part. Intuitively, the three imaginary parts describe a vector and the real part expresses an angle of rotation about the vector. A quaternion has the following properties [31]

$$i^2 = -1, \quad j^2 = -1, \quad k^2 = -1, \quad ijk = -1 \tag{2.13}$$

$$ij = k = -ji \tag{2.14}$$

$$jk = i = -kj \tag{2.15}$$

$$ki = j = -ik \tag{2.16}$$

For any given quaternion, there are two unit quaternions which can be used to represent it. They are

$$q_1 = \begin{bmatrix} w \\ x \\ y \\ z \end{bmatrix}, q_2 = -q_1 = \begin{bmatrix} -w \\ -x \\ -y \\ -z \end{bmatrix} \tag{2.17}$$

Both of them represent the same orientations. Here, q_1 is used as a unit quaternion and

$$x^2 + y^2 + z^2 = 1 \tag{2.18}$$

Similarly, a quaternion can also be written as trigonometric functions as a complex number,

$$q_1 = (cos\frac{\theta}{2}, v\, sin\frac{\theta}{2}) \tag{2.19}$$

Assuming a vector v in 3-D space, the rotation of v by a quaternion q_1 is defined as

$$p_{rotated} = q_1 v {q_1}^{-1} \tag{2.20}$$

Figure 2.5 shows a geometric interpretation of the above equations, where the vector v is rotated an angle θ about the vector q_1.

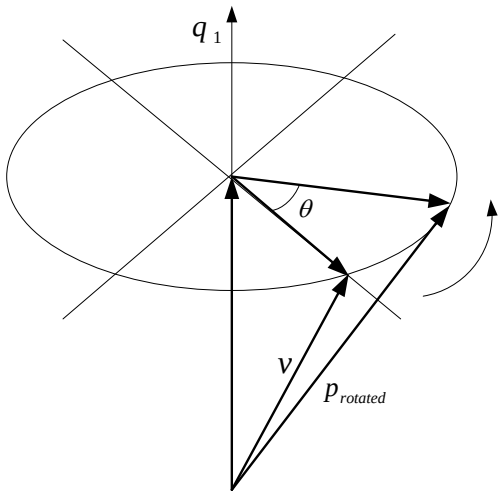


Figure 2.5 Quaternion rotation operator geometry [32]

As a matter of fact, the vector v can be considered as a quaternion with real part being zero. This representation is called vector-angle pairs which just use a vector and a rotation angle to describe rotation in 3-D space. This method avoids the singularities and just involves only two trigonometric functions rather than three successive Euler rotations. However, this method requires more direction information to evaluate the new rotation quaternion besides earth gravity. In order to get this information, people usually use the combination of accelerometer, gyroscope and magnetometer as a sensor node, which can detect not only the direction of gravity, but also the direction of earth magnetic field.

Direction cosines, Euler angles, Homogenous transform matrices and quaternions and vector-angle pairs are widely used by both the graphics and robotics communities. Direction cosines are simple and straightforward. However, it shows up difficulties when representing the dynamic rotation. Homogenous transform matrices include both rotation direction and the location information, which can depict the motion directly and roundly. While, using 4×4 matrix will inevitably increase the computational complexity. Quaternions and vector pairs require the storage of only 7 numbers compare to sixteen numbers in Homogenous matrices to represent a rotation. Although this method is widely used in motion capture application, adding the magnetometer for each sensor node will involve both hardware and software complexity.

2.3 Existing Body Motion Capture Algorithms

Recently, there has been a growing number of attempts to track the movements of human body parts using inertial sensors including accelerometer, gyroscope, etc. Accelerometers measure acceleration of a moving object with respect to the gravity and gyroscopes measure the rotation angular velocity. These sensors can also be combined with magnetometers which detect the earth magnetic field direction during the motion. Since the study on this area has been explored for many years, there are lots of algorithms based on different orientation representation methods discussed above. For example, a quaternion-based attitude filter is applied in [18,33], which is now widely used in body motion capture when combined inertial sensors and magnetic sensors. Another example is using Homogeneous matrix which can be found in [34]. This method is popular in robotics in which both the orientation of each segment and the location changing of the whole body during the motion would be taken into account.

Although direction cosines is an easier way to depict static orientation in 3-D space compared to other methods, the difficulties in dynamic motion representation is also obvious. In this session, a quaternion-based attitude filter and a Homogeneous matrix-based motion capture method are discussed. Both of them use sensor nodes with three sensors (i.e., accelerometer, gyroscope and magnetometer) for orientation tracking and location detection.

2.3.1 A Quaternion-Based Attitude Filter Used in Full Body Motion Capture

Figure 2.6 shows a block diagram of the complementary quaternion-based attitude estimation filter used in [18]. The filter takes inputs from three separate sensors, accelerometer, magnetometer, and gyroscope. Each of them is a tri-axis device and these axes are aligned together to form an orthogonal 3-D body coordinate (${}^{b}X$, ${}^{b}Y$, ${}^{b}Z$) with respect to the reference coordinate, (${}^{e}X$, ${}^{e}Y$, ${}^{e}Z$). The inputs of the filter are composed of three angular rates measured by the gyroscope (p, q, r), three accelerations measured by the accelerometer (h_1 , h_2 , h_3) and three magnetic vectors measured by the magnetometer (b_1, b_2, b_3) in body coordinate. Its output is a unit quaternion, q, which contained the orientation information of the rigid body.

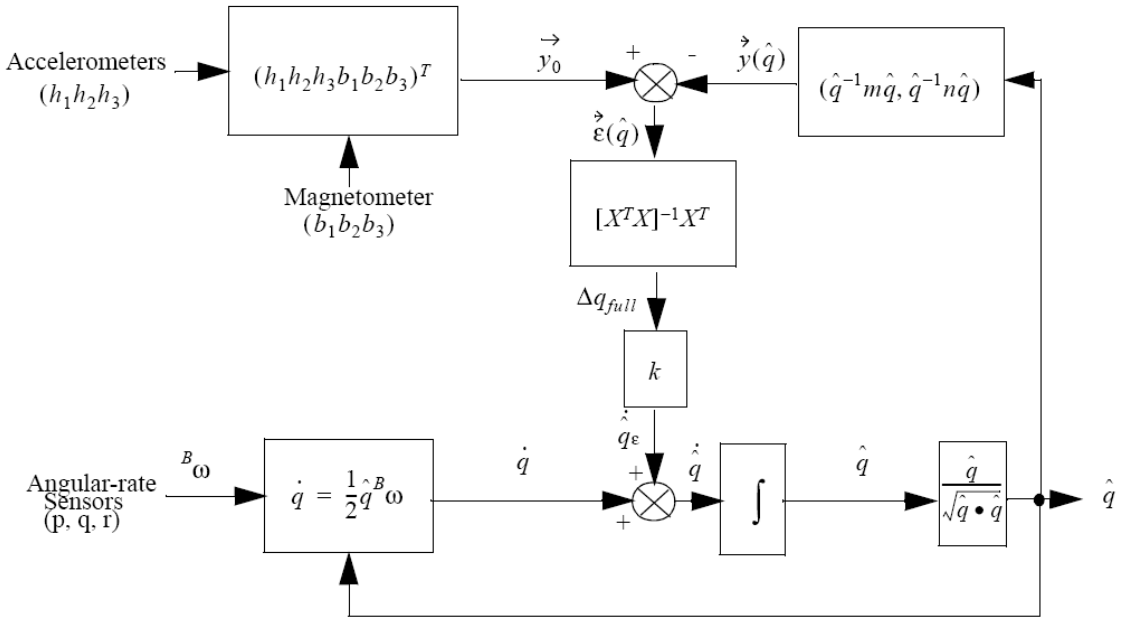


Figure 2.6 Quaternion-based attitude filter [18]

From the block diagram in Figure 2.6, the quarternion q is estimated based on the angular rate data collected from the gyroscope.

The rate quaternion can be obtained using the relationship

$$\dot{q} = q\left(0, \frac{1}{2}p, \frac{1}{2}q, \frac{1}{2}r\right) = \frac{1}{2}q\ {}^{b}\omega \tag{2.21}$$

where the indicated product is a quaternion product and ${}^{b}\omega$ is the angular rate measured by the gyroscope in the body coordinate [18].

The new estimated quanternions q_{n+1} can be calculated base on q at this step, which is [18]:

$$\begin{aligned}\hat{q}_{n+1} &= \hat{q}_n + \frac{1}{2}\hat{q}_n\ {}^{B}\omega\Delta t + \alpha\left[X^T X\right]^{-1} X^T \vec{\varepsilon}(\hat{q}_n) \\ &= q_n + k\Delta t\Delta q_{full} + q_{measured}\Delta t\end{aligned} \tag{2.22}$$

where Δq_{full} is the full correction step, $0 < \alpha < 1$, X is defined as

$$X_{ij} = \left[\frac{\partial y_i}{\partial \hat{q}_j}\right] \tag{2.23}$$

Once the quaternion is obtained for each step, the corresponding orientation can be calculated by Eq. (2.20).

In error free and noiseless environment, the unit quaternion q can be computed only using the gyroscopes. However, in reality, there is always error existing when the angular-rate sensors acquire data. This error will be accumulated at each step and make the orientation calculation result significantly away from the real value. Therefore, in order to correct the error, the accelerometer and magnetometer data are used to get the unchanged earth gravity and magnetic field direction. In each step, the error vector is

$$\vec{\varepsilon}(\hat{q}) = \vec{y_0} - \vec{y}(\hat{q}) \tag{2.24}$$

where $\vec{y_0}$ is the measured values of gravity vector and magnetic field, $y(q)$ is the computed values based on the estimation of current orientation (calculated in the last step). Once the error information is obtained, this error can then be corrected by using Gauss-Newton iteration.

The quaternion-based attitude filter is free from orientation singularities and can continuously correct the drifting error by the feedback. This property of filter makes it particularly applicable to human body capturing applications with highly linear acceleration. In order to improve the accuracy of this quaternion-based attitude filter,

many researchers began to consider other optimal algorithms by adding a Kalman filter into this structure. The results of experiment have shown a superior performance [35,36].

2.3.2 A Human Motion Capture Method Based on Homogeneous Matrix

A real-time articulated human motion capture algorithm based on Homogeneous matrix is proposed in [34]. As depicted in Figure 2.7, two connective segments are represented in two different body coordinate, where segment i has three orthogonal X_i-, Y_i- and Z_i-axis and segment $i+1$ also has three orthogonal X_{i+1}-, Y_{i+1}- and Z_{i+1}-axis. Both of them are constructed by a tri-axes accelerometer, a tri-axes magnetometer and a tri-axes gyroscope. K_i^{i+1} is the rotation axis with respect to which the segment $i+1$ was rotated. The angle θ is defined as the joint angle by which the segment $i+1$ was rotated with respect to segment i.

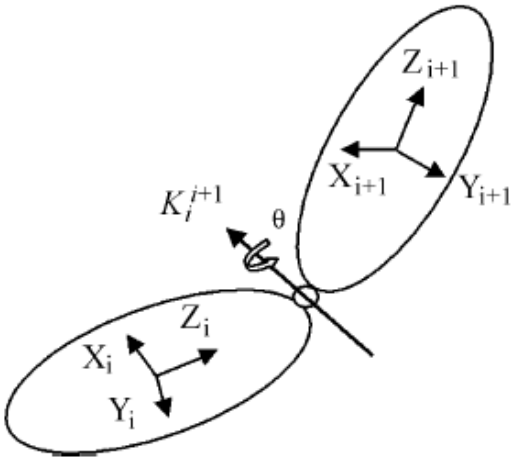


Figure 2.7 Physical segments model and the definition of their orthogonal coordinates [34]

Based on the orientation representation of the joint angle and the rotation axis, the rotation matrix $Rot(K_i^{i+1}, \theta)$ between the coordinate frames of the segment i and segment $i+1$ can be expressed as [30]

$$Rot\left(K_i^{i+1},\theta\right) =$$

$$\begin{bmatrix} K_x^2(1-\cos\theta)+\cos\theta & K_xK_y(1-\cos\theta)-K_z\sin\theta & K_xK_z(1-\cos\theta)+K_y\sin\theta \\ K_xK_y(1-\cos\theta)+K_z\sin\theta & K_y^2(1-\cos\theta)+\cos\theta & K_yK_z(1-\cos\theta)-K_x\sin\theta \\ K_xK_z(1-\cos\theta)-K_y\sin\theta & K_yK_z(1-\cos\theta)+K_x\sin\theta & K_z^2(1-\cos\theta)+\cos\theta \end{bmatrix} \tag{2.25}$$

where $K_i^{i+1} = \begin{bmatrix} K_x & K_y & K_z \end{bmatrix}$ denotes the rotation axis in coordinate system X_i-Y_i-Z_i. Let $\begin{bmatrix} g_x^i & g_y^i & g_z^i \end{bmatrix}^T$ and $\begin{bmatrix} g_x^{i+1} & g_y^{i+1} & g_z^{i+1} \end{bmatrix}^T$ be the gravity components in coordinates X_i - Y_i - Z_i and X_{i+1} - Y_{i+1} - Z_{i+1} measured by the accelerometers respectively. $\begin{bmatrix} m_x^i & m_y^i & m_z^i \end{bmatrix}^T$ and $\begin{bmatrix} m_x^{i+1} & m_y^{i+1} & m_z^{i+1} \end{bmatrix}^T$ are the earth magnetic field components measured by the magnetometers attached on segment i and segment $i+1$ respectively. Then, the rotation of segment $i+1$ can be denoted as

$$\begin{bmatrix} g_x^i & g_y^i & g_z^i \end{bmatrix}^T = Rot(K_i^{i+1},\theta)\begin{bmatrix} g_x^{i+1} & g_y^{i+1} & g_z^{i+1} \end{bmatrix}^T \tag{2.26}$$

and

$$\begin{bmatrix} m_x^i & m_y^i & m_z^i \end{bmatrix}^T = Rot(K_i^{i+1},\theta)\begin{bmatrix} m_x^{i+1} & m_y^{i+1} & m_z^{i+1} \end{bmatrix}^T \tag{2.27}$$

When considering the dynamic process, two time points t and $t+\Delta t$ are defined as a start point and an end point of the process. Assuming that the time gap Δt is very short ($\Delta t \to 0$) so that the rotation angle $\Delta\theta$ in Δt is close to zero ($\Delta\theta \to 0$). Then, $\cos\Delta\theta \approx 1$, $\sin\Delta\theta \approx \Delta\theta$ and the rotation matrix could be simplified as [34]

$$Rot\left(K_i^{i+1}(\Delta t),\Delta\theta\right) = \begin{bmatrix} 1 & -K_z^i(t)\cdot\Delta\theta & K_y^i(t)\cdot\Delta\theta \\ K_z^i(t)\cdot\Delta\theta & 1 & -K_x^i(t)\cdot\Delta\theta \\ -K_y^i(t)\cdot\Delta\theta & -K_x^i(t)\cdot\Delta\theta & 1 \end{bmatrix} \tag{2.28}$$

where $K_i^{i+1}(\Delta t) = \begin{bmatrix} K_x^i(t) & K_y^i(t) & K_z^i(t) \end{bmatrix}$ denotes the rotation axis of the segment i in the frame X_i-Y_i-Z_i.

According to Eq. (2.26) and Eq. (2.27), dynamic rotation of the segment i from time t to $t+\Delta t$ could be expressed by

$$\begin{bmatrix} g_x^i(t+\Delta t) & g_y^i(t+\Delta t) & g_z^i(t+\Delta t) \end{bmatrix}^T = Rot^{-1}(K_i^{i+1},\theta)\begin{bmatrix} g_x^i(t) & g_y^i(t) & g_z^i(t) \end{bmatrix}^T \tag{2.29}$$

$$\begin{bmatrix} m_x^i(t+\Delta t) & m_y^i(t+\Delta t) & m_z^i(t+\Delta t) \end{bmatrix}^T = Rot^{-1}\left(K_i^{i+1},\theta\right)\begin{bmatrix} m_x^i(t) & m_y^i(t) & m_z^i(t) \end{bmatrix}^T \tag{2.30}$$

where $Rot^{-1}\left(K_i^{i+1},\theta\right) = Rot^T(K_i^{i+1},\theta)$. Expressions $\begin{bmatrix} g_x^i(t) & g_y^i(t) & g_z^i(t) \end{bmatrix}^T$ and $\begin{bmatrix} g_x^i(t+\Delta t) & g_y^i(t+\Delta t) & g_z^i(t+\Delta t) \end{bmatrix}^T$ are the gravity components in X_i-Y_i-Z_i measured by the accelerometers at times t and $t+\Delta t$ respectively. $\begin{bmatrix} m_x^i(t) & m_y^i(t) & m_z^i(t) \end{bmatrix}^T$ and $\begin{bmatrix} m_x^i(t+\Delta t) & m_y^i(t+\Delta t) & m_z^i(t+\Delta t) \end{bmatrix}^T$ are the earth magnetic field components in segment i at times t and $t+\Delta t$ respectively. When $\Delta t \to 0$, consider Eq. (2.29) and Eq. (2.30) to yield an equivalent equation as follows [34]:

$$\begin{bmatrix} \dot{g}_x^i \\ \dot{g}_y^i \\ \dot{g}_z^i \\ \dot{m}_x^i \\ \dot{m}_y^i \\ \dot{m}_z^i \end{bmatrix} = \begin{bmatrix} 0 & \omega_z^i & -\omega_y^i & 0 & 0 & 0 \\ -\omega_z^i & 0 & \omega_x^i & 0 & 0 & 0 \\ \omega_y^i & -\omega_x^i & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \omega_z^i & -\omega_y^i \\ 0 & 0 & 0 & -\omega_z^i & 0 & \omega_x^i \\ 0 & 0 & 0 & \omega_y^i & -\omega_x^i & 0 \end{bmatrix} \begin{bmatrix} g_x^i \\ g_y^i \\ g_z^i \\ m_x^i \\ m_y^i \\ m_z^i \end{bmatrix} \tag{2.31}$$

where $\omega_x^i = K_x^i(t) \bullet \left(\frac{\Delta\theta}{\Delta t}\right)|_{\Delta t\to 0}$, $\omega_y^i = K_y^i(t) \bullet \left(\frac{\Delta\theta}{\Delta t}\right)|_{\Delta t\to 0}$ and $\omega_z^i = K_z^i(t) \bullet \left(\frac{\Delta\theta}{\Delta t}\right)|_{\Delta t\to 0}$ denote the components of the angular velocity measured by the gyroscope in the coordinate system X_i-Y_i-Z_i. Let $X^i = \begin{bmatrix} g_x^i & g_y^i & g_z^i & m_x^i & m_y^i & m_z^i \end{bmatrix}^T$, which can be used as the state vector in Kalman filter.

When considering the position of rigid bodies, a Homogeneous matrix is used here to describe position information, defined as

$$T_i^{i+1} = \begin{bmatrix} Rot(K_i^{i+1},\theta_i) & l_i \\ 0,0,0 & 1 \end{bmatrix} \tag{2.32}$$

where l_i is the physical length of segment i and θ_i is the joint angle. Then the orientation of connected segments can be described as products of these Homogeneous matrices.

This method involves in Homogeneous matrix to represent orientation and displacement of individual or connected segments. However, the mass matrices multiplication reduced the computation efficiency.

3 Euler Angle-Based Body Motion Capture Algorithm

As discussed in Chapter 2, both quaternion-based attitude filter and the Homogeneous matrices-based methods require three sensors: gyroscope, magnetometer and accelerometer. However, building these three sensors on one sensor board will significantly increase the complexity in both hardware technology and algorithm computation. Although there are some commercial products, which combined all the three sensors into one sensor node, available nowadays, the price is usually very high. In order to reduce the cost and the complexity and unsure reasonable accuracy, a motion capture system using tri-axis accelerometers and tri-axis gyroscopes to track ten ADLs is presented here. Instead of quaternion-based attitude filter or the Homogeneous matrix-based methods, Euler angle-based body motion capture algorithms are used with the feature of simple, intuitive and usually have low computation complexity. In this chapter, two Euler angle-based body motion capture algorithms are discussed.

Euler angles and rotation matrix are widely used in representing dynamic orientation changing. Each rotation can be written as a rotation matrix. If a vector rotates to different directions, it can be represented as a vector multiplied by corresponding matrices with certain order. Euler angles are generally much more familiar to users, their values are more intuitive and predictable and those angles can be easily converted from the rotation rates collected from the gyroscopes. The main consideration of this method is the singularity because all the elements in rotation matrix are trigonometric functions. In mathematics, singularity usually happens when the denominator is zero and the numerator equals to a constant, or some functions such as tangent functions which will jump from $+\infty$ to $-\infty$ when rotation angles cross $\frac{\pi}{2}$. However, the Euler-Angle based

algorithms discussed here do not involve in tangent functions and all the ADLs are normal activities. In the experiment, the sampling rate of the sensors is 50 samples per second. Since the rotation speed of a body segment can barely reach to 90 degrees per sample, there is no singularity occurring.

3.1 Motion Capture using Single Rotation Order Algorithm

The basic test environment is simple. A test subject wearing multiple sensor nodes performs specific actions which can be tracked, logged and monitored. For each motion, the test subject has a start posture; this initial position of segments can be represented as a set of vectors in earth coordinate as ${}^{e}V_i$, $i = 0, 1, 2, \cdots$, n represent different samples. Since the sensor(s) attached on the body segments will change together with body segments during movement, the body segment vector ${}^{b}V_i$, where $i = 0, 1, 2, \cdots$, n is the samples, in the body coordinate system is fixed at all times during movement. Neglecting minor offsets in sensor orientation on a body segment, the body coordinate is equal to the earth coordinate prior to movement, i.e., ${}^{e}V_0 = {}^{b}V_0$.

3.1.1 Euler Angle Rotation between Earth Coordinate and Body Coordinate

Figure 3.1 shows the three basic rotations between the earth coordinate and the body coordinate. Where, roll, ϕ, is the rotation referenced to the X-axis, pitch, θ, is the rotation referenced to the Y-axis and yaw, ψ, is the rotation referenced to the Z-axis.

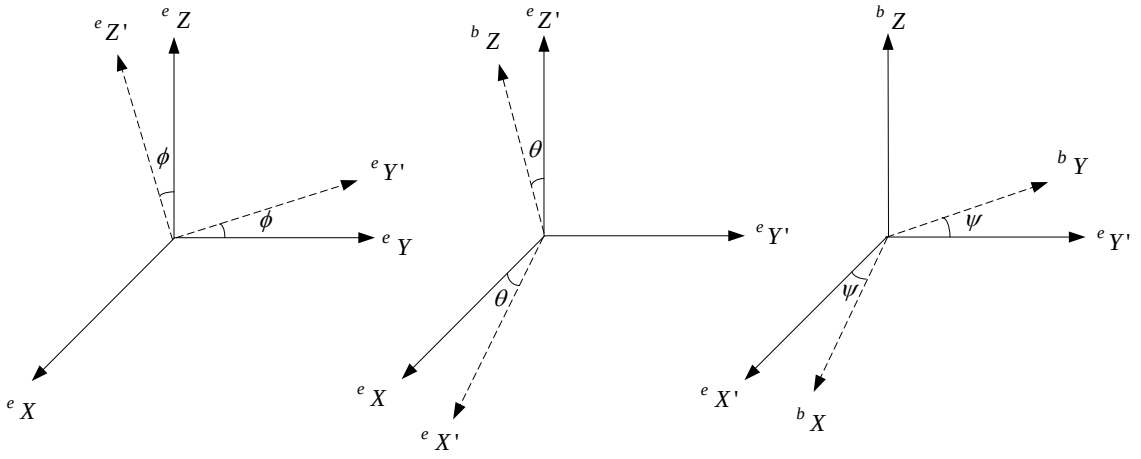


Figure 3.1 Coordinate transform between earth and body coordinate

Using the earth reference coordinate system as ^{e}X, ^{e}Y, ^{e}Z and the body coordinate system as ^{b}X, ^{b}Y, ^{b}Z, the corresponding rotation matrices are given by:

$$R(X,\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi & -sin\phi \\ 0 & sin\phi & cos\phi \end{bmatrix} \tag{3.1}$$

$$R(Z,\psi) = \begin{bmatrix} cos\psi & -sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.2}$$

$$R(Y,\theta) = \begin{bmatrix} cos\theta & 0 & sin\theta \\ 0 & 1 & 0 \\ -sin\theta & 0 & cos\theta \end{bmatrix} \tag{3.3}$$

where, $R(Z,\psi)$ is the rotation matrix for the rotation of the Z-axis by an angle ψ, $R(Y,\theta)$ is the rotation matrix for the rotation of the Y-axis by an angle θ and $R(X,\phi)$ is the rotation matrix for the rotation of the X-axis by an angle ϕ. The Euler angles are conventionally defined such that a segment first rotates the X-axis coordinate by ϕ, then rotates the Y-axis coordinate by θ and finally rotates the Z-axis coordinate by ψ. Therefore, the total rotation matrix can be written as ${}^{e}_{b}R$ which means rotate the body coordinate with respect to the reference coordinates (earth coordinate):

$$
\begin{aligned}
{}^{e}_{b}R &= R(Z,\psi)R(Y,\theta)R(X,\phi) \\
&= \begin{bmatrix} cos\psi & -sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos\theta & 0 & sin\theta \\ 0 & 1 & 0 \\ -sin\theta & 0 & cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi & -sin\phi \\ 0 & sin\phi & cos\phi \end{bmatrix} \\
&= \begin{bmatrix} cos\psi\, cos\theta & -sin\psi\, cos\phi + cos\psi\, sin\theta\, sin\phi & sin\psi\, sin\phi + cos\psi\, sin\theta\, cos\phi \\ sin\psi\, cos\theta & cos\psi\, cos\phi + sin\psi\, sin\theta\, sin\phi & -cos\psi\, sin\phi + sin\psi\, sin\theta\, cos\phi \\ -sin\theta & cos\theta\, sin\phi & cos\theta\, cos\phi \end{bmatrix} \quad (3.4)
\end{aligned}
$$

Also, this rotation matrix can be used to describe the coordinate transformation of a vector in different coordinate systems. Assuming that a vector ${}^{b}V$ is fixed in body coordinate and it will rotate with the body coordinate system. Then, the coordinates of ${}^{b}V$ in earth system, represented as ${}^{e}V$ is calculated as

$$
{}^{e}V = {}^{e}_{b}R^{-1} \; {}^{b}V \tag{3.5}
$$

or

$$
{}^{b}V = {}^{e}_{b}R \; {}^{e}V \tag{3.6}
$$

Eq. (3.5) and Eq. (3.6) reflect the relationship between the body and the earth coordinate systems in human body motion capture applications. The algorithms in this chapter are based on these basic equations.

3.1.2 Single Rotation Order Algorithm

Figure 3.2 shows the block diagram describing the orientation tracking of a body segment using a gyroscope. The angular rates collected by the gyroscope are denoted as p, q and r corresponding to the X, Y and Z axes. Upon integration (in time), these angular rates are converted to rotation angles, i.e., pitch θ, yaw ψ and roll ϕ. The rotation matrix is based on these rotation angles and the new orientation can be calculated.

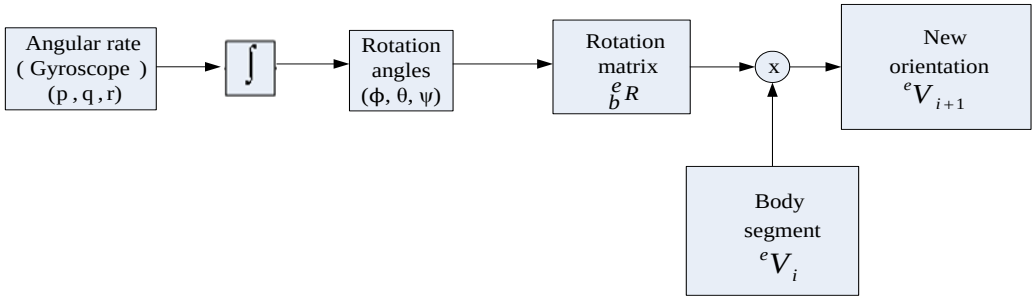


Figure 3.2 Block diagram of the single rotation order algorithm

At step i, the angular rates p_i, q_i, r_i are collected by the gyroscope sensor and the corresponding angles are given by

$$\theta_i = \int_{t_i}^{t_{i+1}} p_i dt \tag{3.7}$$

$$\psi_i = \int_{t_i}^{t_{i+1}} q_i dt \tag{3.8}$$

$$\phi_i = \int_{t_i}^{t_{i+1}} r_i dt \tag{3.9}$$

and the rotation matrix R_i at step i is

$$\begin{aligned} R_i &= R(Z,\psi_i)R(Y,\theta_i)R(X,\phi_i) \\ &= \begin{bmatrix} \cos\psi_i & -\sin\psi_i & 0 \\ \sin\psi_i & \cos\psi_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_i & 0 & \sin\theta_i \\ 0 & 1 & 0 \\ -\sin\theta_i & 0 & \cos\theta_i \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi_i & -\sin\phi_i \\ 0 & \sin\phi_i & \cos\phi_i \end{bmatrix} \end{aligned} \tag{3.10}$$

Since the body segment vector is fixed in the body coordinate, its orientation in the earth coordinate for current sample can then be estimated as:

$$^{e}V_i = R^{-1} \times {}^{b}V_i \tag{3.11}$$

where, R_i is the rotation matrix from the body coordinate to the earth coordinate, $^{e}V_i$ and $^{b}V_i$ are the body segment estimation for sample i in the earth coordinate and the body coordinate respectively. For each sample, three angular rate data are measured from the gyroscope and a corresponding rotation matrix can be calculated by these three data. Then, Eq. (3.11) can be rewritten as:

$$^{e}V_i = R_i^{-1} \times R_{i-1}^{-1} \cdots R_0^{-1} \times {}^{b}V_0 \tag{3.12}$$

Setting $^{e}_{b}R = R_i^{-1} \times R_{i-1}^{-1} \cdots R_0^{-1}$, then:

$$
\begin{aligned}
{}^{e}V_{i+1} &= {}^{e}_{b}R \times {}^{b}V_{0} \\
&= {}^{e}_{b}R \times {}^{e}V_{0}
\end{aligned} \tag{3.13}
$$

Therefore, every current location of the vector can be calculated by the original vector multiply to a series of rotation matrices. This method is very straightforward since all the orientation information is converted directly by the rotation angles. Ideally, if the gyroscopes used in the sensor nodes have high enough sensitivity and accuracy, the orientation tracking can be achieved using the angular rate data alone. However, with the limitation of sensitivity and environmental disturbances, gyroscope data usually contain noise and drifting, which lead to accumulated error. Meanwhile, it is difficult to determine the rotation sequences. Here, assume that the rotation is first with X axis, then with Y axis and lastly rotates with Y axis. In reality, however, since the three updated angular rates are obtained at the same time, it is hard to say which axis is rotating firstly.

3.2 Motion Capture using Optimal Rotation Order Algorithm

In the above single rotation order algorithm, the rotation is assumed with X axis first, then Y axis and lastly with Z axis. However, this is not always the case in real human body movement. Although different order can be chosen before simulation, it is always fixed and unchangeable during the calculation. In this section, how the order affects the orientation detection results will be discussed, as well as the optimal rotation order algorithm which has better performance.

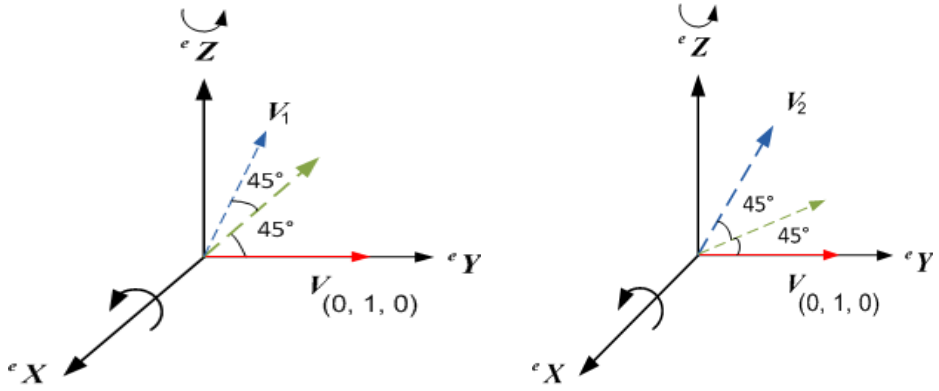
3.2.1 How the Rotation Orders Affect Orientation

Inertial motion capture systems utilize accelerometers and gyroscopes to determine the orientation of a body. A motion of such a body can be viewed as a series of rotations of different body segments. Angular rate data collected by a gyroscope contain the rotation information of a given body segment. After time-integration, an angular rate is converted to an angle traversed in rotation. There are three axes corresponding to three Euler angles: pitch θ, yaw ψ and roll ϕ as shown in Figure 3.1. If a rotation sequence is

setting as: firstly rotates the X-axis by ϕ, then Y-axis by θ and finally rotates the Z-axis by ψ, the corresponding rotation matrix is given in Eq. (3.4).

Euler angle sequences can be divided naturally in two classes: type 1 sequences have no repeating axes (i.e., XYZ, YZX, ZXY, ZYX, XZY, YXZ); type 2 sequences repeat the first axis as the last axis (i.e., XYX, XZX, YXY, YZY, ZXZ, ZYZ). There are in total 12 possible rotation sequences. Since all the three axes are considered for each sample in the experiment, only the type 1 sequences are discussed here.

Figure 3.3 shows an example of how the rotation order can affect the rotation vector. Consider a vector starting at orientation $V = (0, 1, 0)$. In Figure 3.3 (a), V is first rotated 45 ° about the X-axis, then 45 ° about the Z-axis. This yields a resultant transformed vector, $V_1 = (0.5, 0.5, -0.707)$. In Figure 3.3 (b), reverse the order of rotation, i.e., first rotate V 45 ° about the Z-axis and then 45 ° about the X-axis to obtain $V_2 = (0.707, 0.5, -0.5)$. It is clear that V_1 is different from V_2 due to the order of rotations.



(a) Rotate V 45 °with X-axis then rotate 45 °of Z-axis

(b) Rotate V 45 °with Z-axis then rotate 45 °of X-axis

Figure 3.3 Rotate vector V by different orders

3.2.2 Optimal Rotation Order Algorithm

Since the rotation order can affect the orientation detection results significantly, a way to determine the order how the body segments rotate is necessary. The combination of gyroscopes and accelerometers is widely used in motion capture applications. Since

the direction of earth's gravitational field does not change with time, it provides a reference via an accelerometer to check the result of the gyroscope orientation estimation. One important consideration in using angular rate for motion capture is the sequence of rotations or the rotation orders.

In an experiment, the gyroscope sensors collect the three angular rates for X, Y, Z axes at the same time, thus it is difficult to determine the order by which the rotations occurred. A block diagram of the orientation tracking procedure implemented using a gyroscope and an accelerometer is given in Figure 3.4.

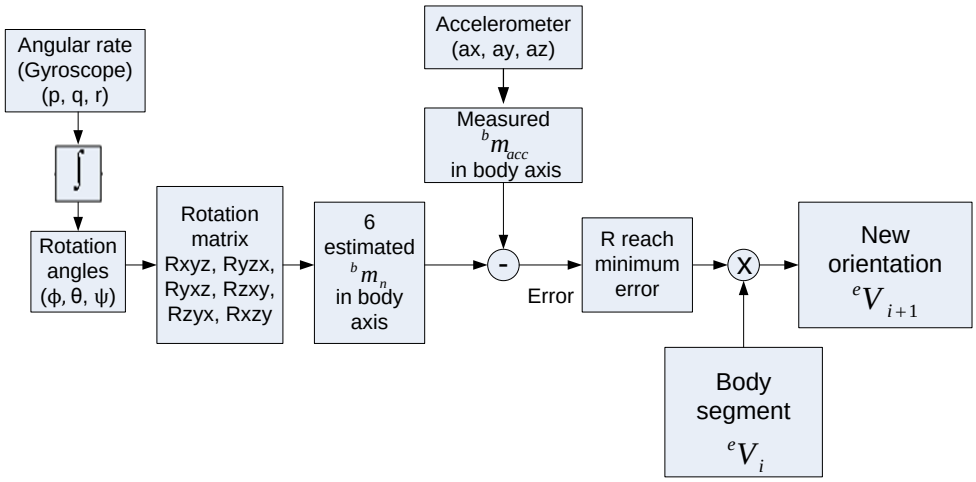


Figure 3.4 Block diagram of optimal rotation order algorithm

In Figure 3.4, all six possible rotation sequences are taken into account, that is: X-Y-Z, Y-Z-X, Y-X-Z, Z-X-Y, Z-Y-X, X-Z-Y. Therefore, six different rotation matrices can be generated according to the above sequences, which are denoted by R_{xyz}, R_{yzx}, R_{yxz}, R_{zxy}, R_{zyx} and R_{xzy}. Let ${}^{e}m$ refer to the direction of gravity in earth coordinate and ${}^{b}m$ represents the earth gravity estimated in a body coordinate system. The direction of gravity is a fixed vector in earth coordinate. By multiplying ${}^{e}m$ by all six possible rotation matrices (independently), six estimated gravity vectors for sample *i* is obtained by:

$$ {}^{b}m_{i,n} = R_n \times {}^{b}m_{i-1} \tag{3.14}$$

where, ${}^{b}m_{i,n}$, $n = 1, 2, \cdots, 6$ are the six estimated gravity vectors at step i in body coordinate, R_n, $n = 1, 2, \cdots, 6$ are the rotation matrices for six possible rotation orders. Defining ${}^{b}m_{acc}$ as the direction of gravity measured by accelerometer, the error due to an estimated gravity vector is approximately given by:

$$Error_n = {}^{b}m_{i,n} - {}^{b}m_{acc} \tag{3.15}$$

The rotation matrix R_i that yields the smallest error (defined above) is then used as the rotation matrix at step i. The orientation, ${}^{e}V_{i+1}$ for the next step, $i + 1$, is calculated as

$${}^{e}V_{i+1} = R_i^{-1} \times {}^{e}V_i \tag{3.16}$$

Compared to single rotation order algorithm, optimal rotation order algorithm makes use of the data obtained from accelerometer to detect the earth gravity direction which serves as a reference to estimate the real rotation sequence. Consequently, it will get more accurate motion capture results.

4 Body Motion Capture System Architecture

The inertial wireless sensors used in this motion capture research is developed on the latest version of SHIMMER (Sensing Health with Intelligence Modularity, Mobility and Experimental Reusability) platform. SHIMMER is a small wireless sensor platform launched by Realtime Technologies Ltd. Designed as a wearable sensor unit, SHIMMER can incorporate wireless ECG (Electrocardiogram), EMG (Electromyogram), GSR (Galvanic Skin Response) accelerometer, gyroscope, tilt and magnetic sensors and records and transmits physiological and kinematic data in real-time [37].

In this research, a 3-axis accelerometer and a 3-axis gyroscope are incorporated in a SHIMMER unit to detect motion data. This section provides an overview of the SHIMMER hardware architecture and discusses the hardware sub-systems contained. The software used in SHIMMER platform is also introduced.

4.1 SHIMMER System Hardware Diagram

Figure 4.1 illustrates a block diagram of the SHIMMER baseboard interconnections and integrated devices [38]. SHIMMER system consists of a low-power MSP430F1611 microcontroller, a three-axis accelerometer, a three-axis gyroscope, wireless communication module, IO and memory expansions (up to 2GByte MicroSD). The microcontroller controls the complete operation of the device and provides operational alerts and messages. Angular rate and earth gravity are measured by the gyroscope and the accelerometer respectively. The data can be sent to a computer via the wireless connection or stored in the MicroSD device. SHIMMER also has power control features including soft-power switching, power monitoring and low-battery shutdown.

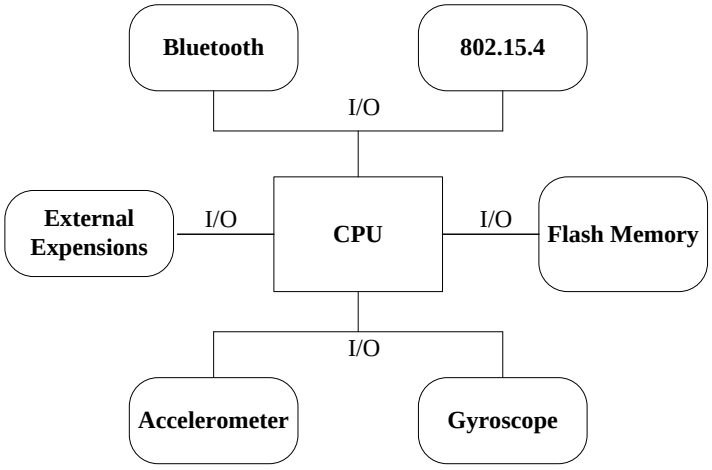


Figure 4.1 SHIMMER System Diagram

The detailed functional description of each subsystem will be discussed in this chapter. They are microcontroller, accelerometer, gyroscope, wireless communication module, IO and memory expansions.

4.1.1 MSP430F1611 Mixed-Signal Microcontroller

The processor used in SHIMMER system is the Texas Instruments 16-bit ultra-low-power microcontroller MSP430F1611 which is designed in portable measurement applications. It has two built-in 16-bit timers, a fast 12-bit ADC, dual 12-bit DACs, one or two Universal Synchronous/Asynchronous Receiver/Transmitter (USART), I^2C, DMA and 48 I/O pins. The MSP430F1611 microcontroller offers extended 10Kbyte RAM and 48Kbyte flash for memory-intensive applications and large C-stack requirements [39]. In this body motion capture system, the 8-channel 12-bit ADC is used to convert the data measured from the gyroscope from analog to digital signal. The three-axis gyroscope data and three-axis accelerometer data conversion use 6 channels of the microcontroller's built-in ADC.

4.1.2 MMA7361 3-axis MEMS (Micro Electro Mechanical systems) Accelerometer

The MMA7361L is a low-power, low-profile capacitive, 3-axis micromachined accelerometer produced by Freescale Semiconductor Inc.. Typical features include signal

conditioning, single-pole low pass filter, temperature compensation, self test, 0g-detect which detects linear freefall and g-select which allows for the selection between 2 sensitivities. Zero-g offset and sensitivity are factory set and require no external devices. The MMA7361L also has a sleep mode that makes it ideal for handheld battery powered electronics [40].

This tiny 3-axis accelerometer has already used in many different applications, such as tilt and motion sensing in 3-D gaming, image stability, text scroll and motion dialing in cell phone, e-compass tilt compensation in navigation and also falls and near falls detection on laptop or PC.

4.1.3 Gyroscope

The gyroscope is a SHIMMER extension module which provides 3-axis angular rate sensing with a rigid board implementation to ensure a perpendicular z-axis. It integrates dual-axis angular rate gyroscopes to perform complex motion sensing applications. The gyro board uses next-generation MEMS technology that offers higher performance. The gyro board is connected to the SHIMMER main board via an internal connector pin and is contained within the SHIMMER enclosure. With fixed reference output, the gyro board runs off a secondary Low Drop Out power supply regulator (LDO) for improved power-supply noise rejection [41]. Key features include [42]:

- Zero-signal reference levels for offset calibration
- Auto-zero control for precision sensing applications
- Gyro temperature monitoring for precision sensing
- Secondary power regular rejects system noise
- Programmable indicator
- Pinhole reset

The gyroscopes provide 500 degrees/sec full scale and 2.0mV/degree/second sensitivity. When combined with the SHIMMER's integrated accelerometer, both offer a full 6-degree freedom motion capture.

4.1.4 Wireless Communications Module

As a wireless platform, SHIMMER ensures the high quality of communication with both 802.15.4 and Bluetooth radio solutions, but the two radios cannot be operated simultaneously.

IEEE 802.15.4 is a standard which specifies the physical layer and media access control for low-rate wireless personal area networks (LR-WPANs) which focuses on low-cost, low-speed ubiquitous communication between devices [43]. The emphasis is on very low cost communication of nearby devices with little to no underlying infrastructure, intending to exploit this to further lower power consumption.

Bluetooth (IEEE 802.15.1) is a proprietary open wireless technology standard for exchanging data over short distances (using short wavelength radio transmissions) from fixed and mobile devices, creating personal area networks (PANs) with high levels of security [44]. The key features comparison of 802.15.4 and Bluetooth radio are shown in Table 4.1. Choice should be based on application needs and available resources.

Table 4.1 Features of 802.15.4 and Bluetooth radio [38]

Radio	802.15.4	Bluetooth
Power Profile	Years	Days
Data Rate	20 - 250 kbit/s	1 Mbit/s
Network Latency	15 – 30ms	2ms – 20s
Number of Devices per Network	2 - 65,000	8
Prebuilt Application	NO	YES
Network Range	10 - 100m	10m
Complexity	Simple	Complex
Security and Reliability	Very high	Medium

Although IEEE 802.15.4 is advantageous for telemetric system due to low energy consumption, wide range and tolerant of more devices, Bluetooth maintains higher data transfer rate. Since there are ten sensors and each sensor will transfer large amount of data to computer for real-time motion reproduction, Bluetooth is chosen here for

communications in this motion capture experiment. Most medical devices are compatible with Bluetooth now, which makes it easier to combine other medical devices for further research on near-fall detection.

4.1.5 IO and Memory Expansions

SHIMMER platform provides both internal and external expansions. Internal expansion is used for internal sensor daughter boards on which there are more sensors can be selected. External expansion is an 18-pin header, which can be used by user for charging, programming and tethered sensor extension. Through external expansion header, one SPI (Serial Peripheral Interface) and one UART (Universal Asynchronous Receiver/Transmitter) interface can be configured to access the data in MicroSD card. The MicroSD card can be installed manually on the SHIMMER baseboard to incorporate extra memory resources. The capacity is up to 2Gbytes, which allows the additional storage of data while the SHIMMER is in motion, during network outages or while changing batteries. The SD host data bypass function is also added to improve usability. It incorporates a wide bandwidth analog multiplexer and tri-state logic buffering on certain signals routed to the external connector to provide direct and immediate access to flash memory using an external SD-flash card controller for high-speed data transfer [38].

4.2 SHIMMER System Software Environment

The SHIMMER platform uses TinyOS operating system for implementation, testing and validation of SHIMMER embedded software (firmware). TinyOS is an open source component-based operating system designed for low power wireless devices [38,45]. The language written in TinyOS is called NESC (Network Embedded Systems C) [46], which is a component-based, event-driven programming language used to build applications for the TinyOS platform. It is built as an extension to the C programming language with components "wired" together to run applications on TinyOS. The functionality in TinyOS operating system on SHIMMER platform includes:

- MicroSD flash storage
- FAT file system

- IP stack for 802.15.4
- Bluetooth configuration, connection management and streaming data transfer
- Real time clock module
- Peripheral control and configuration
- Power supply monitoring

TinyOS is an embedded operating system written in the NESC programming language as a set of cooperating tasks and processes. This programming environment supports a variety of low power devices, with a few kilobytes of memory and wireless communication capabilities [45]. It is designed for networked sensors with minimal hardware requirements. NESC is an extension to C designed to embody the structuring concepts and execution model of TinyOS [47] and uses the custom NESC compiler.

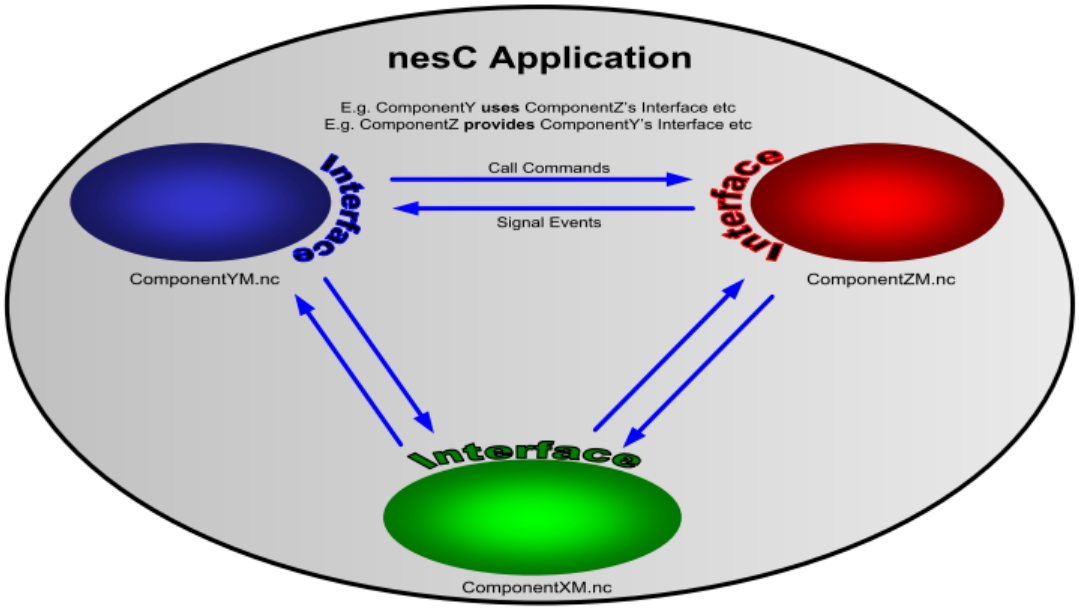


Figure 4.2 NESC application [40]

Figure 4.2 shows the basic idea of NESC application. An application is composed of one or more components linked together to form an executable. For each application, there is a top-level configuration that wires together the components inside. The basic elements in a NESC application are listed as follows:

Components: provide and use well-defined bidirectional interfaces to build blocks of NESC applications. Every component has a "specification", a code block that declares

the function it provides (implements) and the functions that it uses (calls). NESC has two kinds of components: configurations and modules.

Modules: provide application code, implementing one or more interfaces.

Configurations: assemble other components together, connecting interfaces used by components to interfaces provided by others, i.e., "wiring".

Interfaces: declare a set of functions called "commands" that the interface provider must implement and another set of functions called "events" that the interface user must implement. An interface is bidirectional and it acts as the only point of access to a component.

Commands: the functions that an interface provider must implement.

Events: declared by an interface provider but the user of the interface must implement them based on their requirements.

More detailed information of TinyOS and NESC can be found in [45-47].

5 Motion Capture Experiment

Since the FANFARE project is targeting at the elderly people fall and near-fall detection, theoretically, elderly people should be tested in the experiment. However, involving the elderly in the experiment may have some health considerations. For example, whether they can wear those sensors for the long time period experiment based on their body situations, is there any protection to prevent them from unexpected injuries during experiment. Instead, younger people are usually much stronger and healthier than elderly people, and studying on normal activities of youth can help doctors to understand the elderly behaviors. Meanwhile, the body motion capture system presented in the thesis is just a prototype comparing the goal of FANFARE project, which needs a plenty of experiments to verify its feasibility, reliability and security. In the future, once the system is well tested by a series of experiments and the methods are verified in young people, elderly people can be tested in the lab environment under the instructions of doctors. In experiment, two male subjects (subject #1 and #2) and one female subject (subject #3) between twenty and thirty years old are tested to perform the selected ADLs. In order to create a whole body avatar in computer, ten SHIMMER sensors are picked to amount on a human body and capture the motion data. Ten is a minimum number, based on which a whole body avatar can be generated. Adding more sensors into this system may increase the accuracy of the capture results, however, reducing the number of sensors will directly affect the recognition of ADLs. The sampling rate of all SHIMMER sensors is 50 samples per second. The procedure of the conducted motion capture experiment is shown in Figure 5.1.

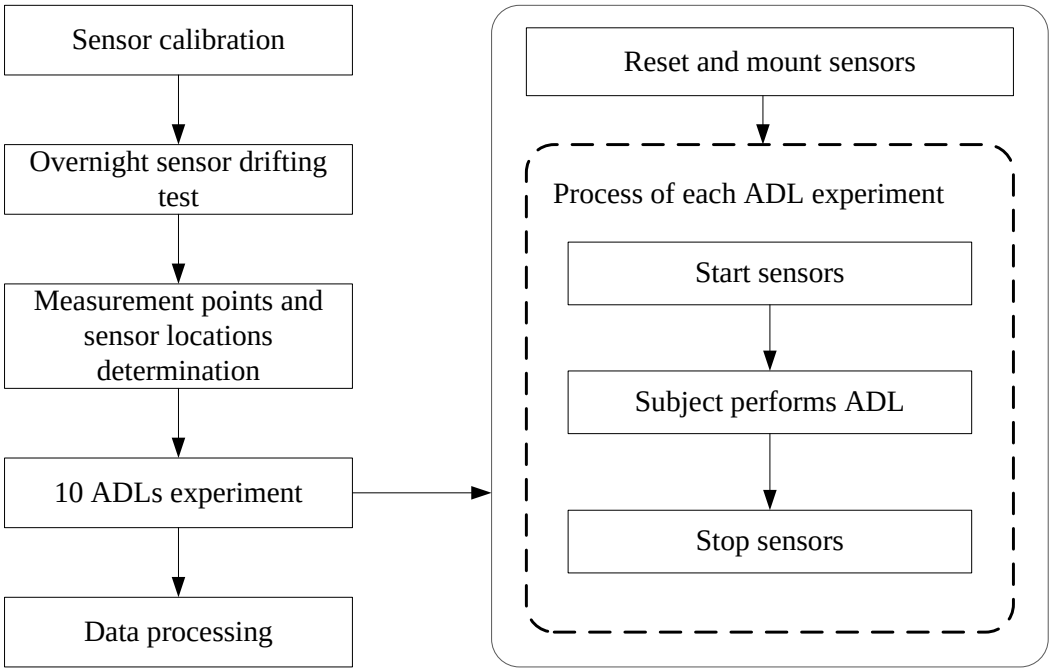


Figure 5.1 Flow chart of the human body motion capture experiment

Before the experiment, sensor calibration is completed to minimize the sensor error and record the scale parameters. After the sensor calibration, since the sensors are easily subject to data drifting errors and the accuracy of results cannot be ensured if the drifting is significant, it is necessary to do an overnight sensor drifting test. Also, the mounting locations of sensor nodes need to be decided prior to the experiment. The corresponding body segment measurements of the test subject need to be recorded for further work. Next, ten ADLs experiment is performed, during which the test subject follows the procedure outlined in the dashed block shown in Figure 5.1 for each ADL test. The remainder of this section is dedicated to a more detailed explanation of each of these steps.

5.1 Sensor Calibration

Sensor calibration is a critical step for the experiment which can minimize errors caused by sensors themselves. Calibrating a SHIMMER node with accelerometer and gyroscope means to obtain the parameters which can be used to convert the data collected by sensors into physically meaningful data with units. Ideally, this would be achieved

using the information available in each sensor's datasheet without any calibration. However, in reality, the measurements from sensors can vary from device to device. Possible reasons include normal manufacturing fluctuations, environment temperature and power supply voltage. Since there are two kinds of sensors: accelerometers and gyroscopes, the calibration for them will be performed separately.

5.1.1 Calibration for Accelerometer

For the calibration of an accelerometer, the three axes are calibrated separately [40,48,49]. When calibrating the accelerometer in one axis, firstly, leave the accelerometer sitting on a well-leveled table with the chosen axis pointing down for around twenty seconds. The acceleration due to earth gravity is measured as $A_{n,+1g}$. Then, flip the sensor to let that axis pointing up for twenty seconds and the measured value is $A_{n,-1g}$. Based on these two values, the expected value m_n corresponding to 0g (midpoint of the values of $A_{n,+1g}$ and $A_{n,-1g}$) can be easily calculated. This 0g value was recorded and it will be taken off from the real experimental value.

Since the output raw value of the accelerometer needs to be converted to real acceleration value $A_n g$, the following formula should be used

$$A_n g = \frac{2\times(A_n - m_n)}{(A_{n,+1g} - A_{n,-1g})} \tag{5.1}$$

where, $A_n g$ is the calibrated value from the accelerometer data and its unit is g ($1g$ equals to the gravity acceleration, 9.81 m/s^2). A_n is the accelerometer raw value for axis n. m_n is the calculated midpoint for axis n. Here, the values of m_n, $A_{n,+1g}$ and $A_{n,-1g}$are only valid for the specific axis for one specific sensor, they should be found out and recorded for each axis in each sensor during calibration.

5.1.2 Calibration for Gyroscope

Gyroscope measures angular velocity in unit degree/second. It can be calibrated by the following steps [50]:

First place the gyroscope on a flat table. Since the gyroscope is neither moving nor rotating, its angular velocity is 0. Take 20 seconds of measurements and calculate the mean values for each X, Y, Z components. Use the following formula to calculate the real value r in degrees/second:

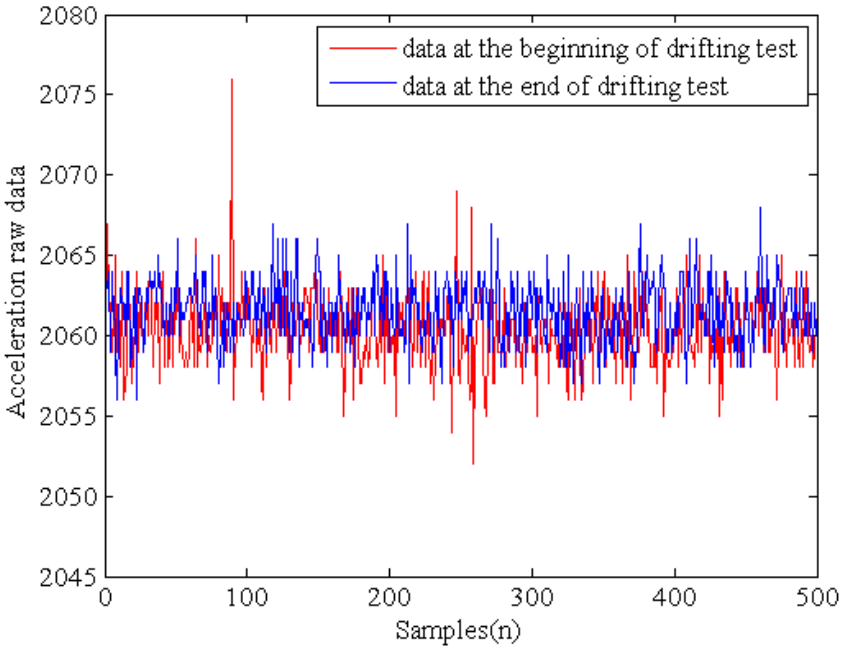
$$r = \frac{r_{adcOutput} - r_{staticOutput}}{2.731} \tag{5.2}$$

where, $r_{adcOutput}$ is the value returned by the motionless gyroscope and $r_{staticOutput}$ is 1843. This value is from manufactory, as well as number 2.731. Repeating the above steps for all ten sensors and recording the motionless value for each axis of each sensor, these values can be used to calculate the real rotation values in the experiment.

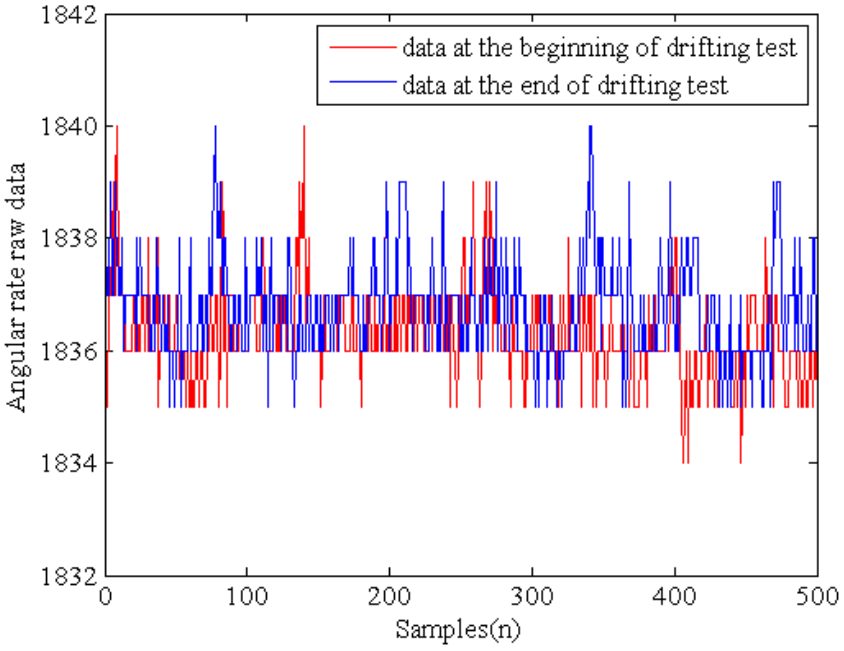
When all of the sensors are well calibrated and the corresponding values are correctly-recorded, the inertial sensors are ready for the experiment.

5.2 Overnight Sensor Drifting Test

After all of the sensors are calibrated, another factor which can significantly affect the sensor accuracy should be considered: drifting. An overnight drifting test is implemented before ADLs motion capture experiment. In this test, all ten SHIMMER sensors were active throughout the night to monitor the data drifting in an isolated environment. In the laboratory, the ten SHIMMER sensors were placed on a flat surface and left in one position without any movement during the drifting test. Then, the Bluetooth was opened on each sensor one-by-one to connect them for data recording. After connection, all sensors could continuously collect data into the on-board SD cards with the sampling rate of 50 samples per second. Then, keep those sensors running (writing data to an SD card) continuously for about 16 hours, from 05:28 pm – 09:54 am (16 hours and 26 minutes) next morning. This provided a total of about 2.96 million samples over the test period. Figure 5.2 (a) shows the X-axis data of the accelerometer of sensor #1 (see Table 5.2) and Figure 5.2 (b) shows the X-axis data of the gyroscope of the same sensor during the test period. In both figures, the red lines stand for the data sampled at the beginning of the drifting test while the blue lines mean the data sampled at the end of drifting test (16 hours later).



(a) Data in X-axis of accelerometer



(b) Data in X-axis of gyroscope

Figure 5.2 Raw data in X-axis collected by sensor #1

From the two figures above, it is clear that the data collected at the beginning of the overnight drifting test and the data collected at the end of that test are almost overlapped.

Obviously, there is not a significant drifting based on the test results. Table 5.1 shows the drifting test results of X axis of sensor #1.

Table 5.1 Average drifting in accelerometer and gyroscope of sensor #1

Sensor #1	X-axis acceleration	X-axis angular rate
Sensor sensitivity	0.0038g	0.3662°/sec
The average of the beginning data in Figure 5.2	2060.4752	1836.2929
The average of the end data in Figure 5.2	2061.7841	1836.7791
Error in raw data	-1.3089	-0.4862
Error in real data	0.005g	0.178°/sec

For all other sensors, drifting test shows the similar results, which indicates that the sensors used in the experiment have reasonable drifting performance and the data collected from sensors have high accuracy.

5.3 Body Measurement Points and Sensor Node Locations Determination

In experiment, sensors need to be mounted properly and firmly on subjects' body segments. The selection of sensor node placements on the human body is critical to the experiment since the movements of different segments of a human body vary significantly. Prior to mounting the sensor nodes on the test subject's body, the length of each body segment is measured and recorded to determine the optimal sensor locations. Figure 5.3 shows the measurement points and sensor mounting positions of all ten sensors. Those sensor locations are picked under doctors' instructions to make subjects feel comfortable when wearing them during the motion and to reduce the influence of body deformation in movements. In experiment, all sensors are attached on the test subject's body using straps.

The sensor nodes are represented by the ten rectangular boxes, they are: 2, 4, 8, 10, 16, 12, 13, 18, 21 and 23. To simplify the experiment, each sensor node is matched to a certain body segment location and do not alternate the sensor locations for a different subject. The numbering scheme of each measurement point and sensor location in Figure

5.3 is shown in Table 5.2. According to Figure 5.3, all of the sensors should be attached firmly at the middle point of body segments. Monitoring a sensor at the middle point of the segment ensures the least number of sensors to track the whole body motion capture.

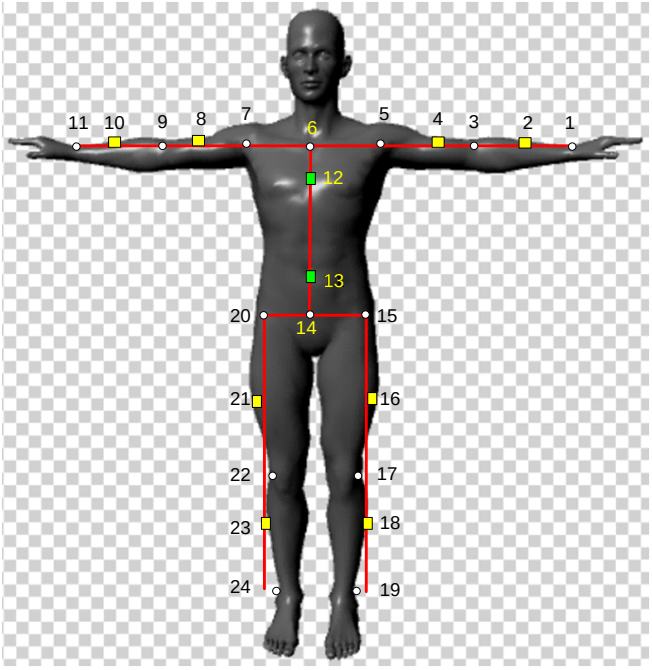


Figure 5.3 Sensor locations and the measurement points [51]

Table 5.2 Measurement points and sensor locations

Sensor No.	Number	Location	Sensor No.	Number	Location
-	1	Left wrist	Sensor #6	13	Waist
Sensor #3	2	Left lower arm	-	14	Hip
-	3	Left elbow	-	15	Left hipbone
Sensor #7	4	Left upper arm	Sensor #4	16	Left thigh
-	5	Left shoulder	-	17	Left knee
-	6	Upper chest	Sensor #10	18	Left calf
-	7	Right shoulder	-	19	Left ankle
Sensor #2	8	Right upper arm	-	20	Right hipbone
-	9	Right elbow	Sensor #1	21	Right thigh
Sensor #5	10	Right lower arm	-	22	Right knee
-	11	Right wrist	Sensor #8	23	Right calf
Sensor #9	12	Back	-	24	Right ankle

Figure 5.3 shows the length information of body segments measured on subject #1. For example, the shoulder length is the measured distance between point 5 and point 7 as in Figure 5.3. These length data will be used in the computer avatar model building and are different for each subject.

Table 5.3 Length of body segments for subject #1

Body segment	Length (cm)	Body segment	Length (cm)
Shoulder width (5-7)	39	Hip width (15-20)	28
Right upper arm (7-9)	30	Right thigh (20-22)	38
Right lower arm (9-11)	28	Right calf (22-24)	42.5
Left upper arm (3-5)	30	Left thigh (15-17)	38
Left lower arm (1-3)	28	Left calf (17-19)	43
Body trunk (6-13)	53	Hip height (13-14)	6

5.4 Body Motion Capture Experiment for Ten ADLs

After the calibration and drifting test, all sensors are ready for the motion capture experiment. The experiment is conducted in a physical therapy laboratory. Where, two male subjects (referred to subject #1 and subject #2) and one female subject (referred to subject #3) are tested to perform all ten selected ADLs. Since three subjects are tested with the same ADLs in the experiment, only the data collected from subject #1 is used for the motion reproduction in data processing which will be discussed later.

To simulate the most common activities in people's daily life, the following ten ADLs are chosen to conduct the experiment since most of the people's daily activities consist of one or more of these ADLs.

(1) Initially lying to standing up
(2) Initially standing to sitting
(3) Initially sitting, then bending the body to tie one's shoes
(4) Initially standing, then walking forward, turning around and finally walking backward
(5) Initially standing, then bending the body to pick up an object (ball) from the floor
(6) Initially standing, then using the right hand to reach forward

(7) Initially standing, then using the right hand to reach upward

(8) Initially standing, then stepping on a stool, holding for 5 seconds and finally stepping down from the stool

(9) Initially standing, then climbing up a stairs with 11 steps, then turning back and holding for 5 seconds and finally stepping down from the stairs and turning back again

(10) Perform the following motion sequence: initially standing, then lie down, stand up, sit down, tie shoes, stand up, walk forward, bend to pick up an object (ball) from the floor, reach forward, reach upward, step onto and down from a stool.

Before the experiment, the length of body segments should be properly measured and recorded (see Table 5.3) and the locations where the ten sensors should be mounted need to be established, as in Figure 5.3. The experiment for each subject can be divided into four steps: reset and mount sensors, start sensors, subject performs ADLs and stop sensors. Where, step (1) is done once and step (2) to step (4) are performed for each ADL.

Step (1) Reset and Mount Sensors

The ten sensors are reset simultaneously before they are attached to subject's body segments. As mentioned in SHIMMER hardware description, the 16-bit counter in each sensor is started to count and generate timestamps by this parallel reset. The timestamps roll over for every 2.05 seconds and will be annotated to the data when the sensor starts to collect data. Since all sensors are reset at the same time, during ADLs experiment, the timestamps generated by different sensors should be the same. This timestamp information is very important for data synchronization in data processing, which will be discussed later. When all of the sensors are synchronized reset, they are attached to the subject's body segments at the locations determined in previous sections. Once the sensors are mounted, they will stay attached to subject's body segments till the whole ten ADLs experiment is done for this subject.

Step (2) Start sensors

After all of the sensors are attached on the body, they need to be connected first before the subject performs any ADL. At the beginning, the subject keeps at the initial posture waiting for the central host computer sending "start" command to each sensor using a Bluetooth network. Usually, the Bluetooth network can only connect seven

terminals (sensor nodes) at the same time and the connection slows down with an increasing number of terminals. Since there are ten sensors in total, the approach used in experiment is that the commands are sent from the central host computer to the sensors one at a time to reduce individual computer-sensor latency. Once one sensor receives "start" command from the host computer, it starts to measure acceleration and angular rate data and record to the SD card. After this sensor node is started, the connection with host computer will be terminated and the next sensor will get connected the same way. When all of the sensors are started, the first step is finished.

Step (3) Subject performs ADL

Usually, the test subject should stay in the starting posture for five seconds to make sure all ten sensors are started, mitigate drifting differences between sensors and even out their independent data fluctuations. After that, the subject will keep the same posture for another five seconds and the data in this period will be used in zero error compensation in data processing. Then, the subject begins to perform the specific ADL and repeat it for five times (except the tenth ADL). Between two repeated movements in one ADL, the subject keeps at starting posture for five seconds, based on which each repeated movement can be easily distinguished. For example in ADL No. 2 experiment, when the subject finish sitting down and standing up motion, he/she will keep the standing posture and wait for another five seconds, then repeat for five times, finally return to the original posture. In ADL No. 10, five-second interval also exists between different movements as listed above. Then the whole ADL sequence is repeated for three times. For each time, repeat step (2) to (4) to make sure the collected data from repeated ADL No. 10 are stored in three different files.

Step (4) Stop Sensors

After the subject finish the ADL, a "stop" command is sent to each sensor one by one, just as the "start" command. The subject needs to stay at the initial posture for five seconds to ensure all of the sensors are disconnected.

In the physical therapy laboratory, there are a front camera and a side camera recording the subject's movements. The video of each ADL experiment is recorded by these two cameras. During the data processing, the visualized motion capture results will be compared to the video to determine if the motion capture is successful or not. Figure

5.4 shows a snapshot of the video recorded by the side camera when the subject is performing ADL No. 2.

(a) Stand up (b) Sit down

Figure 5.4 Experiment snapshot from the side camera

6 Experiment Results and Data Processing

After the experiment, the data stored in the SD card of each SHIMMER sensor node are processed to obtain subject's simulated movements. The sampling rate of the SHIMMER sensors is 50 samples per second. Since the sensors are initialized serially (turned on one-by-one during experiment), it is necessary to synchronize all of the data files prior to any processing and simulation. After the data of all sensor nodes had been synchronized, an optimal rotation order algorithm is applied to the collected data.

6.1 Data Pre-processing

The data stored in the SD cards are only raw data, they should be converted to physically meaningful data using parameters obtained in sensor calibration first. Then, more data pre-processing work needs to be done before body motion capture algorithms can be applied. They are data synchronization and zero error compensation.

6.1.1 Data Synchronization

The purpose of data synchronization is to find out the delay t (which is the time difference caused by two sensors starting at different time) based on the ADL movement start points in the data collected from sensors. The main source of un-synchronization of the data is that the sensors were not started at the same time and consequently they started to store data in the SD card at different time. Therefore, there is a time delay t between the data collected by any two sensors. Ideally, if all the body segments start to move at exactly the same time during one ADL, to synchronize the data and discard this time delay t, a threshold can be set to determine where the movement starts in the sensor data files. In reality, however, different body segments do not start to move at exactly the

same time, there is a time difference t_1 between two body segments starting to move. For instance, during walking ADL, subject may move right leg first and then left leg. Although this time difference is normally small (less than 2 seconds), it still cannot be neglected, which makes the synchronization difficult to achieve.

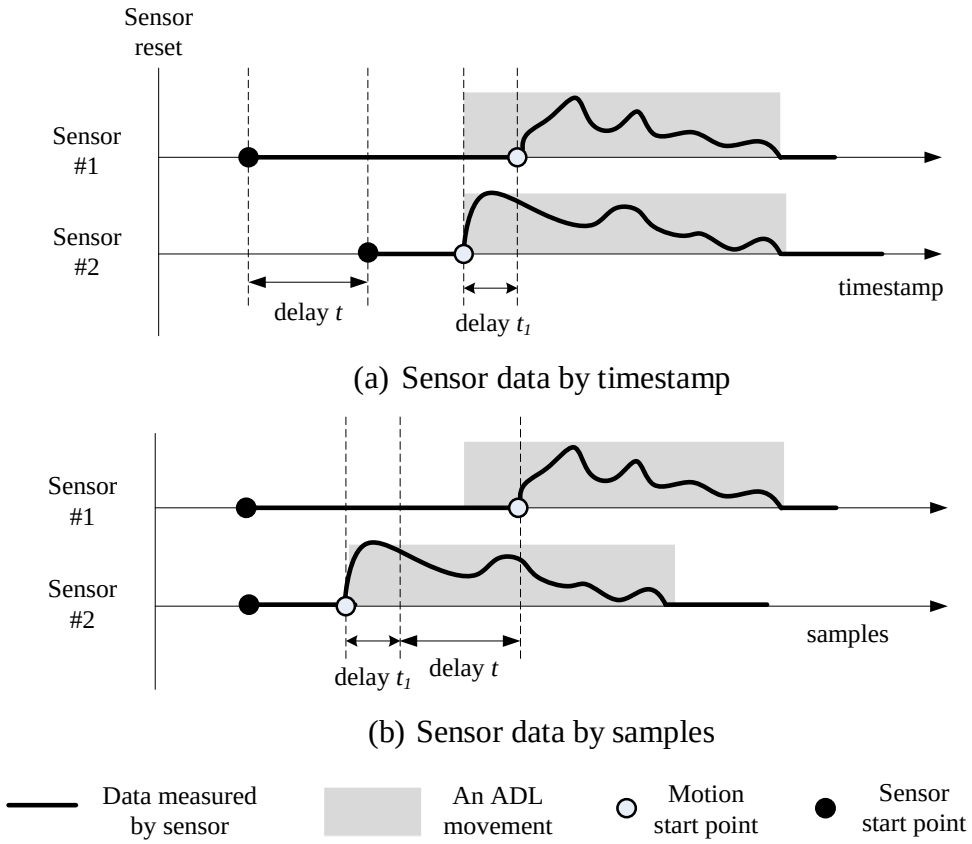


Figure 6.1 Data collected from sensor #1 and sensor #2

Figure 6.1 shows the example of data collected from sensor #1 and sensor #2. The bold waveforms represent acceleration or angular rate data stored in the SD cards in an ADL experiment. Motion start point is the data sample where the body segment starts to move. Sensor start point is the data sample where the sensor starts to record data to its SD card . For each sample of the data, there is a timestamp associated. The timestamps are generated by the internal counters in sensors which roll over at a period of around 2 seconds. Since the sensors (hence the counters) were started by resetting simultaneously, all the data samples measured at the same time in different sensors should be associated with the same timestamp. This is an important clue for the fine data synchronization (will be discussed in detail later). In Figure 6.1 (a), at the beginning, sensor #1 and 2 were reset

to count at the same time, but sensor #1 was started first by computer via wireless connection. Therefore its sensor start point is actually earlier than that of sensor #2 by delay t. Meanwhile, the body segment on which sensor #2 was attached moved earlier than that of sensor #1 by t_1. The target of synchronization is to find out the delay t. Since there is no obvious time information in the data, the synchronization could only base on the data samples. Figure 6.1 (b) shows the sensor data by samples, where the motion start point is the key point to find delay between two sensors. A two-step synchronization scheme will be discussed. The first step is coarse synchronization which can find the delay $t + t_1$ based on motion start points in different sensor data and the second step is fine synchronization which can find the delay t_1 based on timestamps. Then, the delay t can be easily calculated. By removing the delay t in sensor #1, these two sensors are well synchronized.

Here, since the accelerometer and gyroscope in one sensor node are started at the same time, their data are synchronous. Therefore only the angular rate data collected by gyroscope in sensor #1 and sensor #2 are used as example, shown in Figure 6.2. For the reason of simplicity, only X-axis data are shown.

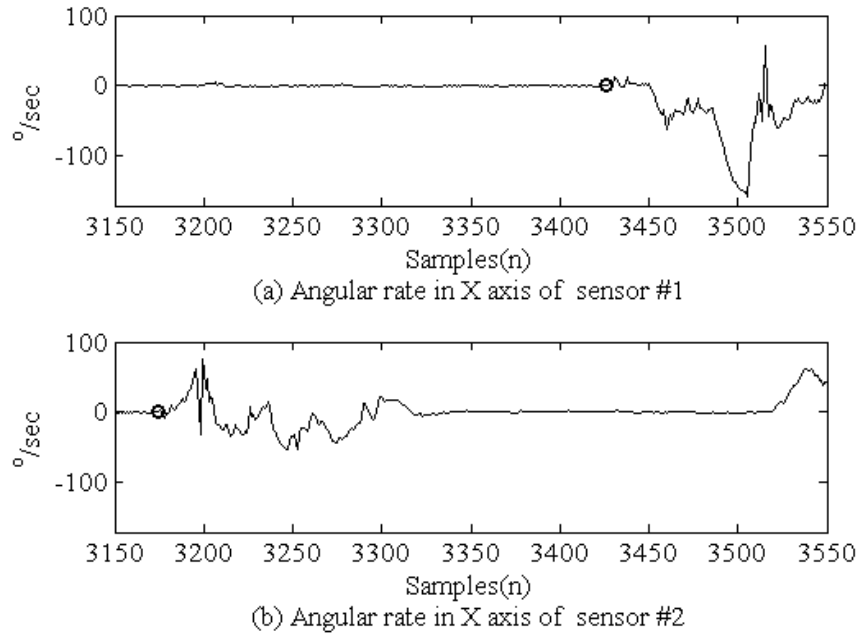


Figure 6.2 The comparison of X-axis angular rate data between sensor #1 (a) and sensor #2 (b)

In each figure, the circle highlights the movement start points of different sensors attached on the body segments. Obviously, there is a delay around 250 samples between two sensors.

In order to synchronize the data, a two-step synchronization scheme is applied, i.e. coarse synchronization and fine synchronization. The absolute value of X, Y, Z-axis angular rate data are added up to combine the information from all three axes and make the fluctuations more significant, as shown in Figure 6.3.

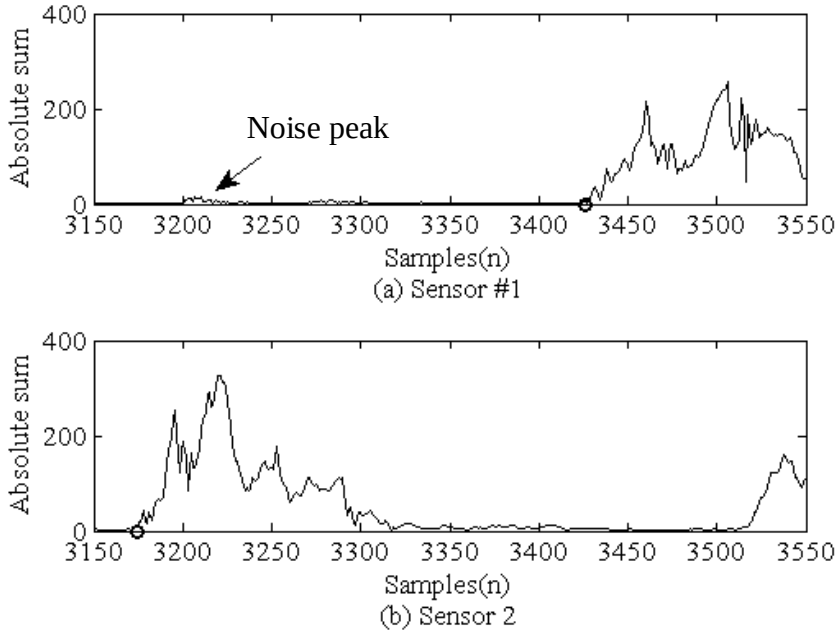


Figure 6.3 Absolute sum of X, Y, Z-axis angular rate data collected by sensor #1 (a) and sensor #2 (b)

1) Coarse Synchronization

During coarse synchronization, the point (sample) when each individual body segment starts to move needs to be found out by analyzing the data stored in SD card. To find the start point, two thresholds are set, H1 and H2. If a sample data is larger than H1, it is a possible start point, noted as X. If the second condition

$$a1 \times Ave\{X, \ldots, X+20\} + a2 \times Ave\{X+21, \ldots, X+40\} > H2 \qquad (6.1)$$

is satisfied, then start point is X. Otherwise, point X will discarded since it is only a noise peak, like the one pointed by the arrow in Figure 6.3 (a). Here, variables $H1$, $H2$, $a1$,

*a*2 need to be carefully determined. The circles in the figure show the correct start point of each segment movement.

2) Fine Synchronization

After the start point of each sensor data file, i.e., start time of the movement of each body segment, is found, the whole body movement now can be synchronized easily. For fine synchronization, first pick the start point in one sensor as the reference (here, sensor #1 is chosen), then according to the start point obtained in coarse synchronization of other sensors, find the nearest point with the same timestamp as the reference timestamp. Once these points are obtained, the delay between two sensors is calculated by directly subtracting one point to the other. After that, discard the first part (with length equals to delay) of the data collected by the sensor which is connected earlier compared to the other one. By now, these two sensors are synchronized. Then, apply this method to all the data and synchronized them for an ADL, as shown in Figure 6.4.

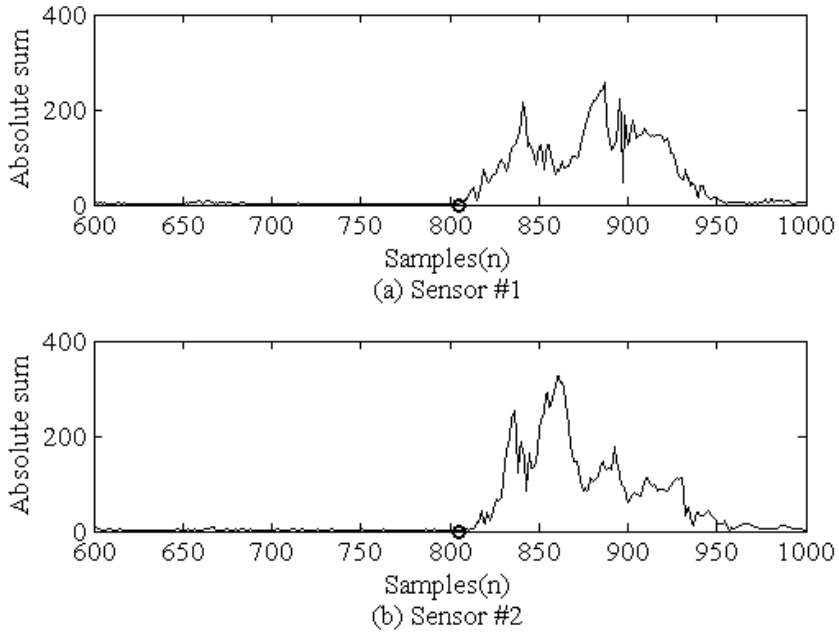


Figure 6.4 Absolute sum of X, Y, Z-axis angular rate data in sensor #1 (a) and sensor #2 (b) after synchronization

6.1.2 Zero Error Compensation

As discussed in Chapter 5, all sensors were calibrated carefully before experiment. However, according to the data collected from the sensors, angular rate data collected before subject started to move is not zero. Instead, there is an offset which is zero error of the sensor. If this offset gets mixed into the real angular rate, the motion capture will get worse with time since the errors are accumulated. Therefore, this zero error, which is assumed constant during one ADL, should be compensated.

For each ADL, the subject would keep a starting posture before all sensors were well-connected and waited another amount of time. Therefore there is a time period from all the ten sensors were connected to the subject began to move. For each sensor, the zero error of angular rate during a specific ADL can be thought as the average value of this period of time since there should be no angular rate while subject was staying original posture. Then, the angular rate of the real body movement should be subtracted by the zero error calculated, i.e., the average value of staying original posture period. This process should be applied to each individual angular rate data since the zero error changes during different ADL experiment and varies from different sensors. Table 6.1 compares the offset of X, Y, Z-axis average angular rate data before and after applied zero error compensation in sensor #1. It is clear that in Table 6.1 the average offset is significantly reduced by applying this data pre-processing procedure.

Table 6.1 Comparison of offset before and after zero compensation of angular rate data in sensor #1

Sensor #1	X-axis angular rate (^{o}/sec)	Y-axis angular rate (^{o}/sec)	Z-axis angular rate (^{o}/sec)
Offset before zero error compensation	0.0320	0.2595	0.1110
Offset after zero error compensation	-0.0024	0.0240	-0.0005

6.2 Body Motion Capture Simulation Results

After data synchronization and zero error compensation, the optimal rotation order algorithm discussed previously is applied to get subject's body motion during each ADL. To track subject's whole body movement, location and orientation information in 3-D space for each body segment need to be calculated first. For example, Figure 6.5 shows the location of left thigh during the second ADL (initially standing to sitting, as discussed in Chapter 5).

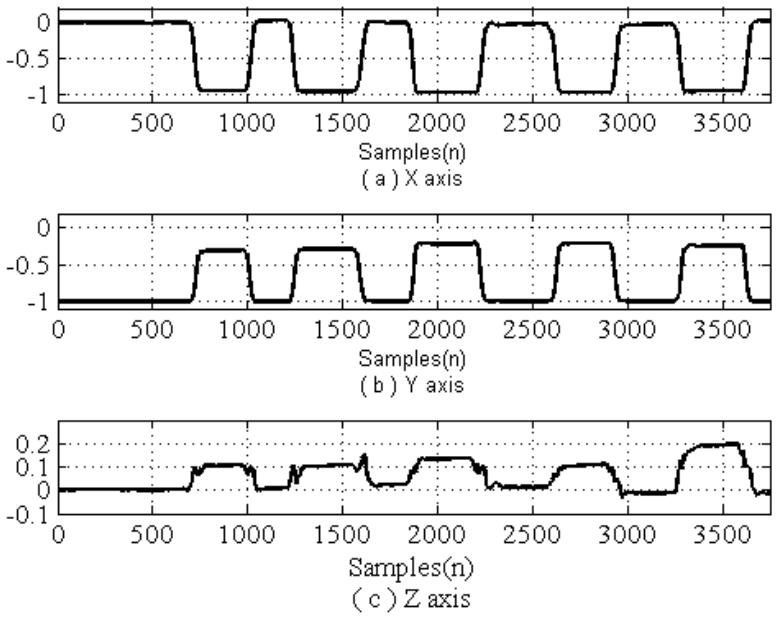
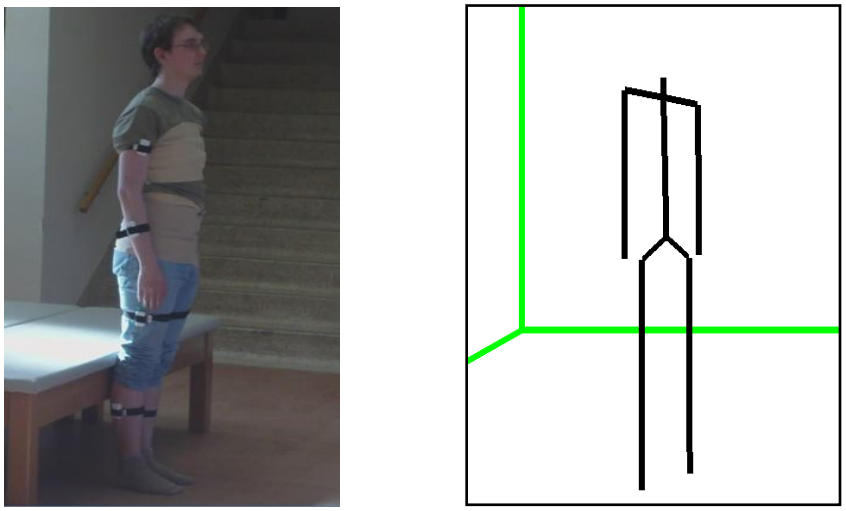


Figure 6.5 The location of left thigh in 3D space using optimal rotation order algorithm

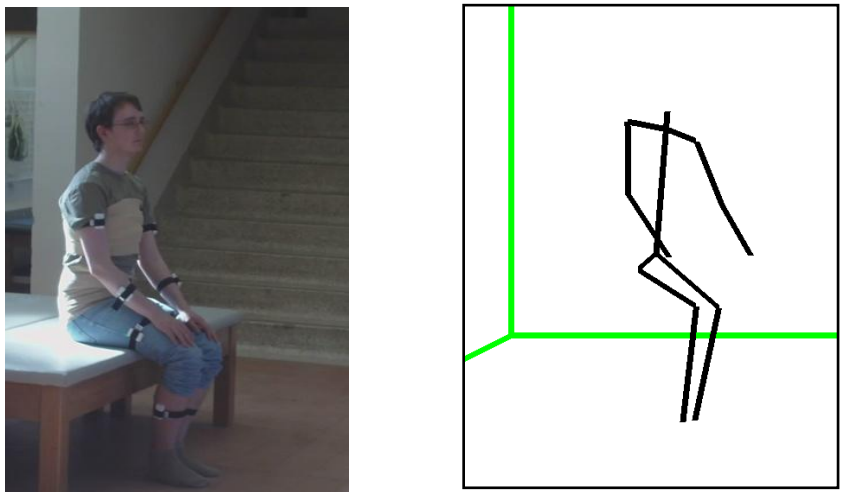
Here, to simplify the simulation, the length of the left thigh is scaled to 1 (unit length) and vector ${}^{e}V_0 = [X\ Y\ Z]^T = [0\ -1\ 0]^T$ is used to represent the original location (standing up) of the left thigh. Also, the left hipbone is assumed to be at a fixed point $[X\ Y\ Z]^T = [0\quad 0\quad 0]^T$. Therefore, ${}^{e}V_{\mathrm{i}} = [X\ Y\ Z]^T = [0\ -1\ 0]^T$ means subject is standing up (for example, sample 500) and subject is sitting down when ${}^{e}V_{\mathrm{i}} = [X\ Y\ Z]^T \approx [-1\ 0\ 0]^T$ (for example, sample 1500). Once location and orientation data of

one body segment are obtained, it can be easily to extend to other body segments and to all ADLs.

After all of the body segment locations are calculated, a simple avatar model is generated using OpenGL language, which can be viewed on a computer screen. This model shows the dynamic motion of the subject's body in each ADL and reproduces the subject's motion similar to experiment video. The virtual dimensions of the avatar model are proportional to the physical dimensions of the actual test subject which were measured before experiment.



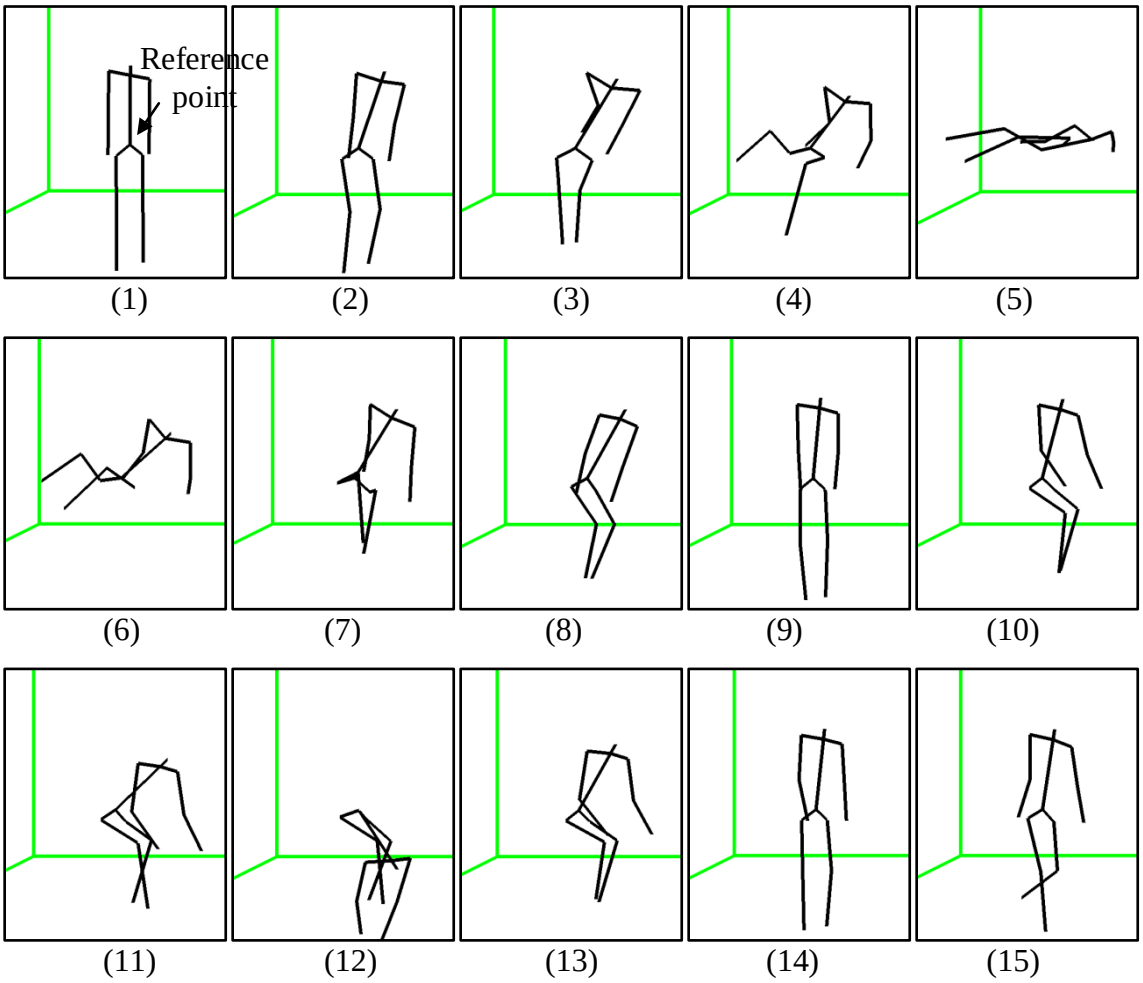
(a) standing up posture



(b) Sitting down posture

Figure 6.6 Comparison of video snapshot and avatar model for ADL No. 2

In Figure 6.6 (a) and (b), the avatar model of the standing up and sitting down posture are compared to the video snapshot. In the figure, the motion capture for the second ADL is successfully achieved. Actually, for all of the ADLs in the experiment, the avatar model using optimal rotation order algorithm can successfully track the subject's motion. In order to show the result of all ADLs tracking, the ADL sequence No. 10 discussed in Chapter 5 is used here in Figure 6.7 since this sequence consists of the most ADLs tested in experiment. Specifically, in Figure 6.7, subject #1 is performing the following motion sequence: initially stand still (1), then lie down (2-5), stand up (6-9), sit down and tie shoes (10-12), stand up, walk forward and backward (13-16), bend to pick up an object (ball) from the floor (17-18), reach forward (19-20), reach upward (21-23) and finally step onto and down from a stool (24-30). Here, since the absolute displacement of the subject is not important in this research, the reference point is set to the intersection point around hip and fixed in coordinates, as shown in Figure 6.7 (1).



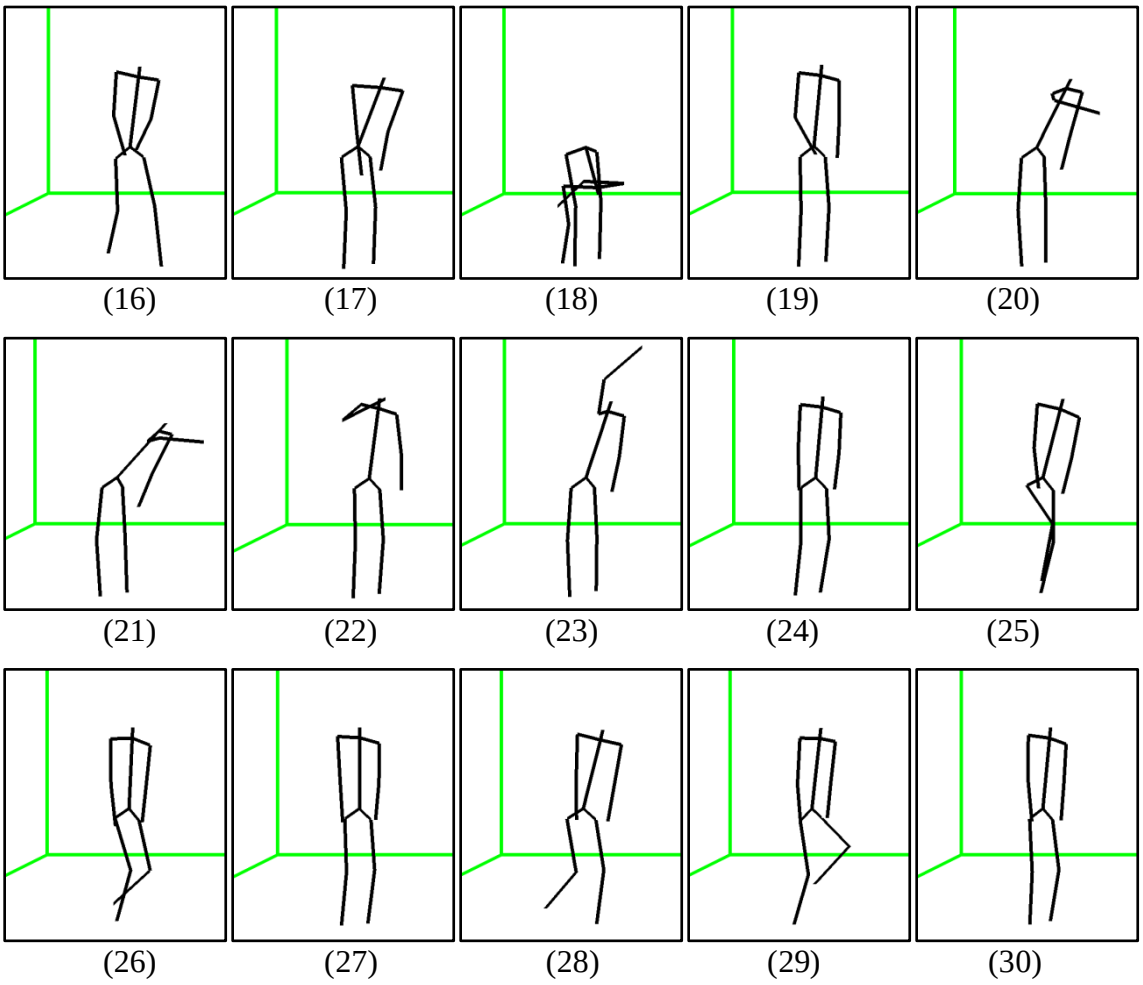


Figure 6.7 Computer model for capturing ADL No. 10 of subject #1

According to the motion capture results of this ADL sequence, the Euler angle-based optimal rotation order algorithm using acceleration and angular rate can track the common daily life activities continuously and accurately. By mounting ten sensors on each body segments, subject's movement can be well captured on computer using data collected from the sensors. This provides great potential not only on medical care applications, also on filming, virtually reality, etc.

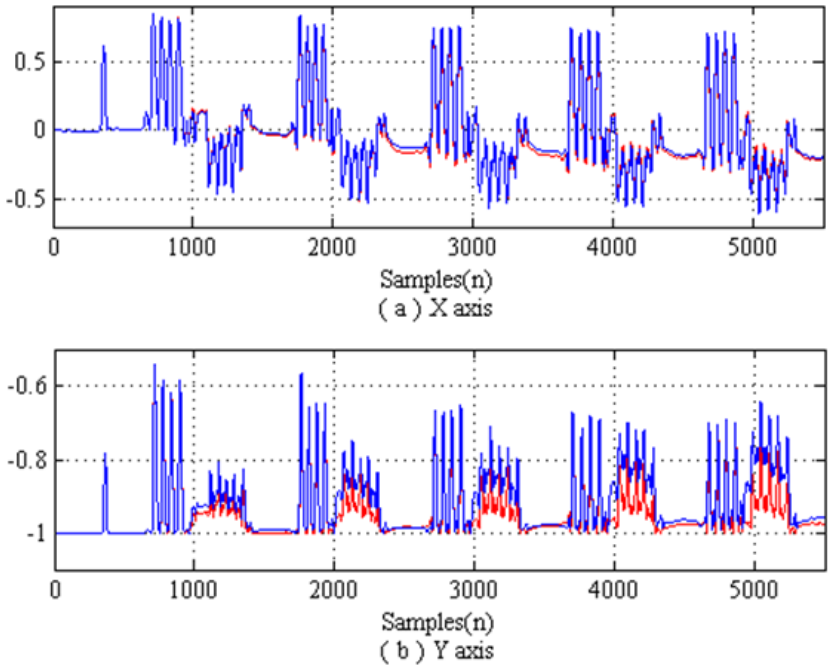
6.3 Motion Capture Results Discussion

During the motion capture data processing, there are two things worth discussing. The first is the comparison between single rotation order and optimal rotation order algorithms. The second is a phenomenon which can be seen in avatar model when the

subject turns 180° around, which is happened in ADL No. 4 (walking forward, turning around and walking backward) and ADL No. 9 (climbing up the stairs, turning around and climbing down the stairs).

6.3.1 Comparison Between Single and Optimal Rotation Order Algorithms

For single rotation order algorithm, the rotation order must be fixed to one of the six rotation orders: X-Y-Z, Y-X-Z, Z-X-Y, X-Z-Y, X-Z-Y or X-Z-Y. While in optimal rotation order algorithm, the rotation order is not fixed. It is determined by the comparison of estimated and measured earth gravity direction. Therefore, optimal rotation order algorithm can correct the error induced by rotation order at each step and avoid error accumulation which happens in single rotation order algorithm. For example, Figure 6.8 shows the comparison of left thigh location in ADL No. 9 (climbing up and down steps) obtained using single and optimal rotation order algorithms. The red lines are the location information obtained using optimal rotation order while the blue lines are that using single rotation order (i.e., Z-X-Y).



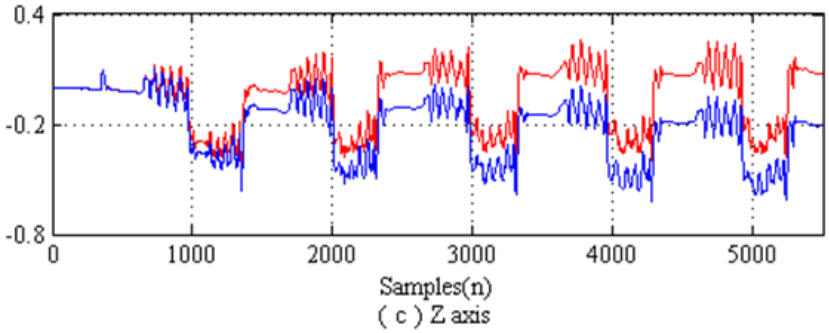
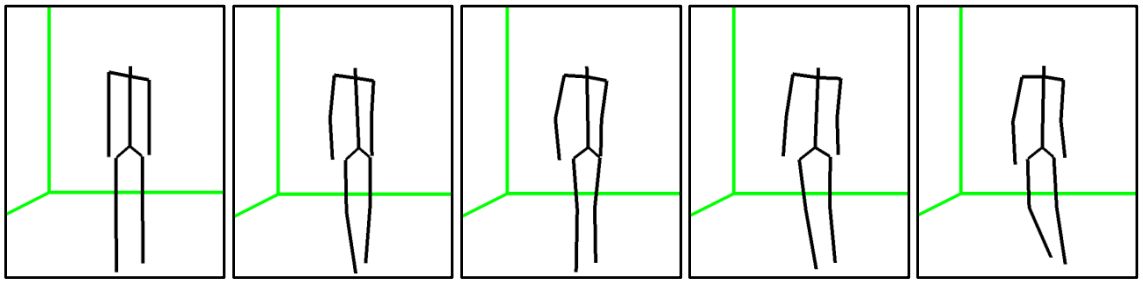


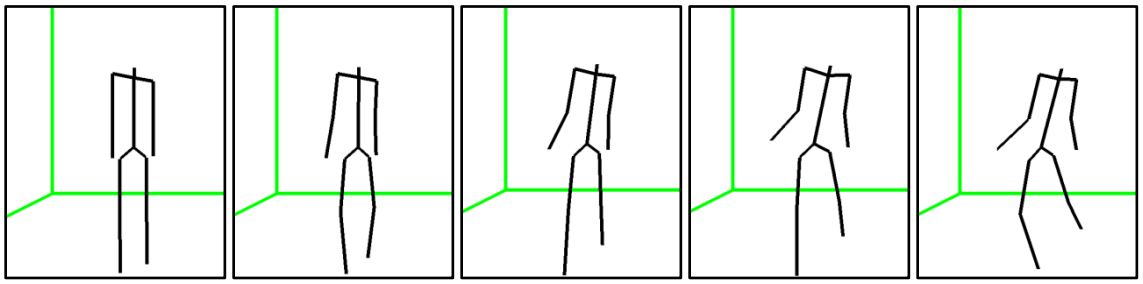
Figure 6.8 Left thigh locations using single (Z-X-Y) and optimal rotation order algorithms

In Figure 6.8, for single rotation order algorithm, the Z-axis data of the left thigh location vector begin to depart from the actual value when the subject starts to move and the error gets accumulated with experiment time getting longer. This error can be also easily reflected in the avatar model movement, as shown in Figure 6.9.



(1) (2) (3) (4) (5)

(a) Motion capture results using optimal rotation order algorithm



(6) (7) (8) (9) (10)

(b) Motion capture results using single rotation order algorithm

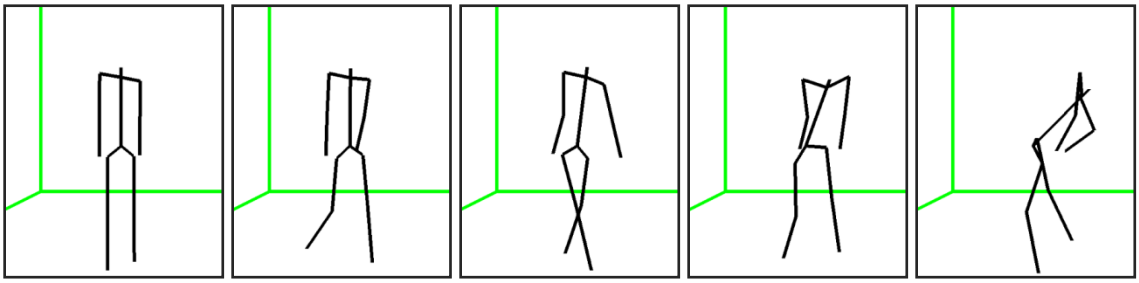
Figure 6.9 Comparison of motion capture results of ADL No. 9 using signal and optimal rotation order algorithms

Here, ADL No. 9 climbing up and down stairs is tested. The five figures in Figure 6.9 (a) and (b) corresponds respectively to the standing postures before each time the subject

climbs up stairs. Obviously, the avatar model calculated using optimal rotation order algorithm can return to its original position well, while that using single rotation order could not go back to original position due to the error accumulated. Compared to single rotation algorithm, although calculating six rotation matrices in each step will increase the algorithm complexity, optimal rotation order algorithm has better capture results and higher accuracy. In the thesis, since the accuracy is a critical factor which will directly affect the motion recognition by doctors and further affect the near-fall detection results, optimal rotation order algorithm is used in FANFARE project for daily ADLs capture. Single rotation order algorithm can be used in other applications in which high accuracy is not necessary, but high efficiency and low algorithm complexity are their concerns.

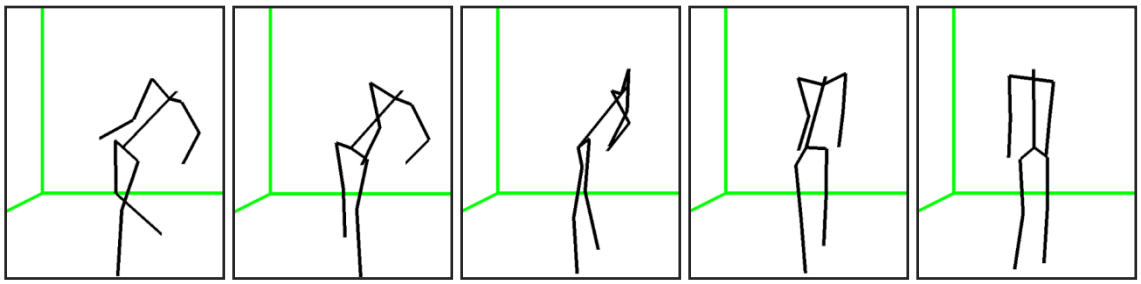
6.3.2 Motion Distortion during Turning 180° Around

For the motion capture results of all the ADLs, all daily life activities can be tracked in reasonable accuracy but except one, i.e., when subject is turning around of 180° in ADL No. 4 and ADL No. 9 as shown in Figure 6.10 and Figure 6.11. Figure 6.10 (a) shows the sequences of a model walking forward in ADL No. 4 and the motion is well captured. However, as shown in Figure 6.10 (b), after the model turned around of 180°, the walking back model is distorted. Specifically, the subject's back bends backward a lot while all the other body segments look normal. The same thing happened in Figure 6.11 in which a model is climbing up and down stairs.



(1) (2) (3) (4) (5)

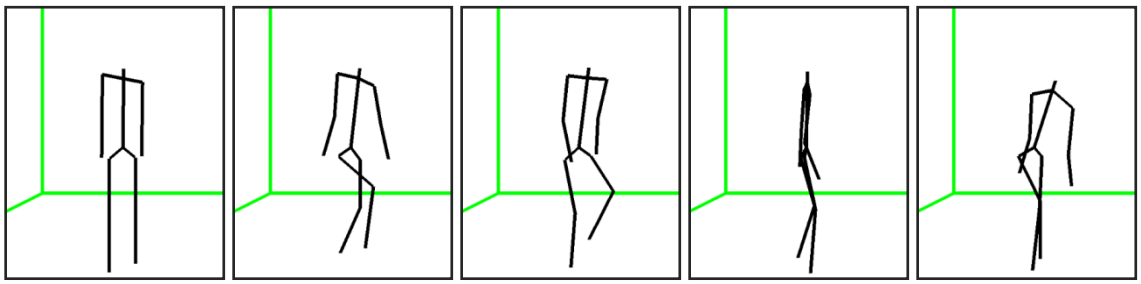
(a) Walking forward



(6) (7) (8) (9) (10)

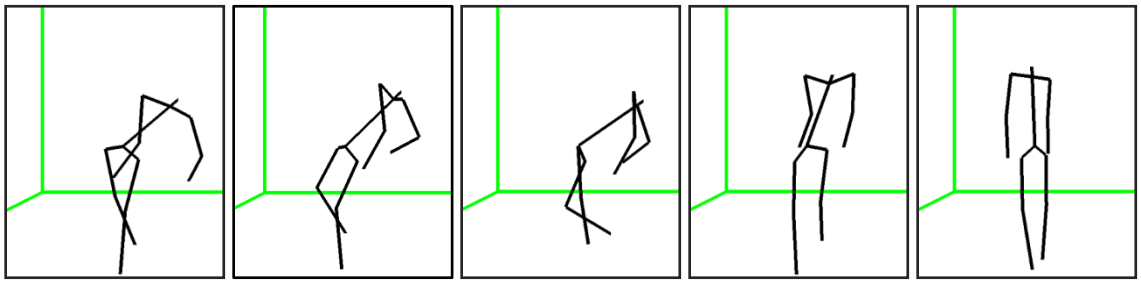
(b) Walking backward

Figure 6.10 ADL No. 4 walking forward and return



(1) (2) (3) (4) (5)

(a) Climbing up stairs



(6) (7) (8) (9) (10)

(b) Climbing down stairs

Figure 6.11 ADL No. 9 climbing up and down stairs

The reason of this phenomenon is that when two sensors were attached to the subject's back, the line through them is not actually parallel with the subject's back. Instead, there is an angle, as shown in Figure 6.12.

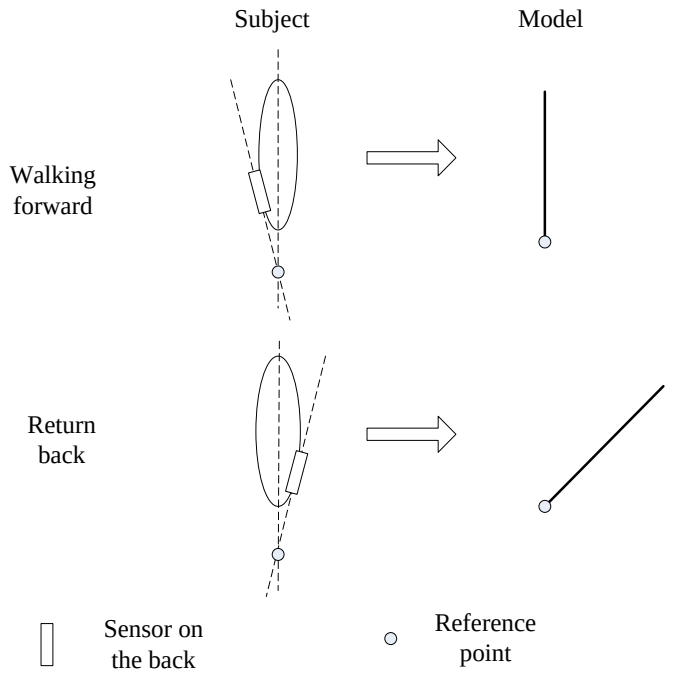


Figure 6.12 Walking forward and return

Therefore, assuming that the subject's back is parallel to Z-axis (Z-axis is pointing up to space), then the model's back will bend backward when the subject turns around 180^o and walks back, as shown in Figure 6.12.

6.4 Comparison to Previous Work

Table 6.2 shows the comparison between this system and some previous inertial motion capture systems. The tactile motion suit is presented by Y. Fujimori et al. in 2009 [52], the purpose of which is to combine motion information and contact information to achieve high accuracy whole body motion capture. The accelerometer tracking system is developed by L. Dong et al. [53] with ambition to track all the daily body activities at good accuracy. In [54], S. Y. Sun et al. uses the micro SMU (Sensor Measurement Unit) for lower body motion capture. An upper body motion capture using Ultra-miniaturized Inertial Measurement Unit (IMU) is proposed in [55] by Z. Lin et al..

One challenge in Table 6.2 is how to compare the accuracy of different systems. Existing classification-based fall detection systems usually detect falls by accounting the

numbers of detected correct falls and false falls in experiment, based on which a percentage of accuracy can be easily calculated. However, it is very difficult to obtain an exact accuracy based on motions performed by the computer generated avatar and real subjects. Firstly, the computer created avatar is usually much simpler than a real human body, so they have different body shapes. Secondly, the physical constrains of a human body and deformations during the motion are not considered when generating the avatar. Finally, there is not a standard method which can measure the differences of locations and orientations between the avatar and the subject. Nowadays, some researches use inertial sensor system based computer avatar compare to optical and image sensor based avatar to obtain a more intuitive comparison, since the later can achieve a very high accuracy and has been verified in various applications, such as computer animation, pattern recognition. However, this thesis is mainly focused on the inertial sensor motion capture system. If implementing the optical sensor based avatar in this research, the whole work will be doubled. In the future, the optical sensors could be included in the FANFARE project for further comparison and near-fall detection. Here, three ranks are defined for roughly comparing the accuracy according to the practical use in clinical point of view:

(1) Good. Generating a 3D computer avatar, capturing the whole body movements without significant distortion, and all the captured movements can be clearly recognized by eyes.

(2) Medium. Generating a 3D computer avatar, capturing partial body movements without significant distortion, and all the captured movements can be clearly recognized by eyes while some segments information are lost.

(3) Relatively low. Generating a 2D computer avatar, capturing the whole body or partial body movements without signification distortion, and all the captured movements can be recognized by eyes.

In Table 6.2, all the previous systems except accelerometer tracking system are using accelerometers, gyroscopes and magnetometers for motion data collection. Compared to the SHIMMER sensor system which only uses accelerometers and gyroscopes, adding magnetometers to system as in tactile motion suit will increase the cost as well as the complexity in hardware design and algorithm computation. Furthermore, magnetometers

are very sensitive to the disturbance from other magnetic fields in the environment. The tactile motion system requires subjects to wear the suit during the motion, which make the subjects uncomfortable and restrict their movements. While in the thesis, ten SHIMMER sensors are attached on the body segments by separate straps which are flexible and light and allow free motions without any restrictions. For the accelerometer tracking system, although it has lower complexity in system design than the SHIMMER system, using 2-axis accelerometer to capture gravity related vectors will lose one DOF (Degree Of Freedom). Meanwhile, without compensation from angular information, it suffers from drifting problems and the accuracy is relatively low.

Most previous systems utilize wired communications and data transmission between sensors and sensors or sensors and computers, such as tactile motion suit, SMU and IMU. This strategy restricts the motion in a relatively small area and wearing wires on the body can easily obstruct subjects' movements. SHIMMER sensors uses wireless communications and there is a SD card on each sensor node, which can continuously record motion data and support any motion capture without distance restriction.

In the experiment, SHIMMER sensors captured the whole body movement of three subjects with ten types of ADLs which covers all the basic daily activities of an elderly person. While, tactile motion suit only tested stretching and lying down activities, SMU focused on lower body motion such as bending and extending thigh and calf and IMU worked on upper body movements such as lifting arms.

According to the above systems, all of the 3D avatars can capture the specific motions with good accuracy. The tactile motion suit combines contact information with motion information, which can track whole body movements with high accuracy. However, the cost and complexity in both hardware and software are relative high. Although SMU and IMU systems are accurate as well, they focus on only partial body movements instead of whole body motions. Also, using magnetometers will bring disturbance problems. The accelerometer tracking system has lowest complexity and cost compared to other systems, while the accuracy is relative low as well. The SHMMER sensor system has advantages of low cost and low complexity in hardware design, system implementation and computation. Its wireless feature supports outdoor or everyday movement without distance restrictions. It is comfortable and flexible, which shows high

potential in many applications, especially suitable for daily motion capture and near-fall detection.

Table 6.2 Comparison with previous work

Metrics \ System	SHIMMER sensor system in this thesis	Tactile motion suit [52]	Accelerometer tracking system in [53]	Micro Sensor Measurement Unit (SMU) [54]	Ultra-miniaturized Inertial Measurement Unit (IMU) [55]
Sensors	3-axis accelerometer 3-axis gyroscope	3-axis accelerometer 3-axis gyroscope 3-axis magnetometer Tactile sensors	2-axis accelerometers	3-axis accelerometer 3-axis gyroscope 3-axis magnetometer	3-axis accelerometer 3-axis gyroscope 3-axis magnetometer
Size of sensor node (mm)	$53 \times 32 \times 25$	Motion sensor:$43 \times 35 \times 18$ Tactile sensor:120×180	-	-	$27 \times 23 \times 11$
Sensor locations	Upper and lower arm(4), Upper and lower leg(4), Chest(1), Hip(1)	Upper and lower arm(4) Upper and lower leg(4) Chest(1), Hip(1) Back, Front, Shoes	Upper arm(2), Upper and lower leg(4), Trunk(1)	Upper and lower leg(4)	Upper and lower arm(4), Hand(2), Head(1), Neck(1), Shoulder(2), Back(2)
System components	One central computer Ten SHIMMER sensors	One central computer Ten motion sensors Several tactile sensors	Seven motion sensors	One computer, Four SMU	Twelve WB-3 IMU
Power supply	Onboard power unit	Extra power unit	-	Extra power supply	Extra power supply
Communication	Wireless	Wired	Wireless	Wired	Wired
Hardware complexity	Medium	High	Low	Medium	Medium
Algorithm complexity	Low: 3×3 matrix multiplication	High: 4×4 matrix multiplication	High: 4×4 matrix multiplication	High: 4×4 matrix multiplication	High: 4×4 matrix multiplication
Tested motions	Sit, stand, walk, lay down, pick up, reach forward and upward, tie shoes, climb stairs.	Stretching after sitting, Laying down	A series of activities: Standing, swing legs, sitting, lying, leaning body	Adduct, bend and extend thigh; bend and extend both thigh and shank.	Lifting up the right arm or left arm; Lifting both arms
Capture results	Accurately captures all the motions	Accurately captures all the motions	Captures most of the motions, but lost the lower arm movements	Accurately capture all the lower body motions	Accurately captures all the upper body motions
Avatar	3D whole body model	3D whole body model	2D whole body model	3D whole body model	3D upper body model
Cost	Low	High	Very low	Medium	Medium
Accuracy	Good	Good	Relatively low	Medium	Medium

7 Conclusions and Future Work

7.1 Summary and Conclusions

Due to the similarities between near-falls and people's activities of daily life, it is critical to accurately capture people's body movements to help doctors study features of normal activities and further detect near-falls. In this thesis, a full human body motion capture system using wearable inertial sensors, namely accelerometers and gyroscopes, was proposed based on the commercial SHIMMER platform and tested in lab environment.

Before the ADL experiment, ten SHIMMER sensors were well calibrated to get the useful data with units since the raw data collected by sensors are just numbers without any units. Then, an overnight drifting test of all sensors was implemented to test the data drifting in each sensor node. In ADL experiment, the length of each body segment is firstly measured. Based on the measurements, all sensor locations on the human body are carefully selected under the instructions of doctors to make subjects feel comfortable and minimize the deformation affection during movements. After that, nine pre-defined typical ADLs and one ADL sequence were performed by three different subjects in the lab environment. These selected ADLs almost covered all the basic activities during people's daily life. In data processing, both single rotation order algorithm and optimal rotation order algorithm are applied to convert the inertial data collected by sensors to orientation and location information. Finally, a computer avatar created by OpenGL language shows all the captured motions, which can help doctors intuitively studying and analyzing those ADLs.

According to the results of overnight drifting, the accelerometers and gyroscopes can give reasonably high accuracy and no significant data drifting when collecting acceleration and angular data. Also, the carefully selected mounting scheme of sensors on

human body in this project manifests itself as an efficient and optimal arrangement since it can get the whole body movement coverage and ensure the least sensor number. The comparison between single rotation algorithm and optimal rotation algorithm shows that although both of them can capture all the subjects' movements, the later shows higher accuracy. For the single rotation order algorithm, the avatar movement departs from the actual subjects' movement gradually with time. This is because that the rotation error will get accumulated with time due to the fixed matrix rotation order. While, the optimal rotation order algorithm can mitigate the error by using the optimal rotation matrix at each step and consequently make the capture results irrelevant to time. Although optimal rotation order algorithm need more computation in each step, the high accuracy in capture results makes it more suitable for capturing normal activities, since the subjects' movement need to be monitored all the time or during a long period of time. Finally, the animation shows that the created avatar can continuously capture all the subjects' movements accurately and the accuracy is good enough for helping doctors study those activity features based on the avatar.

The comparison with some previous inertial motion capture system shown in Table 6.2 illustrates the advantages of using SHIMMER system for whole body motion capture. Compared to other inertial sensor motion capture systems, this system only applies accelerometers and gyroscopes instead of the combination of magnetometers and the said sensors, which reduces the complexity in both hardware and algorithm computation while still achieves high accuracy. The wireless communications and onboard SD card extend the application for everyday motion capture without restrictions in ADLs and locations in a dwelling of the patients. Also, the straps are flexible and light, which ensures free movements without motion restrictions. In the experiment, ten ADLs are tested by three subjects. Those ADLs are selected to cover all the basic activities in the elderly people's daily movements. However, most previous systems only pick some of the ADLs. According to experiment results, the SHIMMER system can accurately capture all the ten ADLs continuously without significant drifting. This inertial sensor motion capture system shows its great potential in not only medical research for near-fall detection, also other applications as well, such as computer animation, video gaming and virtual reality.

7.2 Future Work

For this body motion capture project, although the subjects' movements could be well captured, there are still some existing issues needed to be solved and improvements can be made in the future work for near-falls detection application.

7.2.1 Existing Issues to be Solved for Current System

The main issues to be solved for the current system include the followings:

1. The problem of reproducing turning-around movement should be solved.

As discussed in Section 6.3.2, after the subject turns around of 180°, the walking back model is distorted. Since it is clear that this problem comes from the un-parallel human back to the line through the sensors attached on the back, the angle between them can be measured first in experiment to avoid this error. It then can be used to correct the difference between walking forward and back.

2. Data can be transferred to central host computer via wireless connection instead of being stored in SD card.

For the current system, all of the data collected by sensors are stored in SD card first and read out for processing after experiment. This scheme cannot achieve real-time computer model displaying. Therefore, the wireless network will be built to allowing real-time data collecting and body movement reproducing.

7.2.2 Improvements for Future Applications

Besides the above two issues existing in the current system, there are other improvements can be made to achieve better motion capture performance.

1. Actual daily life movements instead of pre-defined ADLs will be monitored

In this project, nine pre-determined ADLs and one fixed ADL sequence were tested in the experiment. In future, the experiment will be extended to the actual subjects' daily life movements, which means the subject will not be constrained to move only as the pre-defined movements. If the system is able to track every movement of people's daily life in a relatively long period of time, it will be ready to be applied to near-falls detection applications.

2. **Elderly people can be included as the test subjects.**

Instead of only young subjects involved in the experiment, elderly people will take part in the project. The difference between the movements of young and elderly people can then be studied and corresponding adjustment of the system hardware or motion capture algorithms might be applied.

3. **Sensor number and the sensor size could be reduced.**

Although SHIMMER sensors are small and light, wearing ten SHIMMER nodes on the body, especially for the elderly people, is not convenient. Meanwhile, if the sensor is not attached on the selected location or if the sensor is not tightly fixed on the body during the movement, the accuracy of captured results will be reduced. Therefore, more experiments can be implemented to investigate the relationship of sensor numbers and the accuracy to reduce the sensor numbers while still maintaining the accuracy. Also, more technologies can be involved to reduce the size of sensor nodes or even embedding sensors into clothes instead of using straps.

4. **Standard evaluation system can be built to estimate the accuracy of this system**

Unlike most existing fall detection systems which can directly show the percentage of accuracy based on the classification results, motion capture systems are difficult to calculate the percentage of accuracy according to the motions of avatar. Firstly, the created shape of avatar is usually simpler than real human body. Secondly, there is not a standard method to measure the orientation or location differences between the human body and avatar. Nowadays, the computer animation based on optical sensor or camera recorded information can capture even subtle motions on a face with very high accuracy. In the future, the camera and optical sensor based avatar can be generated to compare with inertial motion capture system created avatar for further comparison. Since they have the same shape and can be put into same 3D space, the comparison between them will intuitively show the differences and help calculate the accuracy.

5. **Combining other sensors to implement near-fall detection**

Inertial motion capture systems can capture all the movements during people's daily life. However, using only inertial sensors can hardly tell the difference between near-falls and other normal activities. In the future, other technologies should be taken into account,

such as pressure sensors. Since near-falls involve in losing balance and getting balance movements, detecting the location of body mass center will helpful for detection balance. When a person is standing, he/she only uses two feet to support the whole body and keep balance. Therefore, the body mass center should be within the area between two feet. When the mass center goes out of this area, this movement can be considered as losing balance. If the body mass center goes back into the feet area again, this motion can be corresponded to the movements of getting balance back. Pressure sensors can detect the pressure under the feet and calculate the body mass center based on collected pressures. Combined with pressure sensors and inertial motion capture system, further experiments can be implemented to detect near-falls in a lab environment.

References

[1] N. Noury, "A Smart Sensor for the Remote Follow Up of Activity and Fall Detection of the Elderly," *2nd Annu. Int. IEEE-EMB Special Topic Conf. Microtechnologies in Medicine & Biology*, 2002, pp. 314-317.

[2] S. Luo and Q. Hu, "A Dynamic Motion Pattern Analysis Approach to Fall Detection," *IEEE Int. Workshop on Biomedical Circuits and Systems*, 2004, pp. 1 - 5-8a.

[3] S. M. Zhang, P. McCullagh, C. Nugent and H. R. Zheng, "A Theoretic Algorithm for Fall and Motionless Detection," *3rd IEEE Int. Conf. Pervasive Computing Technologies for Healthcare*, 2009, pp. 1-6.

[4] C. Dinh and M. Struck "A New Real-Time Fall Detection Approach Using Fuzzy Logic and a Neural Network," *6th Int. Workshop on Wearable Micro and Nano Technologies for Personalized Health (pHealth)*, 2009, pp. 57-60

[5] M. Yu, S. M. Naqvi, A. Rhuma and J. Chambers, "Fall Detection in a Smart Room by Using a Fuzzy One Class Support Vector Machine and Imperfect Taining Data," *IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 1833-1836.

[6] D. K. Arvind and M. M. Bartosik, "Speckled Robotics: Mobile Unobtrusive Human-Robot Interaction Using On-Body Sensor-based Wireless Motion Capture," *18th IEEE Int. Symp. Robot and Human Interactive Communication*, 2009, pp. 421.

[7] H. I. Lin and C. L. Chen, "A Hybrid Control Policy of Robot Arm Motion for Assistive Robots," in *Proc. IEEE Int. Conf. Information and Automation*, 2011, pp. 163-168.

[8] Y. D. Ma, Z. Wu, Z. Q. Zhang and J. K. Wu, "Web3D Technologies and Motion Estimation in Education," *Education Technology and Training 2008 and 2008 Int. Workshop on Geoscience and Remote Sensing*, 2008, pp. 69-72.

[9] O. Mirabella, A. Raucea, F. Fisichella and L. Gentile, "A Motion Capture System for Sport Training and Rehabilitation," *4th IEEE Int. Conf. Human System Interactions (HSI)*, 2011, pp. 52-59.

[10] K. Hachimura, H. Kato and H. Tamura, "A Prototype Dance Training Support System with Motion Capture and Mixed Reality Technologies," in *Proc. IEEE Int. Workshop on Robot and Human Interactive Communication*, 2004, pp. 217-222.

[11] M. Z. Patoli, M. Gkion, P. Newbury and M. White, "Real Time Online Motion Capture for Entertainment Applications," *IEEE Int. Conf. Digital Game and Intelligent Toy Enhanced Learning*, 2010, pp. 139-145.

[12] L. Kovavisaruch, J. Wisanmongkol, T. Sanpachuda, A. Chaiwongyen, S. Wisadsud, T. Wongsatho, B.Tangkamcharoen, B. Nagarachinda and C. Khiawchaum, "Conserving and Promoting Thai Sword Dancing Traditions with Motion Capture and the Nintendo Wii," in *Proc. Technology Management in the Energy Smart World (PICMET) IEEE*, 2011, pp. 1-5.

[13] W. S. Meador, T. J. Rogers, K. Oneal, E. Kurt and C. Cunningham, "Mixing Dance Realities: Collaborative Development of Live-Motion Capture in a Performing Arts Environment," *ACM Computers in Entertainment*, Vol. 2, No. 2, pp. 1-15, 2004.

[14] A. Girson, I. Electronics, "Measurand's ShapeWrap II Motion Capture System," Information Quarterly, Vol. 4, No. 2, pp. 67-68, 2005.

[15] J. Maycock, J. Steffen, R. Haschke and H. Ritter, "Robust Tracking of Human Hand Postures for Robot Teaching," *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2011, pp. 2947-2952.

[16] M. S. Salim, H. N. Lim and M. S. M. Salim, M.Y.Baharuddin "Motion Analysis of Arm Movement during Badminton Smash," *IEEE EMBS Conf. Biomedical Engineering & Sciences*, 2010, pp. 111-114.

[17] S. Das, C. Gleason, S. Shen, S. Goddard and L. C. Perez, "2-D Tracking Performance Evaluation Using the Cricket Location-Support System," *IEEE Int. Conf. Electro Information Technology*, 2005, pp. 1-6.

[18] E. R. Bachmann, "Inertial and Magnetic Tracking of Limb Segment Orientation for Inserting Humans into Synthetic Evironments," Ph.D. Dissertation, Dept. Computer Science, Naval Postgraduate School, Monterey, CA, 2000.

[19] Y. Watanabe, T. Hatanaka, T. Komuro, M. Ishikawa, "Practical Motion Capture in Everyday Surroundings," *IEEE Workshop on Applications of Computer Vision (WACV)*, 2011, pp. 276-281.

[20] Z. Zhang, Z. Wu, J. Chen and J. Wu, "Ubiquitous Human Body Motion Capture Using Micro-Sensors," *IEEE Int. Conf. Pervasive Computing and Communications*, 2009, pp. 1-5.

[21] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell and B. G. Celler, "Implementation of a Real-Time Human Movement Classifier Using a Triaxial Accelerometer for Ambulatory Monitoring," *IEEE Trans. Information Technology in Biomedicine*, vol. 10, pp. 156-167, 2006.

[22] Q. Li, J. A. Stankovic, M. A. Hanson, A. T. Barth, J. Lach and G. Zhou, "Accurate, Fast Fall Detection Using Gyroscopes and Accelerometer-Derived Posture Information," *6th IEEE Int. Workshop on Wearable and Implantable Body Sensor Networks*, 2009, pp. 138-143.

[23] W. Zou, Q. X. Du, K. Yuan and J. F. Li, "An Attitude Tracking Method Considering Acceleration Compensation and Based on UKF," in *Proc. 8th IEEE World Congr. Intelligent Control and Automation*, 2011, pp. 52-57.

[24] G. H. Tao, S. Y. Sun, S. Huang, Z. P. Huang and J. K. Wu, "Human Modeling and Real-Time Motion Reconstruction for Micro-Sensor Motion Capture," *IEEE Int. Conf. Virtual Environments Human-Computer Interfaces and Measurement Systems (VECIMS)*, 2011, pp. 1-5.

[25] Y. J. Jung, D. H. Kang and J. W. Kim, "Upper Body Motion Tracking With Inertial Sensors," in *Proc. IEEE Int. Conf. Robotics and Biomimetics*, 2010, pp. 1746-1757.

[26] G. Welch, E. Foxlin, "Motion Tracking: No Silver Bullet, but a Respectable Arsenal," *Computer Graphics and Applications, IEEE*, 2002, pp. 24-38.

[27] L. Havasi and H. M. Szabo, "A Motion Capture System for Sign Language Synthesis: Overview and Related Issues," *Int. Conf. Computer as a Tool*, 2005, pp. 445-448.

[28] J. Vince, *Mathematics for computer graphics*, Springer, London, 2006.

[29] H. Schaub, *Analytical mechanics of space systems*, Reston, VA: American Institute of Aeronautics and Astronautics, 2003.

[30] J. J. Craig, *Introduction to Robotics: Mechanics And Control*, Addison-Wesley, 1989.

[31] J. Miller, "An Introduction to Quaternions and their Applications to Rotations in Computer Graphics," 2006, pp. 1-10.

[32] J. B. Quipers, *Quaternion and Rotation Sequences*, Princeton University Press, 1998.

[33] E. R. Bachmann, I. Duman, U. Y. Usta, R. B. McGhee, X. P. Yun and M. J. Zyda, "Orientation Tracking for Humans and Robots Using Inertial Sensors," *IEEE Int. Symp. Computational Intelligence in Robotics and Automation*, 1999, pp. 187-194

[34] R. Zhu and Z. Y. Zhou, "A Real-Time Articulated Human Motion Tracking Using Tri-Axis Inertial/Magnetic Sensors Package," *IEEE Trans. Neural Systems and Rehabilitation Engineering*, pp. 295-302, 2004.

[35] X. P. Yun, M. Lizarraga, E. R. Bachmann and R. B. McGhee, "An Improved Quaternion-Based Kalman Filter for Real-Time Tracking of Rigid Body Orientation," *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2003, pp. 1074-1079.

[36] J. L. Marins, X. P. Yun, E. R. Bachmann, R. B. McGhee and M. J. Zyda, "An Extended Kalman Filter for Quaternion-Based Orientation Estimation Using MARG Sensors," *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2001, pp. 2003-2011.

[37] *SHIMMER - Sensing Health with Intelligence, Modularity, Mobility, and Experimental Reusability*, Hardware Guide, DHeG Cambridge, 2006.

[38] *SHIMMER User Manual*, Realtime Technologies Ltd., 2008-2010.

[39] *MSP430F15x, MSP430F16x, MSP430F161x, Mixed Signal Microcontroller*, Texas Instruments Incorporated, 2009.

[40] *±1.5g, ±6g Three Axis Low-g Micromachined Accelerometer*, Freescale Semiconductor Inc., 2008.

[41] *IDG-500 Dual-Axis Gyro Product Specification*, InvenSense Inc., 2010.

[42] *Gyro Specification Datasheet*, Realtime Technology Ltd., 2011.

[43] J. T. Adams, "An Introduction to IEEE STD 802.15.4," *IEEE Aerospace Conf.*, 2006, pp. 1-8.

[44] H. Labiod, H. Afifi and C. D. Santis, *Wi-Fi, Bluetooth, Zigbee and WiMAX*, Springer, 2007.

[45] S. Raman, "TinyOS – An Operating System for Tiny Embedded Networked Sensors," *Paper Presentation for Advanced Operating Systems Course*, 2002.

[46] D. Gay, P. Levis, R. V. Behren, M. Welsh, E. Brewer and D. Culler, "The nesC Language A Holistic Approach to Networked Embedded Systems," *Proc. Programming Language Design and Implementation*, 2003.

[47] D. Gay, P. Levis, D. Culler and E. Brewer, *NESC 1.2 Reference Manual*, 2005.

[48] *SHIMMER Accelerometer Calibration and Data Transformation*, Realtime Technologies Ltd., 2009.

[49] *MMA7260Q XYZ Three-Axis Low g Acceleration Sensor*, Freescale™, 2005.

[50] *SHIMMER 9DOF Calibration User Manual*, Realtime Technology Ltd., 2011.

[51] Human Body Model, http://www.mayavideotutorial.com/images/images/Human_body.png, accessed in July 2011.

[52] Y. Fujimori, Y. Ohmura, T. Harada and Y. Kuniyoshi, "Wearable Motion Capture Suit with Full-Body Tactile Sensors," *IEEE Int. Conf. Robotics and Automation*, 2009, pp. 3186-3193.

[53] L. Dong, J. K. Wu and X. Chen, "A Body Activity Tracking System Using Wearable Accelerometers," *IEEE Int. Conf. Multimedia and Expo*, 2007, pp. 1011-1014.

[54] S. Y. Sun, X. L. Meng, L. Y. Ji and J. K. Wu, "Adaptive Sensor Data Fusion in Motion Capture," *13th IEEE Conf. Information Fusion*, 2010, pp. 1-8.

[55] Z. Lin, M. Zecca, S. Sessa, L. Bartolomeo, H. Ishii, K. Itoh and A. Takanishi, "Development of an Ultra-miniaturized Inertial Measurement Unit WB-3 for Human Body Motion Tracking," *IEEE/SICE Int. Symp. System Integration*, 2010, pp. 414-419.