FINGER TRACKING FOR HUMAN COMPUTER INTERFACE USING MULTIPLE SENSOR DATA FUSION

by

ZhengYang (John) Wang

Submitted in partial fulfillment of the requirements
for the degree of Master of Applied Science

at

Dalhousie University
Halifax, Nova Scotia
April 2023

To all the readers,

I dedicate this work to you, with the hope that this finger tracking technology is just the beginning of a great journey towards a more advanced and inclusive world.

My vision for this technology goes beyond its practical applications. I believe that it has the potential to bridge the gap between humans and machines, allowing for a more seamless and intuitive communication. However, I also believe that it has the power to bring humans closer together, enabling new and creative ways of connecting with each other.

As we continue to advance and explore the possibilities of this technology, I urge you to join me in imagining the endless possibilities that lie ahead. Let us embrace the future with open minds and open hearts, and strive towards a world where technology and humanity are seamlessly intertwined.

Thank you for taking the time to read my work, and for sharing my vision for a better tomorrow.

Table of Contents

List of Tables . . . . . **v**

List of Figures . . . . . **vi**

Abstract . . . . . **viii**

Acknowledgements . . . . . **ix**

Chapter 1 Introduction **1**

1.1 Research Motivation 1

1.2 Applications Overview 3
1.2.1 Click Detection / Scroll Detection 4
1.2.2 Touch Detection / Slide Detection 5

1.3 Thesis Contributions 7

1.4 Thesis Organization 10

Chapter 2 Background Theory/Related Work **11**

2.1 Human Finger Kinematics and Dynamics 11

2.2 Finger Tracking 15

2.3 Inertial measurement unit (IMU) 17

2.4 Time-of-Flight (ToF) Sensor 19

2.5 Ring Device 20

Chapter 3 Proposed System Controller Design **22**

3.1 System Architecture 22
3.1.1 Rigid-Flex PCB 22
3.1.2 Arduino MEGA2560 24

3.2 Number of Sensors Required 27
3.2.1 BNO055 27
3.2.2 VCNL4040 29

Chapter 4 Methodology **31**
4.1 Coordinate Systems 31
4.2 Introduction to Quaternions 33
4.3 Sensor Fusion 34
4.4 Determination of the Fingertip Positions 35
Chapter 5 Experimental Setup **38**
5.1 Hardware Configuration 38
5.2 Sensor Calibration 38
5.3 MATLAB Simulation 41
Chapter 6 Experimental Results **45**
6.1 IMU Sensor Experimental Results 45
6.2 Proximity Sensor Experimental Results 46
Chapter 7 Conclusion **52**
Chapter 8 Future Discussion **53**
Bibliography **54**
Appendice A - Sample Matlab Code **62**
Appendice B - Author's Publication List **65**

List of Tables

Table 3.1 SPECIFICATION OF BOSCH BNO055 28
Table 3.2 SPECIFICATION OF EACH SENSORS IN BNO055 29

List of Figures

Figure 1.1 Multi-Touch technology as an new user interface 2

Figure 1.2 Ray casting interface for object manipulation 3

Figure 1.3 Myo armband 4

Figure 1.4 Basic use of mouse 5

Figure 1.5 Smart Ring Touch Detection 6

Figure 1.6 Continual Slide and Microroll Gestures 7

Figure 1.7 DataGlove 8

Figure 2.1 Anatomy of the Hand 11

Figure 2.2 Extensor assembly and diagram of extensor and flexor tendons of the finger 12

Figure 2.3 Finger extensor assembly schematically 13

Figure 2.4 IMU sensor glove 15

Figure 2.5 Optical hand motion tracking 16

Figure 2.6 Precise Electromagnetic Finger Tracking 17

Figure 2.7 IMU sensor breakout board offer 6 DOF 18

Figure 2.8 Operation principle of ToF system 19

Figure 2.9 Oura Ring Gen3 21

Figure 3.1 System Architecture Block Diagram 23

Figure 3.2 Addressing Differential Signaling In Flex Circuits 24

Figure 3.3 Flex-Rigid Prototype PCB Design 25

Figure 3.4 Arduino MEGA2560 26

Figure 3.5 MEGA2560 BNO055 Wire Digram 27

Figure 3.6 Flex-Rigid Prototype PCB Design 28

Figure 3.7 VCNL4040 IRED Profile 29

Figure 4.1 Coordinate Systems 31
Figure 4.2 BNO055 Inertial Coordinate System 32
Figure 4.3 Quat-90-X 33
Figure 4.4 IMU Proximity Sensor Fusion 35
Figure 4.5 Figner tip location relate to bones and joints 36

Figure 5.1 Hardware Configuration 39
Figure 5.2 Motion Sesnor Calibration Tool 40
Figure 5.3 BNO055 calibration 41
Figure 5.4 PLX DAQ v2 demo windows sample 42
Figure 5.5 IMU system calibration status 43
Figure 5.6 BNO055 magnetometer calibration 43

Figure 6.1 6DOF Animation 46
Figure 6.2 Accelerometer 47
Figure 6.3 Tilt-Compensated Accelerometer 47
Figure 6.4 Gyroscope 48
Figure 6.5 Linear Acceleration 48
Figure 6.6 Linear Velocity 49
Figure 6.7 High-pass Filtered Linear Velocity 49
Figure 6.8 Linear Position 50
Figure 6.9 High-pass Filtered Linear Position 50
Figure 6.10 P.I.P. Flexion Angle thetaAngle relate to VCNL4040 Proximity Data 51
Figure 6.11 D.I.P. - P.I.P. Flexion Interdependence 51

Abstract

As human-computer interface advances from two to three dimensions, new input devices are required to allow low-cost learning and increased human involvement in virtual reality contexts. In this study, we offer a small, finger-worn, wireless motion-tracking platform that can be reprogrammed for a number of functions to enhance mobile computing. First, it functions as a wireless mouse in the air without requiring a surface for a mouse or trackpad, making it appropriate for augmented-reality or virtual-reality systems. Second, it supports single-finger motion tracking in its entirety. A more complex version is able to mix information from more units on multiple fingers to create a greater variety of options. The prototype of our ring will have an ultra-compact wireless sensing platform with an on-board triaxial accelerometer, triaxial magnetometer, triaxial gyroscope, and a short-range wireless Bluetooth transmitter with a Time-of-Flight (TOF) sensor for finger flexion detection. Quantitative and qualitative evaluations of the accuracy and usefulness of the Inertial measurement unit (IMU) sensor breakout board have been conducted. The results demonstrate that the Inertial measurement unit (IMU) sensor finger tracking system is intuitive for mouse-like tasks.

Acknowledgements

I would like to take this opportunity to express my heartfelt gratitude to Dr. Jason Gu, my supervisor, for his constant guidance, support, and encouragement during my research work. His expertise, knowledge, and constructive feedback have been instrumental in shaping my research project and bringing it to fruition.

I would also like to thank the staff in the Electrical and Computer Engineering Department for their valuable assistance and cooperation throughout my research work. Their support and contributions have been pivotal in making my research successful.

Furthermore, I am grateful for the support provided by Dalhousie ideahub build program, which helped me to turn my idea into a reality. The program offered me valuable resources, mentorship, and guidance that were critical in the development and execution of my project.

I would also like to extend my gratitude to Dr. Derek Reilly for his expertise in the field of Human-Computer Interaction (HCI) and for his insightful feedback and recommendations that have contributed significantly to the success of my project.

Finally, I would like to thank my family and friends for their unwavering support and encouragement during my research. Their love and support have been a constant source of motivation throughout my journey.

Thank you all for your invaluable support.

Chapter 1

Introduction

For a long time, desktop and laptop computers were the most commonly used computing platforms, and the computer mouse and keyboard were the primary human-computer interaction devices based on the operating system of those platforms [15].

As the essential hand-held pointing device, the computer mouse has dominated the human-computer interface (HCI). Due to the unique features of modern operating systems, most dynamic trajectories are two-dimensional relative to a surface [45]. Hand movements are often projected onto a display as a pointer[56], making the mouse the ideal device for a computer's graphical user interface. While the mechanical and electrical design of computer input technology has improved significantly, the core characteristics remain unchanged for most computer system users [19].

New computing platforms require a closer connection between the user and their personal device. Similar to the evolution of the computer mouse, Augmented Reality (AR) / Mixed Reality (MR)/ Virtual Reality (AR) technologies require a new 3-dimensional human-computer interface (HCI) device [43]. After years of technological advancements and improved visual, auditory, and other sensory simulations, Virtual Reality (VR) is now capable of providing stunning fictitious worlds. Augmented Reality (AR), which superimposes virtual information in a real-world environment, bridges the gap between the digital and physical worlds [10]. Both technologies fundamentally alter how people engage with digital information.

1.1 Research Motivation

Traditional mouse and keyboard devices for the next human-computer interaction platform in Virtual Reality (VR) and Augmented Reality (AR) environments cannot detect motion because they are intended for a two-dimensional operation system. [24, 77] These conventional input approaches only capture Two-dimensional (2D) motions, but Three-dimensional (3D) interface device need to provide continuous,

accurate tracking of 3D finger and hand location [4]. The resurgence of demand for input mechanisms detached from the Two-dimensional (2D) display. The Three-dimensional (3D) input device must be continuously accessible and unobtrusive in order to facilitate robust and expressive interaction in a variety of circumstances [70].

Figure 1.1: Apple Inc. demonstrate Multi-Touch technology as an new user interface, 2007, Macworld Conference and Expo 2007

Modern operating systems have been primarily based on the computer monitor as an information output device. The computer monitor displays information in pictorial or textual form. That makes the mouse the most representative interface device in 2D space. Users' movement is directly reflected to the operating system from the 2D constraining surfaces, such as the physical surface of the user's deck. That helps the user to point out a specific location with x and y coordinate information. How even, when navigating in 3D space, the limits of the mouse's lack of depth information becomes obvious [58].

Ray-casting interface device [14] has be commonly used for Mixed Reality (MR) platfrom. Those ray-casting interface is a handhold device like laser pointer. One major limitation of laser pointer is accuracy decrease when object's distance increase[56].

Therefore, since the mouser and ray-casting interfaces [57] does not provide an

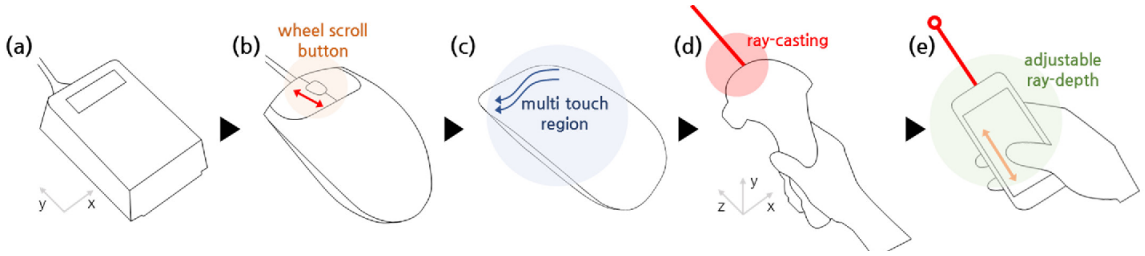

Figure 1.2: Ray casting interface for object manipulation [57]

intuitive user interaction in 3D environment. Thus, Finger motion tracking interface could overcome the limitations of 2D interface device and how to improve finger motion capture technology in order to improve human-computer interaction has been the subject of the research of numerous scientists [61]. When compared to other interactive devices, such as laser pointers, this intuitive engagement will be more readily accepted by users if it resembles natural, direct interaction with the hand. There are a number of prevalent options for finger dynamic capture, including optical motion capture [18, 65] plus computer vision, electromyography (EMG) detection [5, 34], inertial measurement unit (IMU), and electromagnetic induction. Optical detection is now the most common technology for gesture recognition, requiring typically simply an optical lens to capture the hand motion trajectory before machine learning recognizes the matching action. This method offers the advantages of inexpensive hardware costs and the capacity to concurrently distinguish several finger movements, but is limited by the variance of ambient light. The EMG detection technique involves recording and amplifying small muscle bioelectric impulses through the device, and simulating the muscle movements of the hand to produce a dynamic model using these signals. IMU technology can correctly record the trajectory of the device and is unaffected by the external environment, however it can only record a single object and has the issue of accelerometer drifting [76] issue when computing absolute location.

1.2 Applications Overview

Having developed a hardware platform capable of precisely monitoring finger motion enables sports activity analysis applications [30], including hand swing motion, and enables a user-friendly method of interacting with augmented reality and virtual reality apps. There are several methods for tracking finger motion. However, single-finger

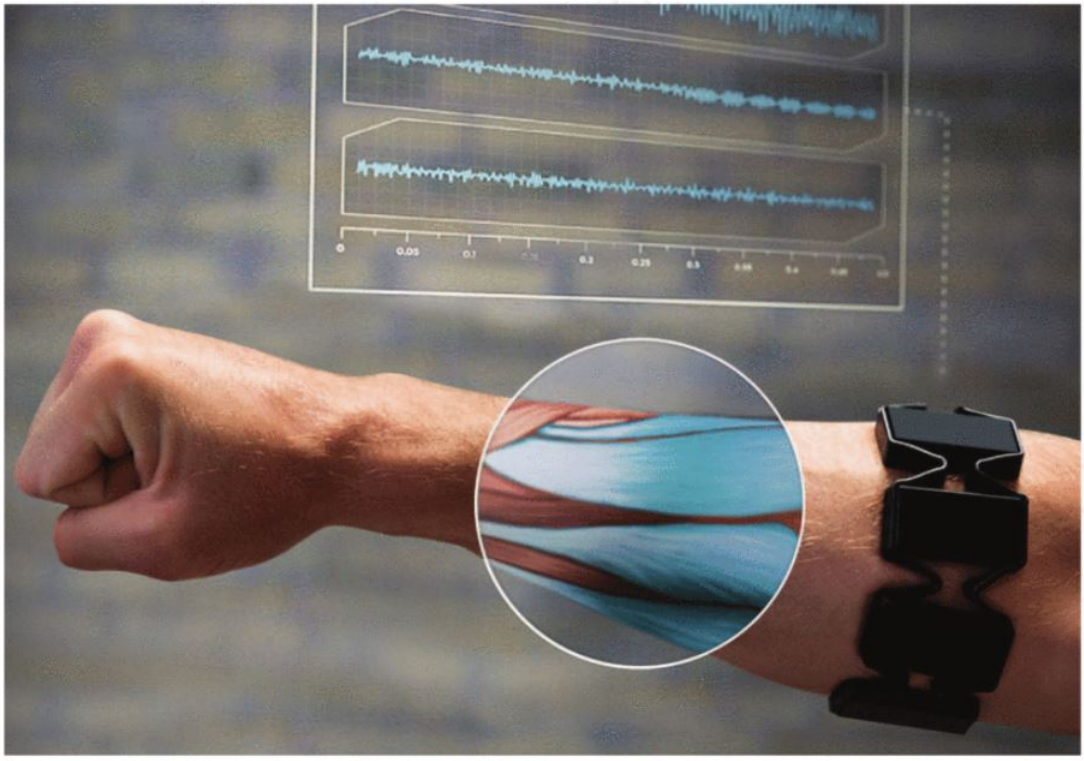

Figure 1.3: Myo armband [29]

integral motion tracking provides a simple, ubiquitous human-computer interface that functions as the user input device for a 3D operating system. This part will explain the potential applications our hardware platform can provide.

1.2.1 Click Detection / Scroll Detection

Commonly envisioned free-hand engagement approaches in AR and VR include taps and motions in the air. However, whether interacting with an app on your AR glasses or painting in a VR game, the feeling of touch is one of the most critical aspects of an immersive experience. The capability to detect taps reliably on environmental and body surfaces enables the development of new types of always-available ambient interfaces. Owing to the intuitive nature of humans and contemporary touchscreen technology, the fingers serve as the interface between the user and virtual objects. The precision of absolute fingertip location monitoring substantially impacts the user experience. The ability to reliably identify the fingertip location based on the finger joint flexion angle enables an immersive method of touching virtual objects with a

fingertip, similar to tapping application icons on a touchscreen. The finger joint flexion angle detection gives not only a fingertip position estimate but also a scrolling motion detection that may be utilized to simulate a scrolling wheel. These two interactions may be included in the single smart ring by using clicking and scrolling movements[24].

Figure 1.4: Basic use of mouse [24]

As seen in Figure 1.4, click and scroll motions combine to constitute the fundamental mode of modern computer operation. The index finger is the only finger involved in both click and scroll motions. Because finger movements are captured completely, users do not need to learn new ways to operate. This helps new users significantly reduce the cost of learning. This allows for a very fluid interaction, utilizing a single smart ring.

1.2.2 Touch Detection / Slide Detection

To provide an complete mouse-like operation feature, there needed to be a new way to represent the left click by middle finger. Here we propose to use thumb with touch-pad on the outside of the ring to perform the second degree of interaction. As demonstrated in Figure 1.5, a user can touch the outside of the ring to open the detail

list. This also help user to perform more complex gestures in 3D environment. User can select 3D object by point their finger to the object, then by touch the ring using thumb to confirm selection. The two fingers interaction mimics the pinch motion typically taken place in 3D operation. Additionally the touch-pad can perform not only touch detection, also can detect continual slide and microroll gestures. [24]

Figure 1.5: This is the first prototype device with a flexible touch-pad around the ring's surface.

As seen in Figure 1.5, the character may be maneuvered across the game's surroundings using sliding motions, similar to a touch screen joystick. As necessary, the user may micro-roll their thumb to change the camera, then switch back to sliding to keep the character moving. This enables highly fluid engagement with a single smart-ring. Additionally, the micro-rolling gesture might be incorporated to speed up the character's movement or facilitate the picking of items during gameplay [24].

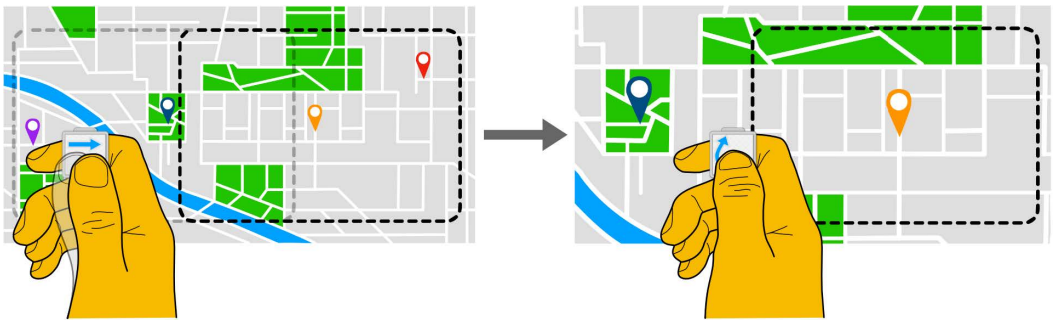

Figure 1.6: One of the map navigation application using continual slide and microroll gestures by researchers [24]

1.3 Thesis Contributions

In this work, we research and develop a ring type wearable device focusing on single finger motion tracking which include finger positioning, flexion, extension and abduction. The device combines with IMU sensor for finger motion and TOF sensor for finger flexion extension and abduction. The ring is a wireless, self-contain ring that not only tracking the absolute position and orientation of the ring, but also simulate the flexion and extension of the finger in real-time. The low-power, battery-operate ring combines absolution position and orientation and extension to present an complete model of single finger motion. As the user moves their finger, any locomotion of the finger can be capture by the ring. Sensors embedded in the ring measure these motion and use these measurements to estimate the pose of the ring with respect to the user body.

This ring is effectively a seven degree-of-freedom tracking system. The device will be tracking all data from triaxial accelerometer, triaxial magnetometer, triaxial gyroscope to generate a relative position reference to origin point and rotational motion of pose and two flexional components (extension and adduction). IMU sensor capture finger rotational motion and TOF sensor capture finger flexion/extension and abduction/adduction.

Compared to other finger tracking approaches, our approach simplifies relatively complex device form-factors to just a ring and does not require any computer vision to tracking finger motion. Our device leverages the insight that pure sensor-based tracking method is better compared to computer vision in low light environment. Our approach also delivers significant improvement in device functionality, compatibility,

and portability.

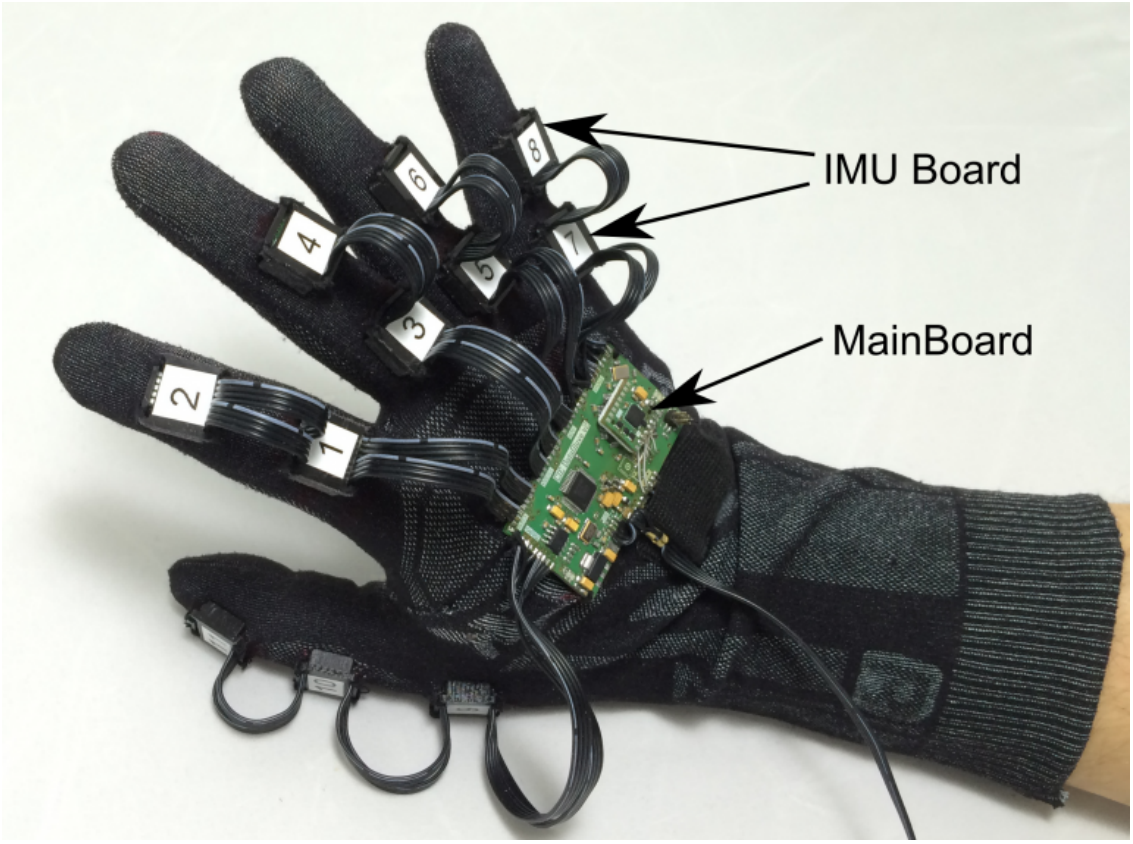

Figure 1.7: This is an typical data glove using IMU sensors to track hand motion [46].

Due to the usage of various motion capture technologies, the device's physical form also varies; there are bracelets, gloves[46], and rings. The glove can collect dynamic information about the hand and individual fingers, but it is not suitable for everyday use; the EMG band can also collect relatively comprehensive information about the hand, but if feedback on touch is required, the device cannot provide the most intuitive feedback due to its distance from the fingertips [29]. Because to the reduced size of the device, the ring can provide more direct vibration feedback and is more easily accepted by users. However, only a single finger operation will be recognized due to the size limits of the device. According to data of existing motions, the index finger is the most commonly and widely utilized of all fingers. Therefore, wearing the motion capture ring on the index finger may accommodate the majority of scenarios involving human-computer interaction.

IMU Finger motion tracking is one of most commonly used method to tracking motion with relatively accurate measuring scale in a variety of domains [32, 74]. Though precise, IMU motion tracking has a reputation for being low cost, easy to deploy and shielding from environmental interference.

There are so many different sensing methods, such as optical camera, and electromagnetic sensing, the accuracy of inertial measurement unit generally depends on the accuracy of the gyroscope. Due to the technical characteristics of using inertial measurement unit, Using IMU as the method for motion tracking fits the trend of wearable technology better. Our device can also be used to track the complete finger motion with flexion/extension and abduction/adduction. With the device, a user can just provide input using their finger while keeping their arm motionless at their side or resting on a table.

Our primary contributions include:

1. Single finger tracking method, including a physics-based iterative approach to 7-DoF pose estimation.

2. Prototype simulation using sensor breakout board to demonstrate the feasibility of using low-cost sensors to track finger motion.

3. The use of customized design smart ring hardware device with a rigid flex Printed circuit board (PCB) will provide a more accurate and reliable method for tracking finger motion compared to the prototype simulation using sensor breakout board.

4. The publication "Finger Tracking for Human Computer Interface Using Multiple Sensor Data Fusion" presented at the 2022 IEEE Canadian Conference on Electrical and Computer Engineering, describes a new approach to finger tracking using multiple sensors, and provides valuable insights for the development of human-computer interfaces.

1.4 Thesis Organization

The rest of this work is organized as follows. Chapter 2 describes the human finger kinematics and finger tracking method used in this work. Chapter 3 gives a detailed description of the proposed system controller design in this work. Chapter 4 describes the simulations of the finger motion based on finger joints angle. Chapter 5 describe the customized hardware and software used to perform the simulations and experimental work. It also describes the calibration of the control hardware. Chapter 6 describes the results for hardware system and discusses their implications. Chapter 7 summarizes the conclusions of this work and suggests areas for future research.

Chapter 2

Background Theory/Related Work

2.1 Human Finger Kinematics and Dynamics

The hand is made up of several bones, muscles, and ligaments that provide a great deal of mobility and dexterity [49]. There are three primary kinds of bones in the hand itself: The 14 Phalanges bones that are located in the fingers and toes of each hand and foot. Each finger has three phalanges (distal, middle, and proximal), while the thumb has only two. The five bones that make up the central portion of the hand [7, 49]. The eight bones that make up the wrist. The two rows of carpal bones are attached to the ulna bone and the radius bone of the arm [22].

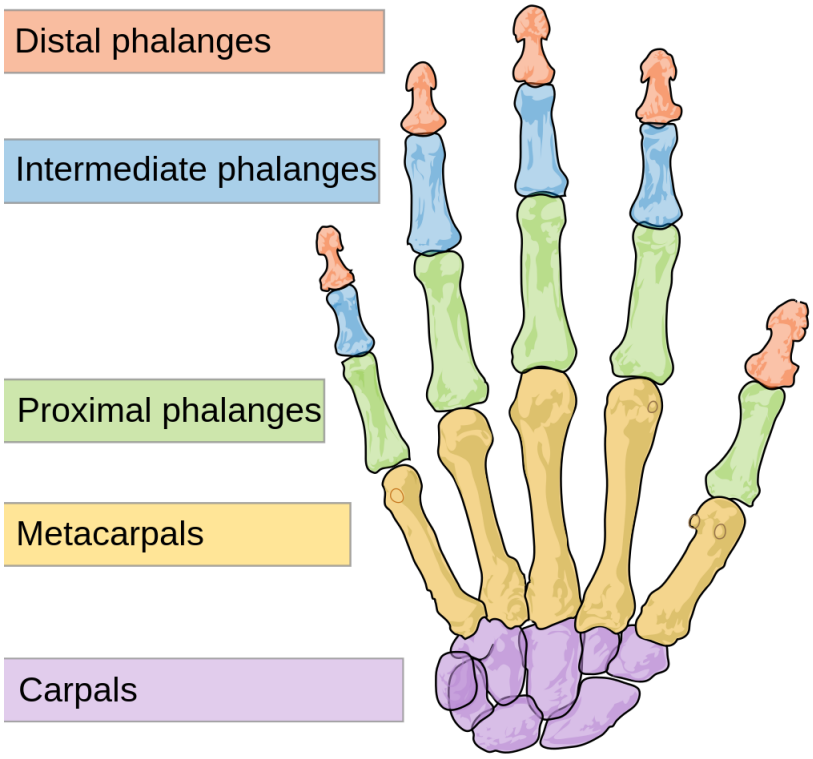

Figure 2.1: 3 major types of bones in the human hand, including phalanges, metacarpals, carpals.

The hand has several muscles, ligaments, tendons, and sheaths. The muscles are the structures that may contract, so enabling the hand's bones to move [13, 20, 53]. The ligaments are fibrous structures that link the joints of the hand together. Sheaths are tubular structures that encompass a portion of the fingers. Tendons link the arm or hand muscles to the bone to enable movement.

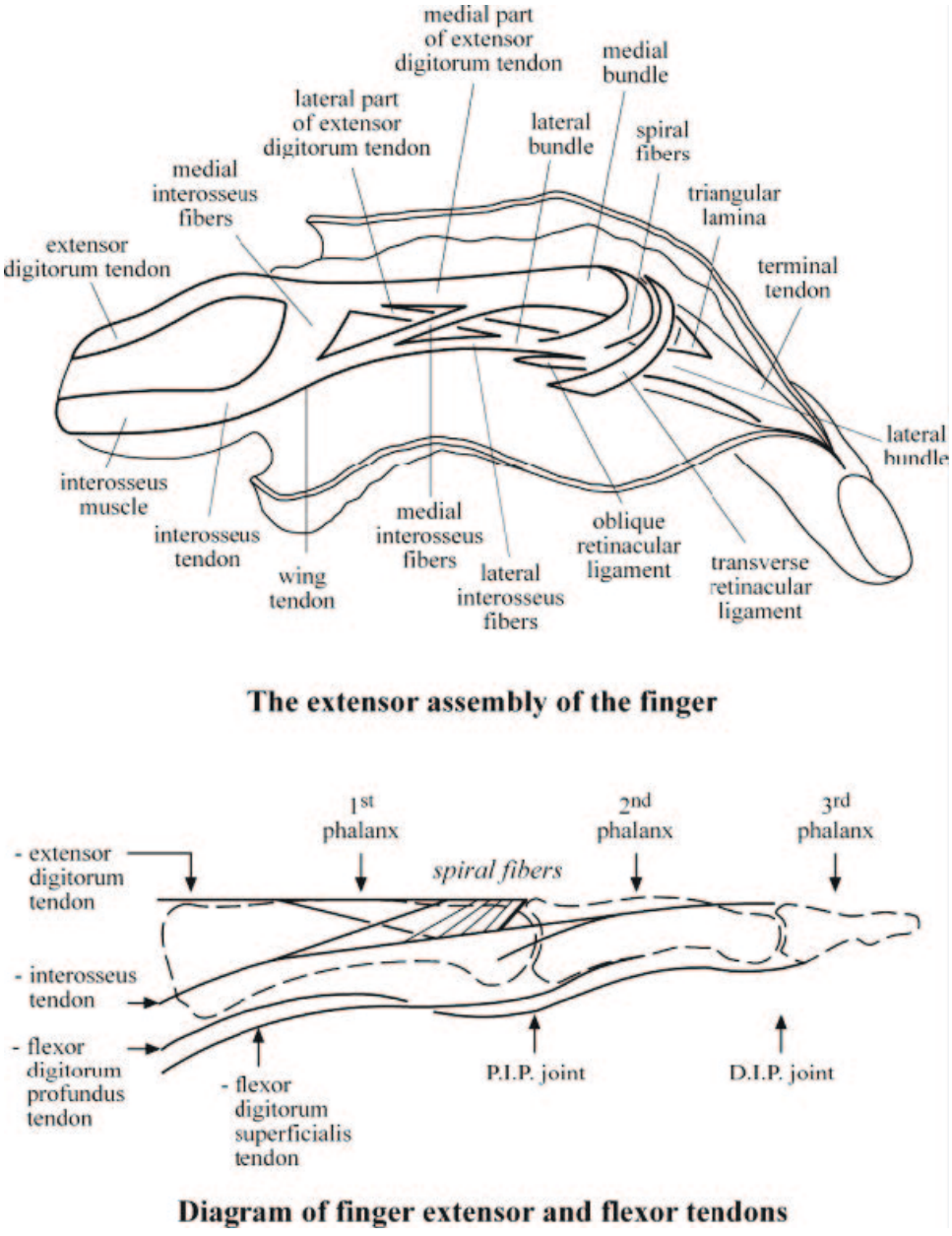

Figure 2.2: Diagram of the extensor and flexor tendons of the finger's extensors. [78]

The extensor assembly of the human finger is made up of tendon fibers from multiple muscles, including the extensor digitorum, interosseus, and lumbrical muscles, and is connected by fibrous ligaments. It stretches the proximal and distal interphalangeal joints, while flexion is done by tendons from the flexor digitorum superficialis and flexor digitorum profundus muscles [64]. The extensor assembly consists of a medial bundle that passes over the proximal interphalangeal joint and two lateral bundles that unite distally into a single terminal tendon. This terminal tendon passes over the distal interphalangeal joint regardless of its flexion angle. The spiral fibers of the extensor assembly link the lateral and central bands of the finger and are located at the distal portion of the first phalanx and at the proximal interphalangeal joint level. According to various authors, these spiral fibers are responsible for controlling palmar displacements of the lateral bands during flexion of the proximal interphalangeal joint [78].

K.J.Van [78] developed an analytical mathematical functional relationship between the D.I.P–P.I.P flexion angles using the model. The Proximal interphalangeal joint (P.I.P.) angle is arbitrarily selected as the independent variable, while the Distal interphalangeal joint (D.I.P.) angle is designated as the dependent variable. Equation 2.1 determines the equivalent D.I.P. angle for any given P.I.P. angle.

$$\varphi(\theta) = \frac{\sqrt{a_x(0)^2 + a_y(0)^2} + s(0) - \sqrt{a_x(\theta)^2 + a_y(\theta)^2} - s(\theta)}{R(\theta)} \tag{2.1}$$

The relevant numbers are defined in Fig. 2.3, which depicts a simplified lateral view of the extensor system with the middle phalanx at its centre and the finger in flexion with the angles θ and φ. [78] .

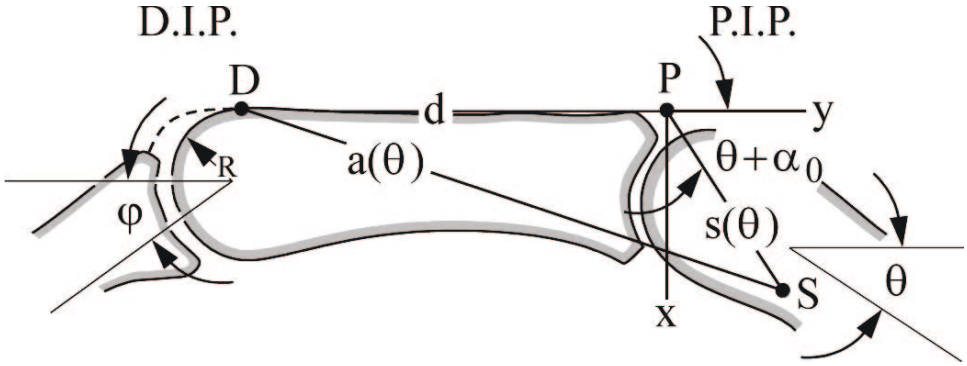

Figure 2.3: Finger extensor assembly schematically. [78]

The length of the distance a(θ) and the length of the (terminal) spiral fibre s(θ) may be expressed using the points D, P, and S from Figure 2.3. The a(θ) is the distance between point D and point S, the a(θ) is the length between point P and point S [78].

When finger joint angles $\theta = \varphi = 0$, the starting values of a(0) and s(0) correspond to the lengths of the lateral tendon and terminal spiral fibre, respectively [78].

The longitudinal distance between the D.I.P. and the P.I.P. joint at the dorsal side of the finger is represented by the distance d between the points D and P. This constant may be determined from anatomical information[78].

The spiral fiber s(0) forms an initial angle of $\alpha(0)$ with the reference line when the finger is extended [78].

Because the distance a(θ) and the length of the (terminal) spiral fibre s(θ) can be expressed by their components in x axis and y axis, from the initial position of the finger, the x component could be neglected in equation 2.1. So the depend on the variable θ, with some simple trigonometry, the mathematical expressions between the lengths a(0), a(θ), the distance d and the fiber lengths s(0), s(θ) as shown below[78]:

$$a_y(0) = s(0) \cdot \cos\alpha_0 + d \tag{2.2}$$

$$a_y(\theta) = s(\theta) \cdot \cos(\theta + \alpha_0) + d \tag{2.3}$$

With y axis component expressions of the lengths a(0), a(θ), the distance d and the fiber lengths s(0), s(θ) apply back to equation 2.1, we get the expression [78]

$$\varphi(\theta) = \frac{\sqrt{(s(0) \cdot \cos\alpha_0 + d)^2} + s(0) - \sqrt{(s(\theta) \cdot \cos(\theta + \alpha_0) + d)^2} - s(\theta)}{R(\theta)} \tag{2.4}$$

and assuming R $\approx$ s $\approx$ constant, we can finally write [78]

$$\varphi(\theta) = \cos\alpha_0 - \cos(\alpha_0 + \theta)[78] \tag{2.5}$$

At low P.I.P. values, Equation 4.1 accurately captures the angular correlation; but, for larger P.I.P. angles, it significantly differs from the actual results [78].

2.2 Finger Tracking

The goal of a finger tracking system is to provide an intuitive transfer of motion between gestures and finger movements [26, 27, 39, 40, 47]. There are numerous implementation possibilities for finger tracking, primarily those utilizing an interface or not. By applying these rotations to a kinematic chain, it is possible to track the entire human hand in real time, without occlusion and wirelessly [26, 27, 39, 40, 47]. Each finger segment of hand inertial motion capture devices, such as Synedrial mocap gloves, contains a miniature IMU-based sensor. For precise capture, at least 16 sensors are required. There are also mocap glove variants with fewer sensors for which the remaining finger segments are interpolated or extrapolated (proximal segments) (distal segments) [22]. Typically, the sensors are put within a textile glove, which makes their use more pleasant [46].

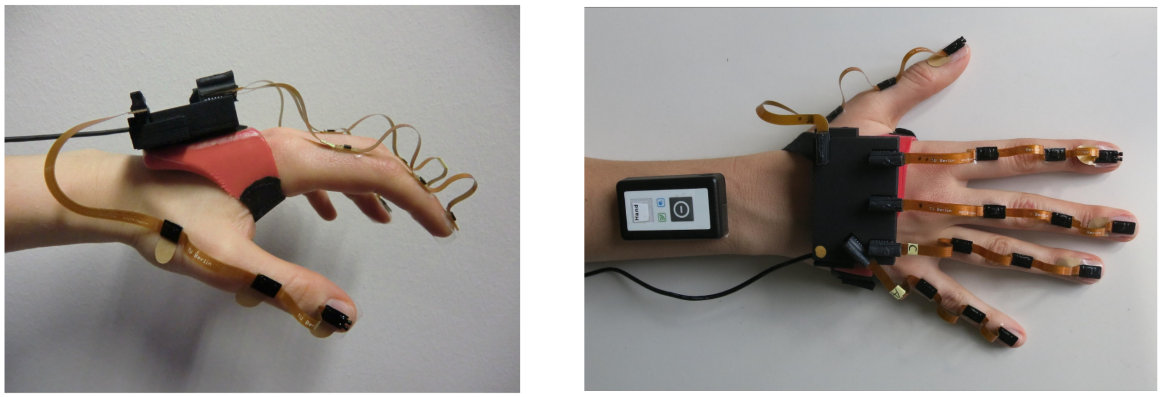

Figure 2.4: Real-time motion tracking of the locations of the fingertip using a modular hand sensor system based on an IMU. Up to five sensor strips with three IMUs each make up the system, which also includes a base unit on the back of the hand and a wireless IMU on the forearm. [59].

Inertial Measurement Units (IMUs): IMUs use accelerometers, gyroscopes, and magnetometers to measure the movement and orientation of an object in three-dimensional space. In the context of finger motion tracking, IMUs are often embedded in a wearable device, such as a ring, to track the movement of the finger. IMUs are relatively low-cost and easy to implement, but they can suffer from drift and may require regular calibration to maintain accuracy [32, 74].

Optical tracking: Optical tracking systems use cameras or other sensors to detect the position of markers placed on the fingers or hands. These systems can provide

very high accuracy and precision, but require a line of sight between the sensors and the markers, which can limit their use in certain applications [18, 65].

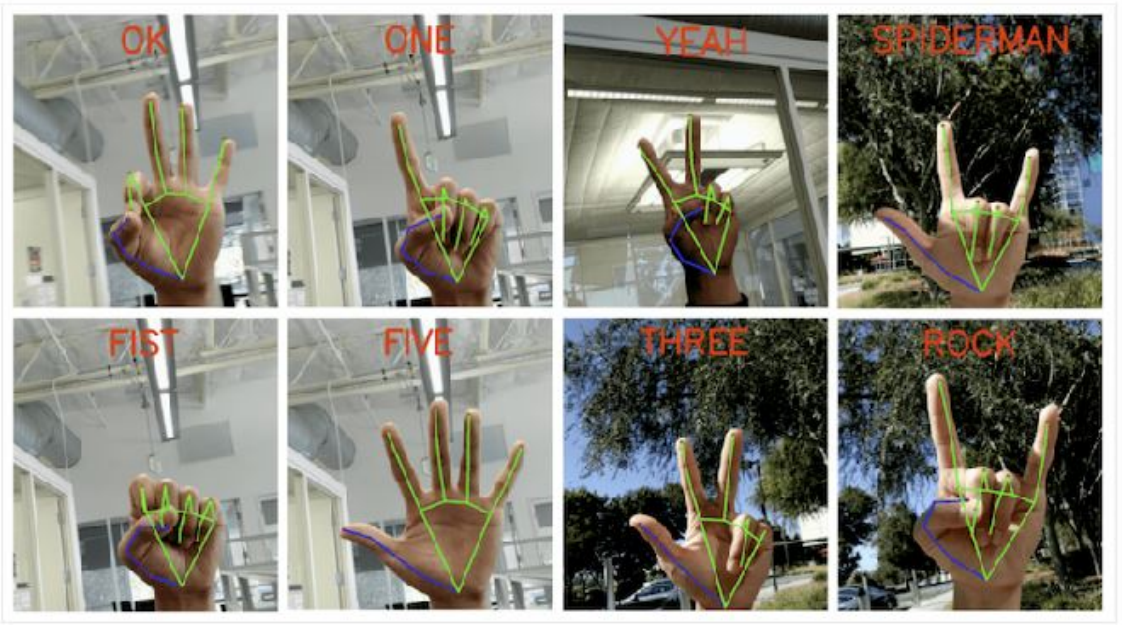

Figure 2.5: This is an typical data glove using IMU sensors to track hand motion [46].

Electromyography (EMG): EMG sensors detect electrical signals produced by muscle contractions and can be used to track finger movement based on muscle activity. EMG is very precise and can provide real-time feedback, but requires electrodes to be attached to the skin, which can be uncomfortable for the user [5, 34].

Capacitive sensing: Capacitive sensing works by detecting changes in capacitance between two electrodes, such as those on a touch screen. In the context of finger motion tracking, capacitive sensors can detect changes in capacitance caused by the movement of the finger [36, 38, 42, 48, 73]. This technology is often used in touchscreens and can provide a high level of sensitivity, but may not be as accurate as other methods for tracking complex finger movements [41, 44].

Magnetic tracking: Magnetic tracking systems use sensors to detect changes in magnetic fields, which can be used to track the movement of magnetic markers attached to the fingers or hands. Magnetic tracking can provide very high accuracy and precision, but requires the use of special equipment and can be expensive [60].

Overall, the choice of finger motion tracking technology will depend on the specific

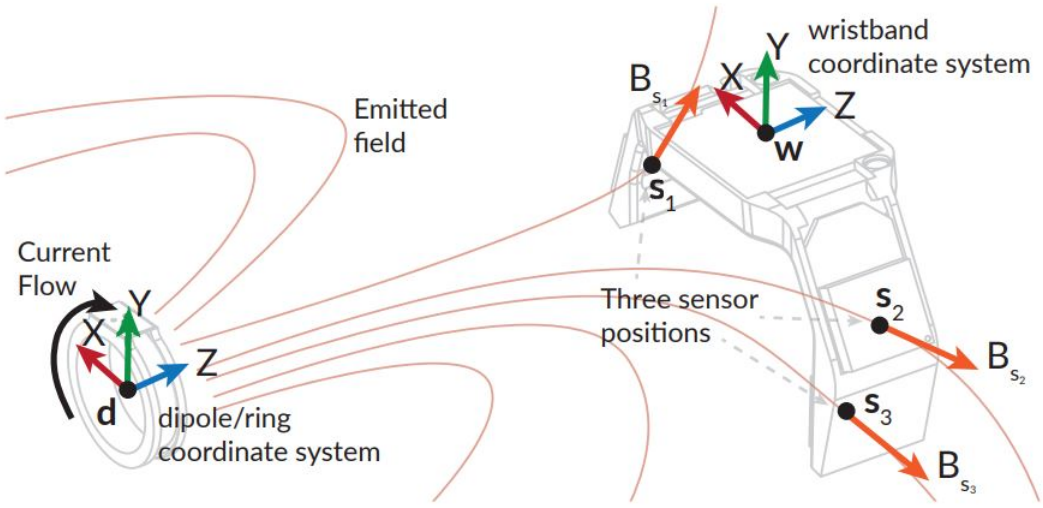

Figure 2.6: Precise Electromagnetic Finger Tracking by Auraring which is desgined by University of Washington [60].

application and requirements of the system. IMUs are often a good choice for wearable devices due to their low cost and ease of implementation, while optical tracking may be preferable for high-precision applications. EMG and capacitive sensing can provide unique benefits in certain contexts, while magnetic tracking may be the best choice for certain specialized applications [18, 65].

2.3 Inertial measurement unit (IMU)

Finger tracking technology using IMU sensors involves using an Inertial Measurement Unit (IMU) to track the movements of the fingers. An IMU is a device that combines accelerometers, gyroscopes, and sometimes magnetometers to measure motion, orientation, and gravitational forces [25, 33, 35, 50, 62].

An inertial measurement unit (IMU) uses accelerometers, gyroscopes, and sometimes magnetometers to measure and report a body's specific force, angular rate, and sometimes its direction. IMUs are referred to as IMMUs when they incorporate a magnetometer. [16] IMUs are often used to maneuver modern vehicles, such as motorbikes, missiles, airplanes (an attitude and heading reference system), uncrewed aerial vehicles (UAVs), and spacecraft, such as satellites and landers. Current advancements enable the creation of Global Positioning System (GPS) systems with

IMU functionality. An IMU enables a GPS receiver to function when signals are absent, such as in tunnels, buildings, or electrical interference [1, 2]. An inertial measurement unit detects linear acceleration with one or more accelerometers and rotational rates with one or more gyroscopes [68].

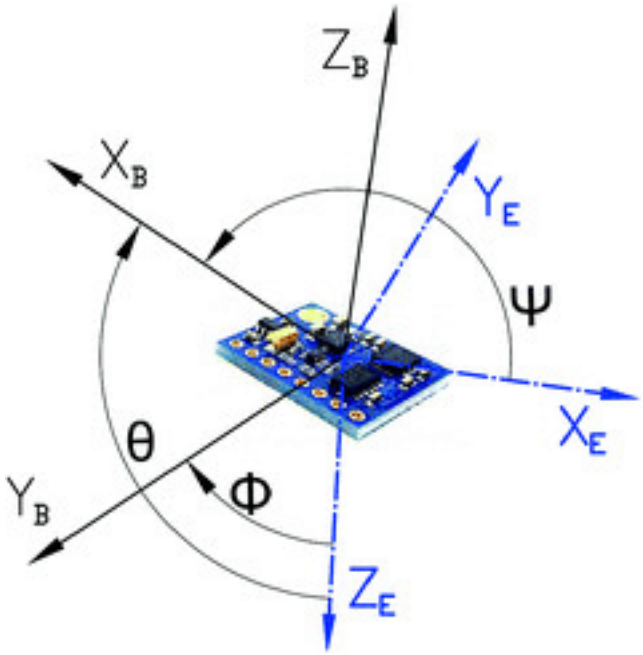

Figure 2.7: 9 axis IMU sensor breakout board can offer up to six degrees of freeedom.

To track finger motion, the IMU sensor is typically mounted on a small wearable device, such as a ring, and placed on one of the fingers. The sensor then detects and records the movement of the finger as it moves through space. By analyzing the data from the sensor, the system can accurately determine the position, orientation, and motion of the finger.

The process of finger tracking using an IMU typically involves several steps. Firstly, the IMU sensor must be calibrated to ensure accurate measurements. This involves setting the initial position and orientation of the sensor, and then collecting data on known movements of the finger to compare to the sensor data.

Once the sensor is calibrated, the system can begin tracking finger motion. The IMU sensor records changes in orientation and acceleration as the finger moves, and these measurements are then used to calculate the position and motion of the finger in 3D space. Algorithms are used to filter out noise and error in the data to provide accurate tracking [37].

One challenge of finger tracking using IMU sensors is the potential for signal interference or drift over time [8, 63], which can reduce the accuracy of the tracking. To address this, advanced algorithms can be used to compensate for drift and maintain accurate tracking over extended periods of time.

Overall, finger tracking technology using IMU sensors has many potential applications, including in sports and fitness tracking, virtual and augmented reality, and human-computer interaction. With continued advancements in sensor technology and algorithm development, the accuracy and precision of finger tracking using IMU sensors is expected to continue to improve in the future.

2.4 Time-of-Flight (ToF) Sensor

Similar to how a bat detects its environment, a ToF camera employs infrared light (lasers invisible to human sight) to calculate depth information. The sensor sends a light signal, travels to the object, and then returns to the sensor. The length of time needed to recover is then calculated and offers depth-mapping capabilities [11, 12, 52, 67]. This provides a significant advantage compared to previous technologies since it can precisely estimate distances over an entire scene with a single laser pulse.

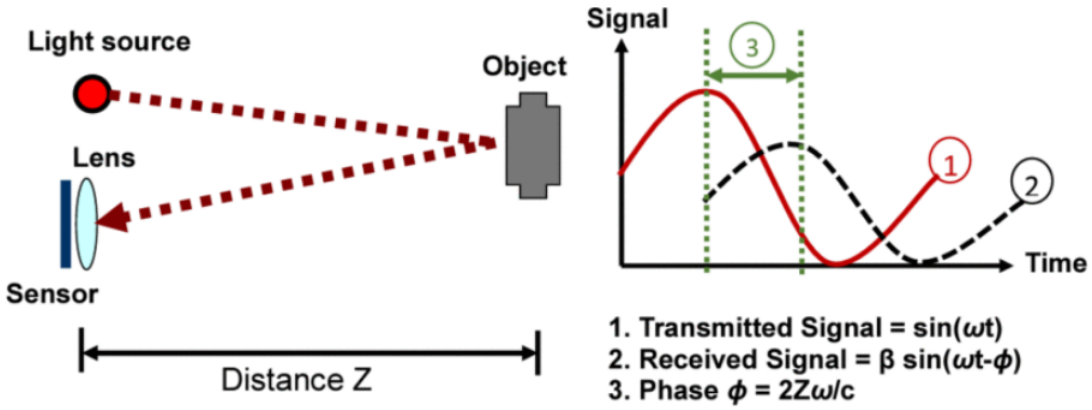

Figure 2.8: Operation principle of ToF system[3]

ToF technology is reasonably priced compared to other 3D-depth range scanning options, such as a structured light camera/projector system. The sensors are excellent

for real-time applications like background blur in on-the-fly video since they can capture up to 160 frames per second (160 data relays per second). And they use very little computing power. And with the appropriate algorithms, features like object identification may be introduced once distance data has been gathered [6, 54].

As time-of-flight cameras provide distance images in real time, it is easy to track movements of humans. This allows new interactions with consumer devices such as televisions [17]. Another topic is to use this type of cameras to interact with games on video game consoles. The second-generation Kinect sensor originally included with the Xbox One console used a time-of-flight camera for its range imaging, enabling natural user interfaces and gaming applications using computer vision and gesture recognition techniques. Creative and Intel also provide a similar type of interactive gesture time-of-flight camera for gaming, the Senz3D based on the DepthSense 325 camera of Softkinetic [66]. Infineon and PMD Technologies enable tiny integrated 3D depth cameras for close-range gesture control of consumer devices like all-in-one PCs and laptops (Picco flexx and Picco monstar cameras) [51].

2.5 Ring Device

A smart ring device is a small electronic wearable that can be worn on the finger like a regular ring. It is designed to track various activities and health metrics of the user. The ring contains a variety of sensors, including accelerometers, gyroscopes, and magnetometers, which allow it to track the movement and orientation of the finger in 3D space [77, 59].

The device is typically connected to a smartphone via Bluetooth, and the data it collects is sent to a dedicated mobile application for analysis and visualization. The app can display information such as step count, calories burned, heart rate, sleep quality, and other health-related metrics [71].

Smart rings can also be used for other purposes, such as controlling other smart devices, receiving notifications, and even making payments. Some models come equipped with Near-field communication (NFC) technology that allows them to function like a contactless credit card [24].

Smart rings are designed to be lightweight and comfortable, with a battery life that typically lasts for several days. They are often made from materials such as titanium

Figure 2.9: Oura Ring Gen3

or ceramic, and can be customized with interchangeable bands and faceplates to suit the user's style preferences.

Chapter 3

Proposed System Controller Design

3.1 System Architecture

During the proof-of-concept stage, our initial system comprised an Arduino Mega2560 Board, selected as the sensor control and data readout unit for its general high compatibility and availability of software examples. For the sensor portion of the system, sensor breakout boards are linked to the Arduino board using the I2C protocol. When all sensor data readings from the sensor breakout board have been captured through serial communication connection on a desktop computer. The proof of concept system is used to test the finger tracking software algorithm and hardware communication; nevertheless, the system was not wearable, which is one of the primary design issues that must be corrected for the final design. The final design incorporates a rigid-flexible PCB construction that allows the prototype to be shaped into a ring shape. The final ring-shaped design comprises a Bluetooth Low Energy (BLE) unit, a micro–Inertial Measurement Unit (mIMU) for finger tracking, a Time-of-Flight camera for finger flexion tracking, and a cell battery.

3.1.1 Rigid-Flex PCB

Rigid-flex PCBs are a specialized type of printed circuit board that combines both rigid and flexible PCBs. They consist of multiple layers of flexible circuits attached to rigid sections using adhesive or other bonding methods. The result is a single PCB that can flex and bend, making it ideal for use in devices with limited space and unusual shapes, such as smart rings.

In the case of a smart ring, the rigid-flex PCB is typically made up of multiple layers, with each layer serving a specific purpose. For example, one layer might contain the microcontroller and other electronics, while another layer contains the sensors and battery. The layers are then connected using flexible interconnects, which allow the

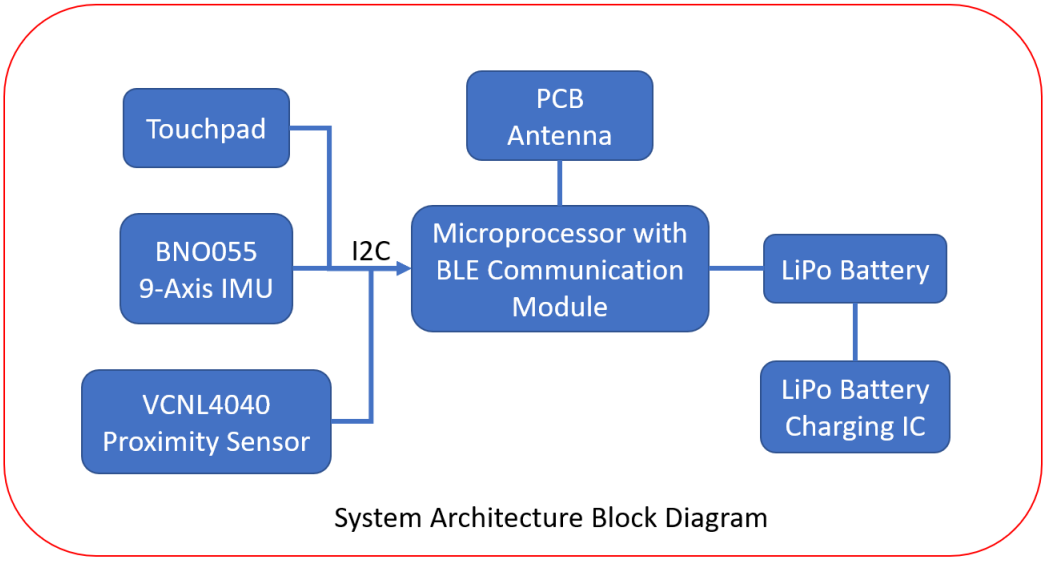

Figure 3.1: Purposed System Architecture Block Diagram

ring to bend and flex without damaging the components.

One of the key advantages of using a rigid-flex PCB in a smart ring design is that it allows for a compact and lightweight form factor. This is especially important for wearable devices, which need to be comfortable and unobtrusive for the user. Additionally, the use of a rigid-flex PCB can help reduce the number of interconnects and other components needed, which can simplify the design and assembly process.

However, designing and manufacturing a rigid-flex PCB can be more complex and expensive than traditional PCBs [28], as it requires specialized software and equipment. Additionally, the flexibility of the board can also make it more susceptible to damage from bending and twisting, which must be taken into account during the design process.

The use of a rigid-flex PCB is an effective way to incorporate electronics into a smart ring design, providing a flexible and lightweight solution that is well-suited to wearable devices.

In designing Print Circuit Board of the ring, we pursued a Flex-Rigid type PCB design that would present a highly integrated system that contain all the sensors we need and date transmitting ability through BLE (Bluetooth Low Energy).

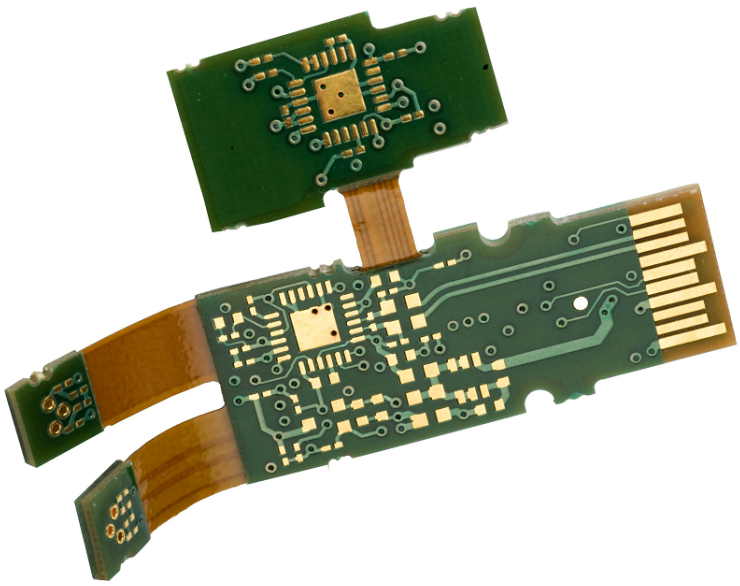

Figure 3.2: This illustration from the Alitum Design Software Guidebook provides a fantastic example of how differential signaling may be addressed utilising rigid-flex PCB design.

3.1.2 Arduino MEGA2560

Arduino MEGA2560 is a popular microcontroller board for embedded system development, and it is widely used in various applications, including motion tracking experiments. The MEGA2560 board has a 16 MHz crystal oscillator and 256 KB of flash memory, 8 KB of SRAM (Static random-access memory), and 4 KB of EEPROM (Electrically Erasable Programmable Read Only Memory). It has 54 digital input/output pins, 16 analog inputs, and 15 PWM (Pulse Width Modulation) outputs, which are more than enough to interface with different sensors for finger motion tracking.

Io order to use Arduino MEGA2560 for finger motion tracking experiments, We did the following steps to conduct the finger motion tracking experiment:

First, we identify the sensors needed for the experiment, various sensors can be used for finger motion tracking, such as accelerometers, gyroscopes, magnetometers, flex sensors, or force sensors.

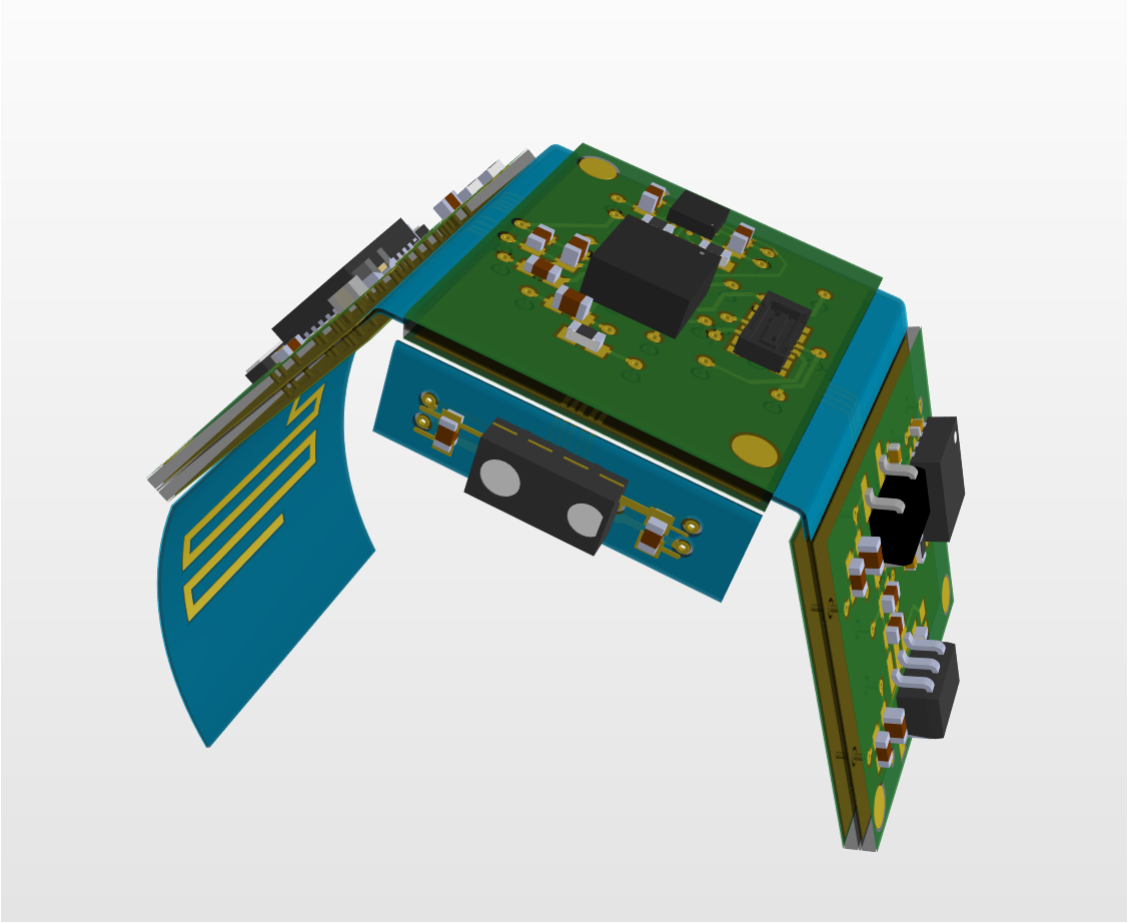

Figure 3.3: Flex-Rigid Prototype PCB Design.

We choose I2C communication protocol as the appropriate interface for the sensors. Depending on the sensor type, different interfaces can be used to connect them to the microcontroller. For example, accelerometers, gyroscopes, and magnetometers typically use I2C or SPI interfaces, while flex sensors and force sensors may use analog inputs.

After the hardware setup complete. we wrote code to collect data from the sensors: Using the Arduino Integrated development environment (IDE), code can be written to collect data from the different sensors, process it, and store it in memory or transmit it to a computer for further analysis.

Before using the sensors for motion tracking, they need to be calibrated to eliminate any bias or offset that might affect the accuracy of the data. This can be done using various techniques, depending on the sensor type and the desired level of

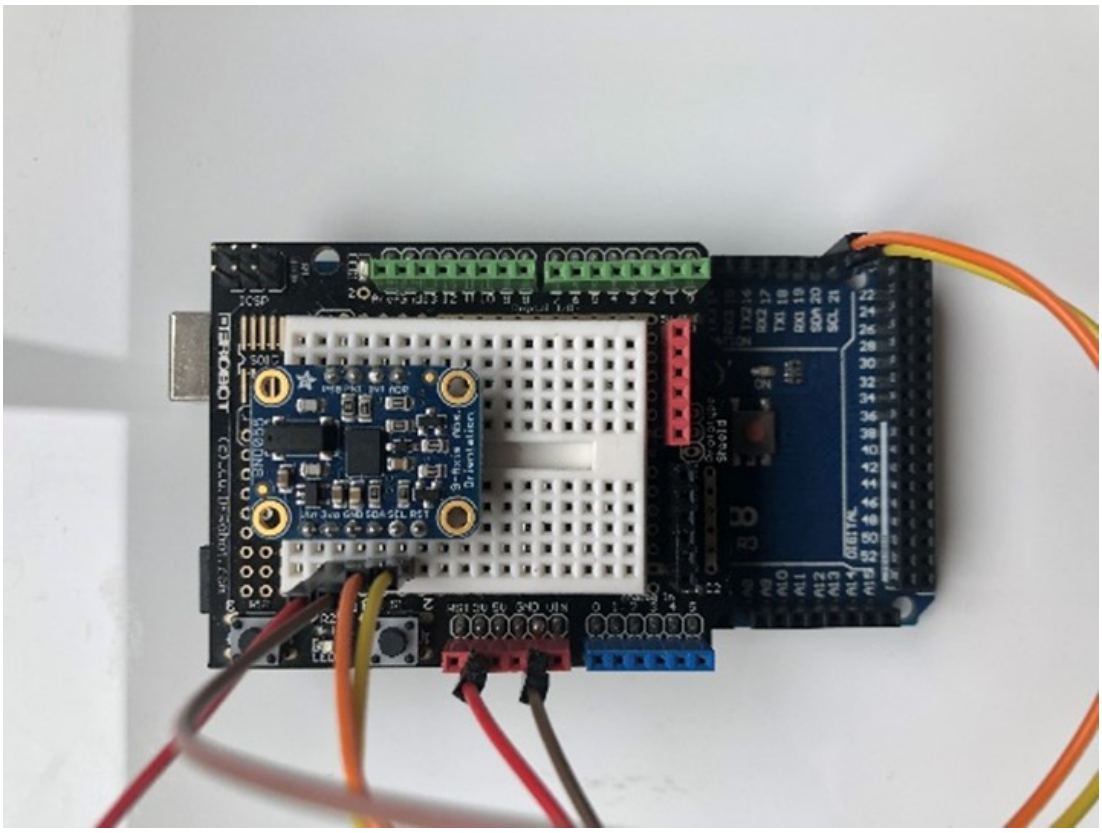

Figure 3.4: Arduino MEGA2560 and Adafruit BNO055 Absolute Orientation Sensor

accuracy.

We combined the data from different sensors to obtain a more accurate representation of finger motion. For example, combining data from accelerometers and gyroscopes can give a more accurate measurement of finger orientation and movement.

Finally After we collecting and combining data from different sensors, it can be analyzed and visualized using various tools, such as MATLAB or Python, to gain insights into finger motion patterns and behaviors.

Using Arduino MEGA2560 as a microcontroller for finger motion tracking experiments provides a low-cost and flexible solution that can be easily customized and adapted to various sensor types and configurations. Moreover, the Arduino community offers a wealth of resources, libraries, and code examples that can facilitate the development and implementation of such experiments.

To validate the finger tracking algorithm using BNO055 IMU sensor, we use a

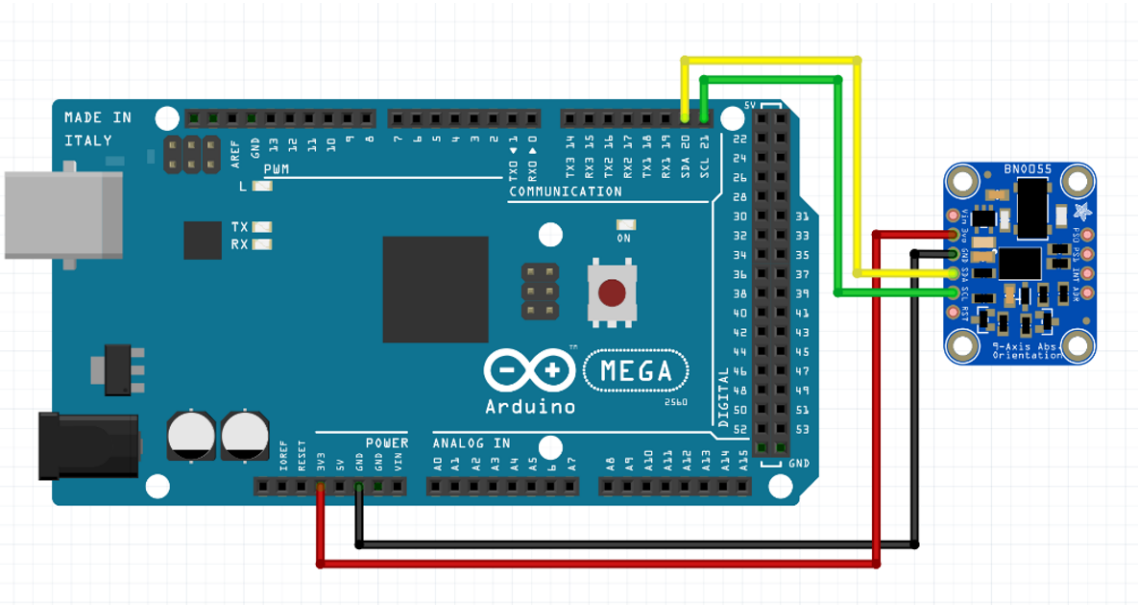

Figure 3.5: Arduino MEGA2560 and Adafruit BNO055 Absolute Orientation Sensor Breakout Board Wire Dirgram

BNO055 breakout board and Arduino MEGA2560 run simulation experiments. To connect the assembled BNO055 breakout to an Arduino MEGA2560, follow the wiring diagram below. The Vin pin is the power supply input for the breakout board, so this pin connects to 3.3V voltage supply pin on the Arduino board. The Ground (GND) is the reference ground pin for the power and data ground, so the ground need connect to the same ground as Arduino board. SCL (Serial Clock) pin control the serial clock speed for in order to communicate using I2C protocol, the SCL pin need connect to the I2C clock SCL pin on Arduino. SDA (Serial Data) is the serial data transfer line to transfer sensor data from breakout board to Arduino.

3.2 Number of Sensors Required

3.2.1 BNO055

In the proposed system, we use the device designed by our laboratory which have motion tracking sensor inside. And the motion tracking sensor BNO055 whose specification of the sensor is listed in Table 3.1 and specification of each axes is listed in Table 3.2

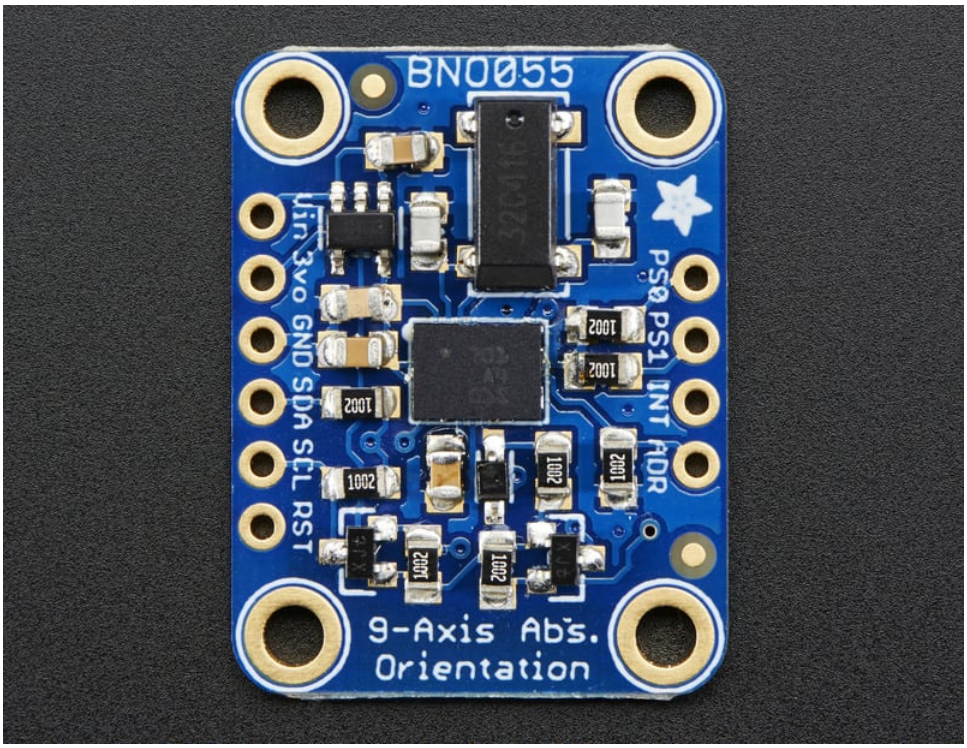

Figure 3.6: Flex-Rigid Prototype PCB Design.

Table 3.1: SPECIFICATION OF BOSCH BNO055

Mode	BNO055 by BOSCH
Type of Sensors	Accelerometer, Gyroscope, Magnetometer
Power Consumption	0.33mA in low power mode
Sampling Rate	100 Hz

Bosch BNO055 is a 9-axis Inertial Measurement Unit (IMU) sensor that is commonly used for finger motion tracking in wearable devices such as smart rings. The sensor integrates three sensors: an accelerometer, gyroscope, and magnetometer, which work together to provide accurate and reliable motion data.

The accelerometer measures linear acceleration, while the gyroscope measures rotational velocity, and the magnetometer measures magnetic field strength. This combination of sensors enables the sensor to determine the orientation and movement of the ring in three-dimensional space, which is crucial for tracking finger motion.

BNO055 also features an on-board processor that fuses the data from the accelerometer, gyroscope, and magnetometer into a single output signal. This output signal contains the orientation data, and the fusion process helps to reduce errors and

drift in the data, resulting in more accurate and stable tracking.

Furthermore, BNO055 provides built-in calibration features to reduce the effects of temperature and other external factors on the sensor's performance. The sensor can perform both self-calibration and user calibration to maintain optimal accuracy over time.

The Bosch BNO055 is a popular choice for finger motion tracking due to its high accuracy, reliable performance, and built-in calibration features.

Table 3.2: SPECIFICATION OF EACH SENSORS IN BNO055

Sensors	Accelerometer	Gyroscope	Magnetometer
Full-Scale Range	+/- 4 g	2000 s	15
Power Consumption	14 bits	16 bits	15 bits
Bandwidth	62.5 Hz	32 Hz	20 Hz

3.2.2 VCNL4040

The VISHAY VCNL4040 is a proximity sensor that can also be used to measure the angle of a joint. The sensor works by emitting an infrared light and measuring the amount of light that is reflected back. By placing the sensor at a specific distance from the joint and measuring the reflected light, it is possible to calculate the angle of the joint.

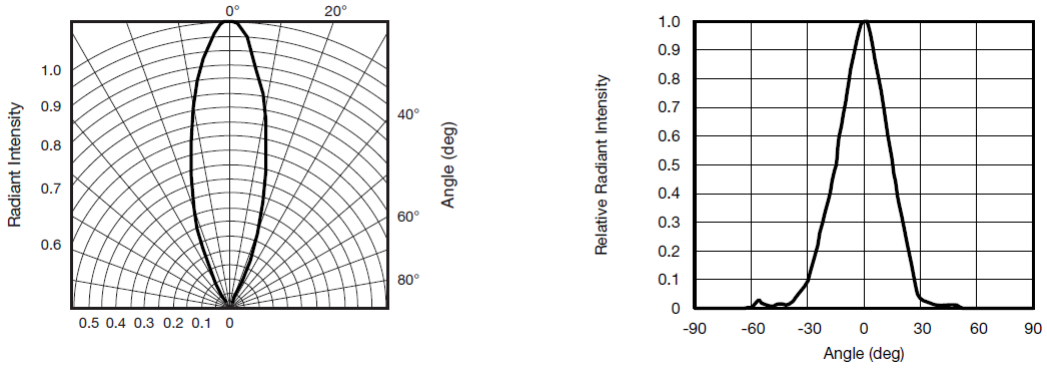

Figure 3.7: VCNL4040 IRED Profile

The VCNL4040 combines a proximity sensor (PS), an ambient light sensor (ALS), and a high-power IRED in a compact design. The CMOS (complementary metal-oxide

semiconductor) method integrates photodiodes, amplifiers, and analog-to-digital conversion circuits into a single chip. The 16-bit high-resolution ALS provides outstanding sensing capabilities with good options to satisfy most applications, regardless of whether the lens design is opaque or very transparent. Both high and low interrupt thresholds may be specified for ALS and PS, enabling the component to utilize as few microcontroller resources as possible.

The proximity sensor has an advanced canceling mechanism to prevent cross-talk efficiently. To accelerate the PS reaction time, intelligent persistence avoids proximity sensing misjudgments and maintains a rapid response rate. In active force mode, a single measurement may be requested, providing further design flexibility and power savings to accommodate a variety of applications.

To use the VCNL4040 for finger joint angle measurement, the sensor should be placed on one side of the joint and a reflector should be placed on the other side. The reflector can be any material that reflects infrared light, such as aluminum foil or a white surface. When the joint is flexed, the angle between the reflector and the sensor changes, causing the amount of reflected light to change. By measuring the amount of reflected light, it is possible to calculate the angle of the joint.

One advantage of using the VCNL4040 for finger joint angle measurement is that it is a non-contact sensor. This means that it can be used to measure joint angles without requiring any physical contact with the finger. This is particularly useful for applications such as virtual reality or augmented reality, where it is important to track finger motion accurately without impeding movement or causing discomfort to the user.

Another advantage of the VCNL4040 is that it is a relatively low-cost sensor compared to other types of joint angle sensors, such as potentiometers or strain gauges. This makes it an attractive option for researchers or hobbyists who want to experiment with finger motion tracking without breaking the bank.

In summary, the VCNL4040 proximity sensor can be used to measure finger joint angles by emitting an infrared light and measuring the amount of light that is reflected back. It is a non-contact sensor that is relatively low-cost and can be used in a variety of applications.

Chapter 4

Methodology

This chapter describes the angle calculation of the P.I.P finger joint angle with aid of the ToF sensor proximity data collection [78]. the basic idea of angle calculation this approach in this chapter is to using ToF sensor which consist of an infrared emitter with an LED and a detector.Together, the infrared emitter and the detector can serve as the infrared proximity sensor. For instance, the amount of IR light detected by the detector can relate to the distance between the ring device and the middle portion of the index finger of the user. In some cases, Lambertian reflectance of skin can be assumed, and light intensity can fall off quadratically with distance (e.g., inverse square law). The distance d can change when the index finger is flexed or extended.

4.1 Coordinate Systems

The coordinate system attached to a finger is established on the axis of the proximal phalanges of index finger. In most human motions, the proximal phalanger generally make transnational motion with rotation, therefor, in the case of only calculating the P.I.P joint angle, the proximal phalanger can be considered as a fixed bone [22, 78].

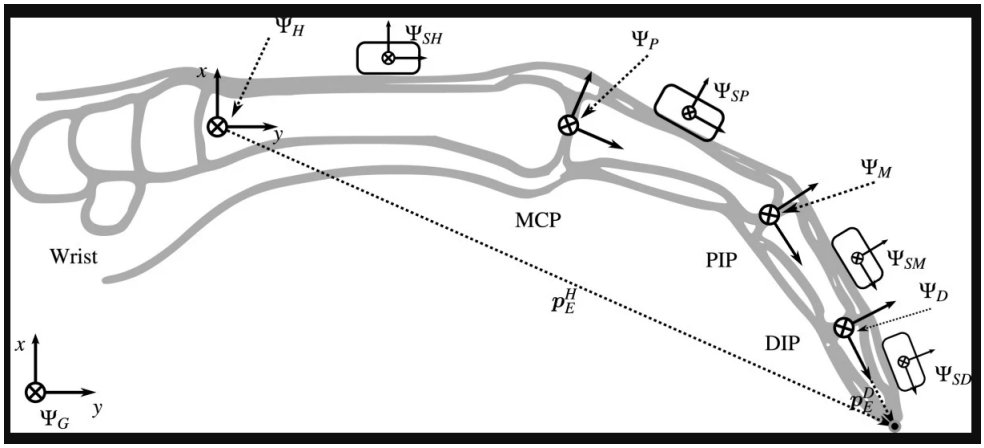

Figure 4.1: The coordinate frame definitions for the left index finger[23]

In finger motion tracking experiments using 9-axis IMUs, three coordinate systems are commonly used to represent the orientation and motion of the device:

Inertial Coordinate System: The inertial coordinate system is a fixed frame of reference relative to the Earth's gravitational field. The three axes of the inertial coordinate system are typically defined as X, Y, and Z, where the Z-axis points upwards perpendicular to the ground, the X-axis points towards magnetic north, and the Y-axis completes a right-handed coordinate system

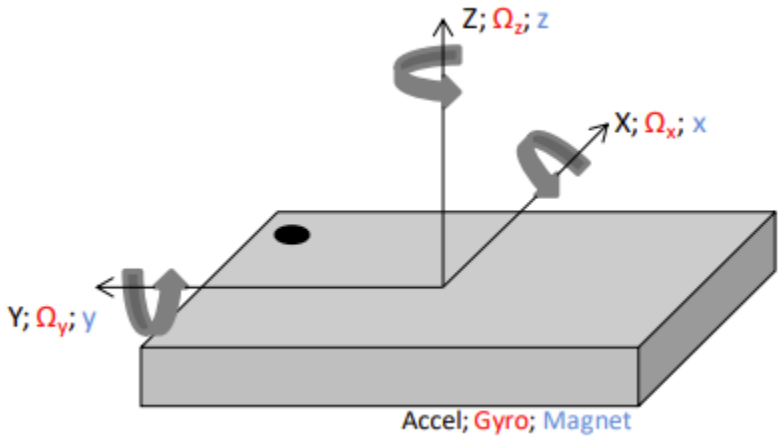

Figure 4.2: The BNO055 Inertial Coordinate System from Bosch BNO055 Datasheet

Body Coordinate System: The body coordinate system is a frame of reference attached to the device being tracked. The X-axis is usually aligned with the long axis of the device [23], the Y-axis is perpendicular to the X-axis and the Z-axis is perpendicular to both the X and Y axes, completing a right-handed coordinate system.

Local Magnetic North Coordinate System: This coordinate system is defined relative to the Earth's magnetic field. The X-axis points towards magnetic north, the Y-axis points towards magnetic east, and the Z-axis is perpendicular to the Earth's surface.

In order to track finger motion, the orientation and position of the device must be measured in each of these coordinate systems. The orientation of the device relative to the inertial coordinate system can be determined using the integrated gyroscopes and accelerometers, while the orientation of the device relative to the local magnetic north coordinate system can be determined using the magnetometer. By combining

data from these sensors, the orientation of the device relative to the body coordinate system can be calculated.

To measure joint angles, additional sensors such as flex sensors or proximity sensors can be used. The output of these sensors can be calibrated and mapped to the joint angle of interest, such as the angle of the finger joint. Overall, combining data from multiple sensors and coordinate systems allows for accurate tracking of finger motion in real-time.

4.2 Introduction to Quaternions

Quaternions are a mathematical concept used in computer graphics and motion tracking to represent the orientation of a 3D object in space [59, 72]. In finger motion tracking, quaternions can be used to calculate the rotation of the finger joints and track their movement.

A quaternion is a four-dimensional complex number that can be represented as a scalar component and a vector component. The scalar component represents the rotation angle, and the vector component represents the axis of rotation. Together, they describe the orientation of the object in 3D space [59].

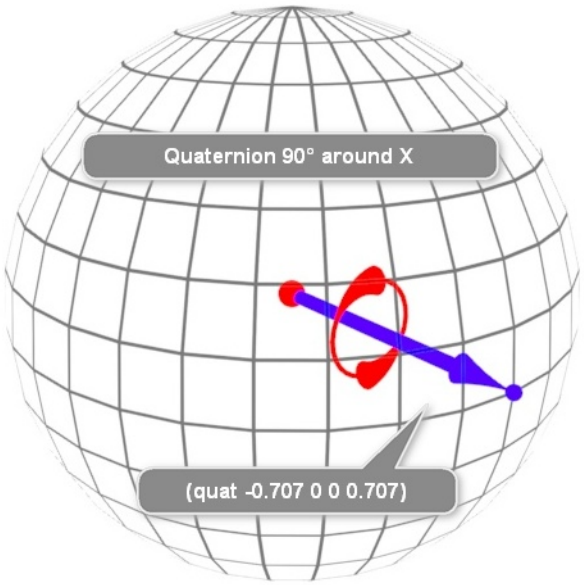

Figure 4.3: This figure shows quaternion representation of object rotated 90 degree around X by paul neale

To use quaternions for finger motion tracking, a 9-axis IMU sensor can be used to measure the acceleration, angular velocity, and magnetic field strength of the finger. The sensor data can then be converted to quaternion form using algorithms such as the Madgwick filter or the Kalman filter.

Once the sensor data is in quaternion form, it can be used to calculate the rotation of the finger joints. For example, the rotation of the proximal interphalangeal (PIP) joint can be calculated by subtracting the quaternion of the middle phalanx from the quaternion of the proximal phalanx. The rotation of the distal interphalangeal (DIP) joint can be calculated in a similar way by subtracting the quaternion of the middle phalanx from the quaternion of the distal phalanx [22, 78].

Quaternions have several advantages over other methods of motion tracking, such as Euler angles. They are not subject to gimbal lock [59], which is a phenomenon where the orientation of an object becomes ambiguous and difficult to track when it reaches a certain configuration. Quaternions are also more numerically stable and efficient than Euler angles, which can suffer from numerical errors and computational complexity.

In summary, quaternions can be used in finger motion tracking to calculate the rotation of the finger joints and track their movement. They provide a stable and efficient way to represent the orientation of the finger in 3D space, and can be used in conjunction with 9-axis IMU sensors to accurately track finger motion.

4.3 Sensor Fusion

Sensor fusion is the process of combining data from multiple sensors to obtain a more accurate and complete representation of the target being measured. In the context of finger tracking experiments using IMU and ToF sensors, sensor fusion involves combining the data obtained from the IMU and ToF sensors to calculate the angles and positions of the finger joints.

The IMU provides information about the orientation of the finger in space, while the ToF sensor measures the distance between the finger and the smart ring. By combining these two sets of data, it is possible to calculate the joint angles and positions in real-time.

Sensor fusion is typically performed using complex algorithms, such as Kalman

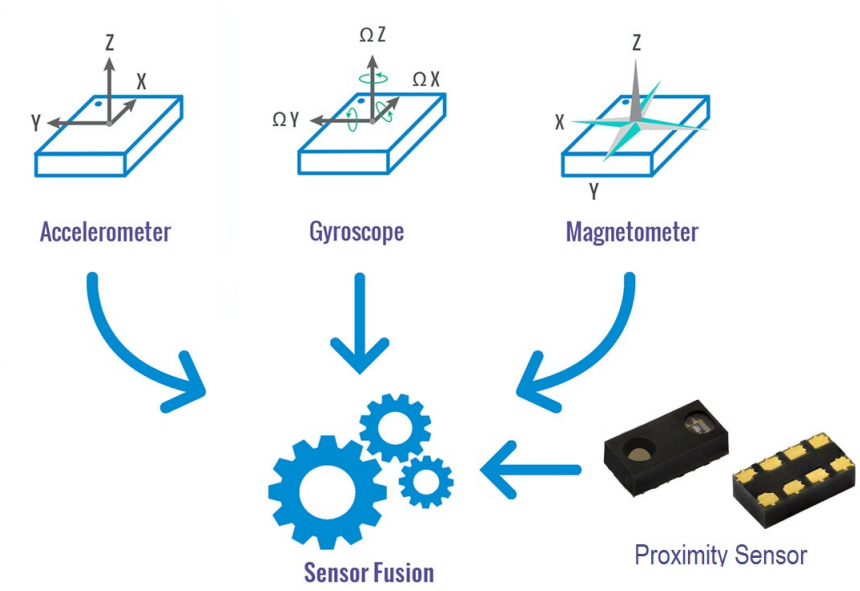

Figure 4.4: Combine 9 DOF IMU and Proximity Sensor into single finger motion tracking system with sensor fusion method

filters or complementary filters. These algorithms take into account the characteristics and limitations of each sensor and use statistical models to estimate the most likely joint angles and positions based on the available data.

In addition to sensor fusion, data recording is also an important part of finger tracking experiments. Data recording involves collecting and storing the data obtained from the sensors for later analysis. This allows researchers to study the motion of the finger in detail and identify patterns or anomalies that may be difficult to detect in real-time.

Overall, sensor fusion and data recording are essential components of finger tracking experiments using IMU and ToF sensors, enabling researchers to obtain accurate and detailed information about finger motion.

4.4 Determination of the Fingertip Positions

To calculate the P.I.P finger joint angles using the ToF sensor data, the flexion sensor's proximity signals related to the distance between the smart ring and the middle portion of the index finger can be utilized. The flexion sensor can measure the distance

and produce proximity signals related to the distance d. These signals can be changing proximity signals, such as successively detected signals, which can indicate changes in the distance d as the index finger is flexed or extended.

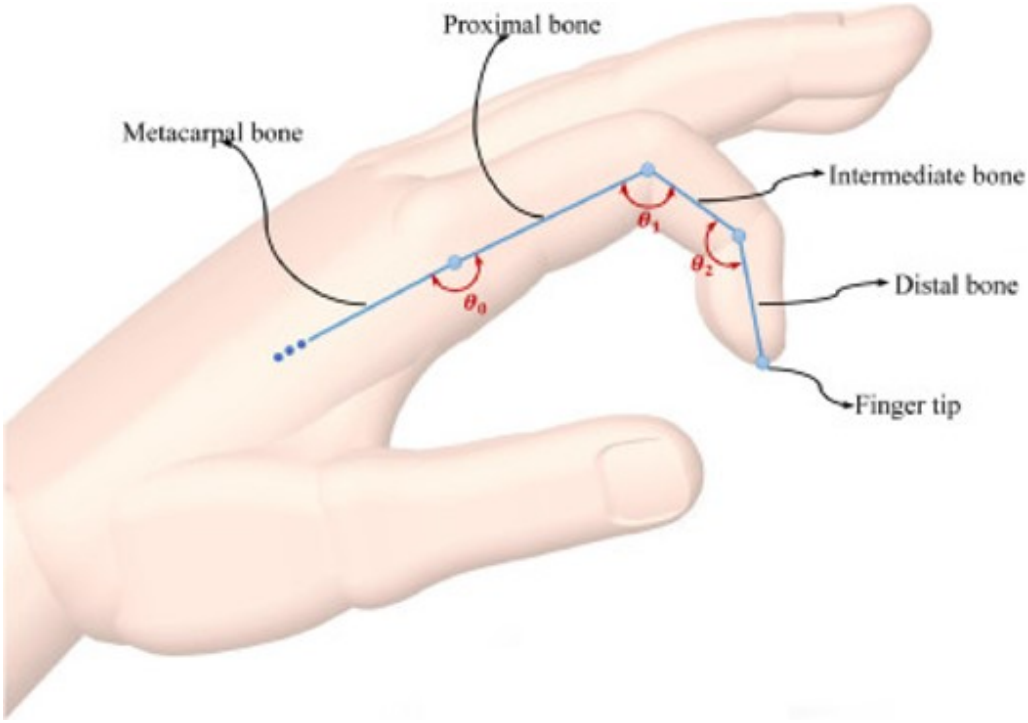

Figure 4.5: Figner tip location relate to bones and joints

In this implementation, the flexion sensor is an infrared proximity sensor consisting of an infrared emitter with an LED and a detector. The infrared emitter and the detector can serve as the infrared proximity sensor and the amount of IR light detected by the detector can relate to the distance between the smart ring and the middle portion of the index finger of the user. In some cases, Lambertian reflectance of skin can be assumed, and light intensity can fall off quadratically with distance, according to the inverse square law.

To prevent spurious reflections from other fingers or an input surface, the infrared emitter and detector can have a narrow viewing angle. By measuring the distance using the flexion sensor, the P.I.P finger joint angles can be calculated based on the relationship between the distance and the angle of the joint. This can be achieved by mapping the distance to a range of angles using a calibration process and then using this mapping to determine the angle based on the measured distance.

Using an analytical mathematical functional relationship between the D.I.P–P.I.P flexion angles developed by K.J.Van [78]. Equation 4.1 determines the equivalent D.I.P. angle for any given P.I.P. angle.

$$\varphi(\theta) = \cos\alpha_0 - \cos(\alpha_0 + \theta)[78] \tag{4.1}$$

In the final step, the fingertip position can be determined with the lengths of the phalanges and the D.I.P–P.I.P flexion angles [78].

Chapter 5

Experimental Setup

5.1 Hardware Configuration

The hardware configuration featured an Adafruit BNO055 Absolute Orientation Sensor, a 9-degrees of Freedom inertial measurement unit (IMU), an Arduino MEGA2560 microcontroller, and a SparkFun Proximity Sensor Breakout VCNL4040. To record finger orientation and motions, the BNO055 attaches to the top side of proximal phalanges. The bottom side of the proximal phalanges are where the proximity sensor breakout is connected. The hardware configuration utilised for this research project is shown in figure 5.1.

5.2 Sensor Calibration

The process of reducing the inaccuracies in the readings acquired from the IMU sensors is referred to as IMU sensor calibration. This process involves calculating and correcting the bias, scale factor, and misalignment parameters respectively. The calibration procedure is very necessary in order to guarantee precise measurements of the sensor readings made while monitoring the motion of the finger.

The calibration of the IMU sensor is comprised of two primary stages: the static calibration and the dynamic calibration. During the static calibration process, the sensor is maintained still while being rotated through a variety of positions so that the bias, scale factor, and misalignment errors may be estimated [21, 55, 31]. The measurements from the sensors that were gathered during this procedure are then input into an appropriate algorithm, such as the least squares approach, in order to determine the calibration parameters.

The author of the PJRC, Paul Stoffregen, developed an excellent cross-platform calibration tool that facilitates soft and hard iron magnetometer calibration.

The sensor is exposed to known acceleration and angular velocity patterns during

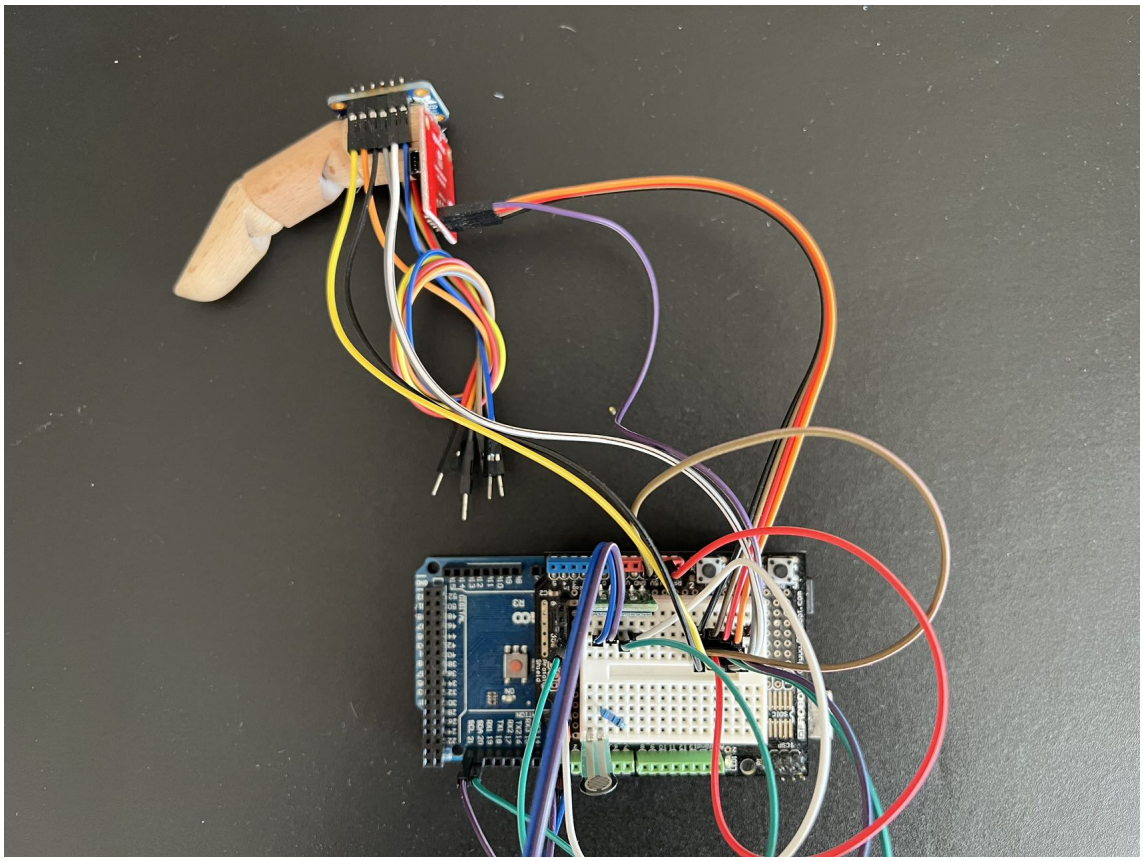

Figure 5.1: Finger motion tracking hardware configuration with multiple sensor breakout boards and microntroller on an wooden finger model

dynamic calibration. The data from the sensor are then compared with the predicted values in order to determine the inaccuracies that are still present, such as temperature drift and nonlinearities. During the finger motion tracking experiment, the predicted errors are applied as a correction factor to the values received from the sensors.

Calibration of an IMU sensor may be accomplished using a variety of approaches, including both direct and indirect procedures. Using a reference sensor, such as a GPS receiver, to make an approximation of the calibration parameters is what the direct technique entails [69, 75]. In order to correct the readings from the sensor, the indirect technique entails estimating the values of the calibration parameters based on the results of a prior calibration or reading them directly from the sensor datasheet.

IMU sensor calibration is essential for collecting precise measurements of the sensor readings in order to perform finger motion tracking. This is true regardless of the technique that is employed to do the calibration.

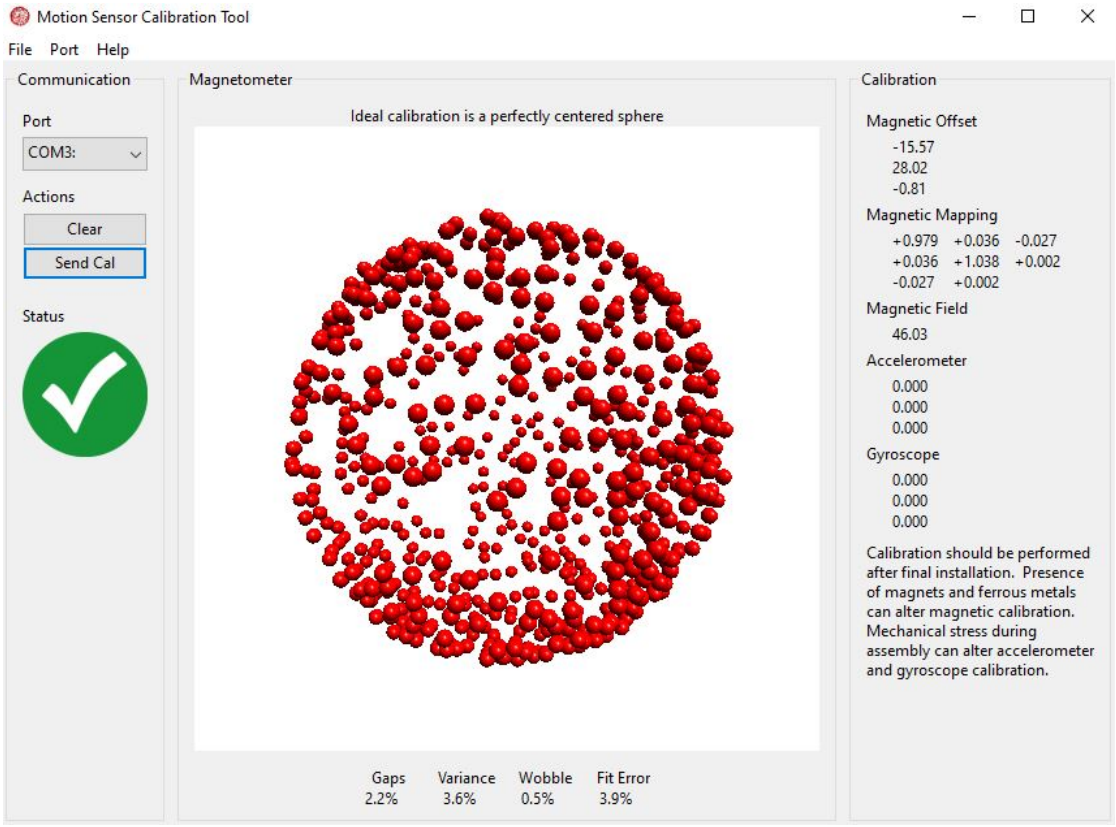

Figure 5.2: Magnetometer calibration profile created by using Motion Sesnor Calibration Tool developed by Paul

In most cases, the calibration procedures for IMU sensors consist of a series of mathematical operations designed to adjust for any mistakes that may occur in the sensor's measurements. Variations in temperature, flaws in the manufacturing process, and electromagnetic interference are only some of the causes of these problems, which may also be caused by other variables [9].

The "six-point" calibration is a typical method that is used for calibrating sensors. This calibration includes monitoring the output of the sensor in six distinct orientations. To acquire these orientations, which are commonly selected to reflect a range of pitch, roll, and yaw degrees, one rotates the sensor along several axes in a clockwise direction.

When the measurements have been obtained, the calibration algorithm will generate a set of correction values. These values may be applied to the readings that are produced by the sensor in order to alter them. During normal operation, the memory of the sensor is normally used to store these correction values, and the firmware of

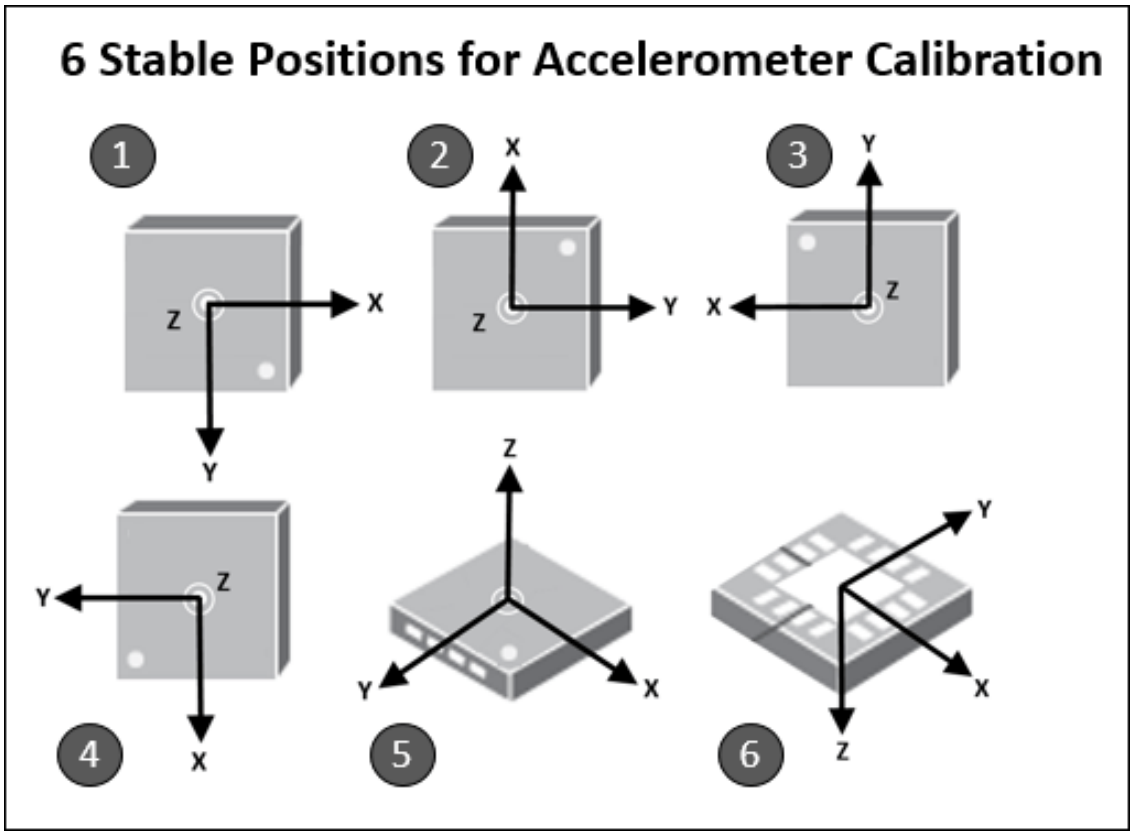

Figure 5.3: Six stable positions to fully calibrate the accelerometer of the BNO055 sensor from mathwork Calibrate BNO055 Sensors document

the sensor is responsible for applying them automatically.

Other calibration algorithms may make use of more advanced approaches, such as machine learning or neural networks, to make predictions about and corrections for sensor faults based on measurements that have been taken in the past. On the other hand, these algorithms could need more processing power and might be harder to put into action than more straightforward calibrating strategies.

5.3 MATLAB Simulation

Sensor data recording is the process of collecting and storing data from various sensors in a system. The recorded data can be used for analysis, visualization, and decision-making. In the case of finger motion tracking using IMU and ToF sensors, data recording involves collecting data from these sensors and storing it in a format that can be easily analyzed.

In the described scenario, the process of data recording involves connecting an Arduino board to an excel DAQ sheet through USB communication protocols. The PLX-DAQ version 2 [NetDevil Nov, 2016] data acquisition software, provided by the Arduino open-source community FORUM, is used to capture all IMU data and save it in an Excel sheet.

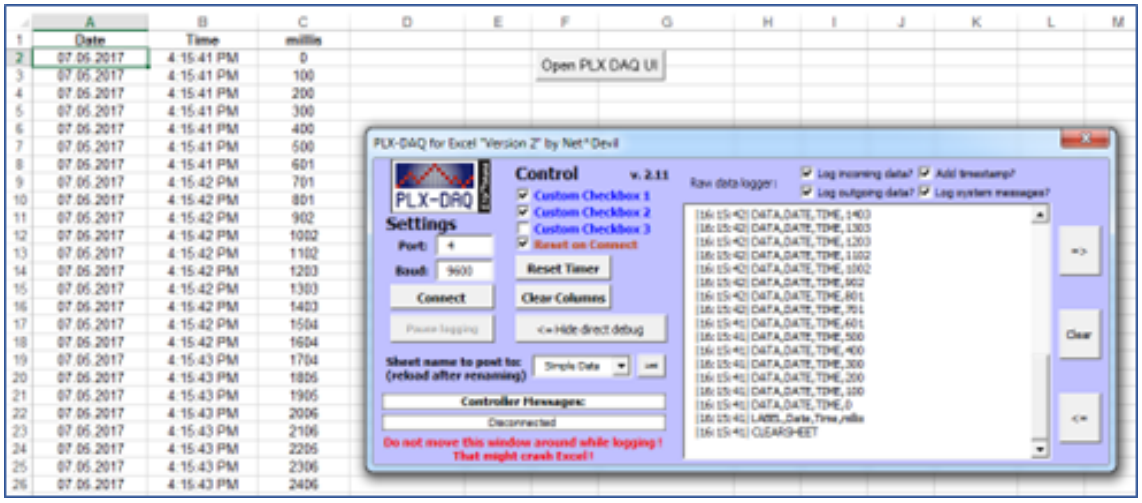

Figure 5.4: PLX DAQ v2 demo windows sample

The BNO055 has algorithms that are designed to continually calibrate the gyroscope, accelerometer, and magnetometer that are included inside the device.

The overall system calibration status, as well as individual values for the gyroscope, magnetometer, and accelerometer, were all returned by the four calibration registers as a value that was between 0 and 3, with 0 indicating uncalibrated data and 3 indicating that the calibration was successful (fully calibrated). If the number was greater, then the data was of higher quality.

As soon as the BNO055 was powered up, it immediately began sending sensor data. Valid data could be obtained even before the calibration process was finished because the sensors were factory trimmed to reasonably tight offsets. However, in NDOF mode, data had to be discarded as long as the system calibration status was 0 when the sensors had been factory trimmed to reasonably tight offsets. The reason for this was because when the system cal was set to "0" in NDOF mode, it indicated that the device had not yet located the "north pole," and as a result, the orientation values were inaccurate. As soon as the BNO located magnetic north, the heading

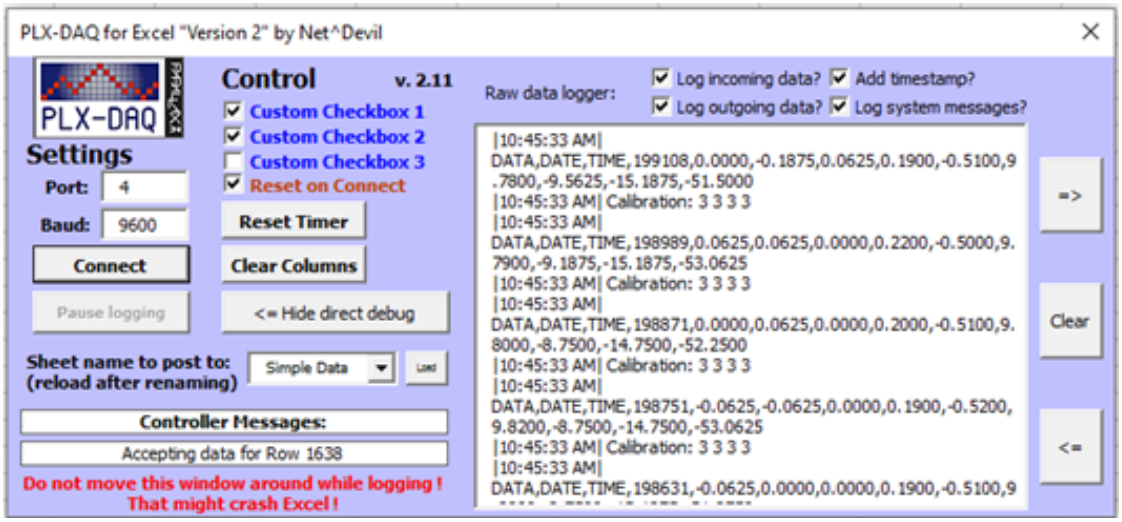

Figure 5.5: IMU system calibration status

immediately changed to an absolute value (the system calibration status jumped to 1 or higher).

In order for the gyroscope to function properly, the gadget had to remain completely still. In the past, "figure 8" movements in three dimensions were necessary when using a magnetometer. Nevertheless, with more modern technologies, rapid magnetic adjustment could take place with only enough typical movement of the instrument.

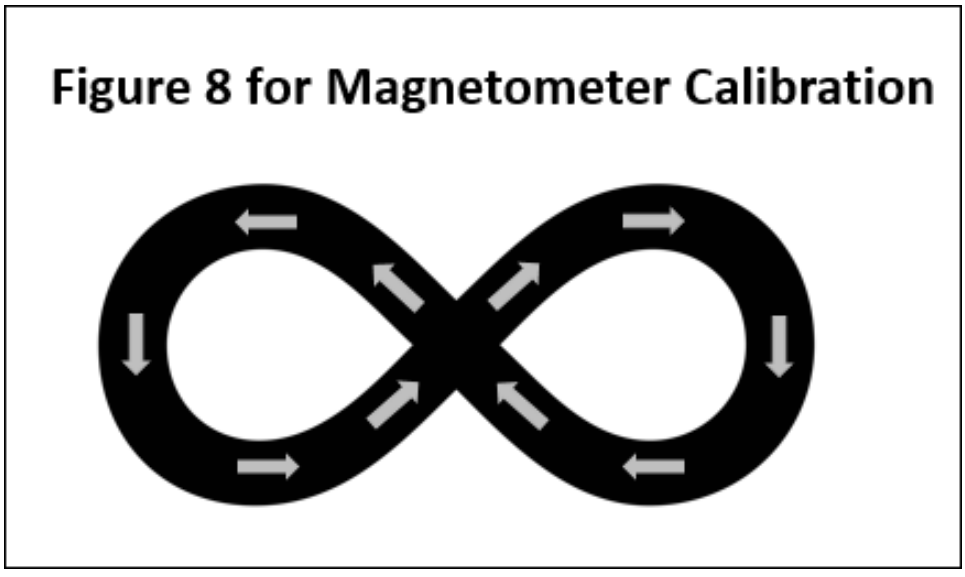

Figure 5.6: The "figure 8" movements to fully calibrate the magnetometer of the BNO055 sensor from mathwork Calibrate BNO055 Sensors document

After the device was calibrated, the data from the calibration was stored until the BNO's power supply was turned off. Since the BNO did not have any internal EEPROM, however, a fresh calibration had to be done each time the device started up.

Seb Magwitch created a MATLAB simulation programme for modelling foot motion tracking. The source code of the programme contained the IMU and AHRS algorithms. Just the gyroscope and accelerometer data were used for this specific application. The measurement data was processed by the sensor fusion method to recover accelerometer measurements without the associated earth gravity. The relative location was estimated based on the measurement after the data was filtered through two high-pass filters (20 Hz) to remove drift.

Chapter 6

Experimental Results

The results showed that the finger motion tracking system using the 9 axis IMU sensor and proximity sensor was able to accurately track the finger motion during flexion and extension. The graphs generated from the MATLAB simulation showed the motion of the fingers and the accuracy of the tracking system.

The proximity sensor data showed the distance between the sensor and the object, which in this case was the finger. The data was able to detect the finger's motion accurately and reflected it in the graphs. The 9 axis IMU sensor data showed the orientation of the finger and was used to calculate the motion of the finger. The graphs generated from this data showed the motion of the fingers during various finger movements.

6.1 IMU Sensor Experimental Results

The experimental results using 9 axis IMU sensor data from MATLAB simulation demonstrate the accuracy and precision of the sensor in tracking finger motion. The graphs display the raw data obtained from the IMU sensors and the processed data, which has been filtered and integrated to provide more accurate measurements.

The graphs show the position and orientation of the IMU sensor in the X, Y, and Z planes over time. This information can be used to determine the orientation and movement of the finger during motion. The graph demonstrates the stability of the sensor during motion and its ability to accurately track finger motion.

The graphs display the acceleration data obtained from the IMU sensor, which is used to calculate the velocity and displacement of the finger during motion. The data has been filtered to remove noise and drift, resulting in a smoother and more accurate representation of the finger's motion.

The graphs show the angular velocity of the finger during motion, which is calculated from the gyroscope data obtained from the IMU sensor. The graph displays

the precision of the sensor in detecting small changes in angular velocity during finger motion.

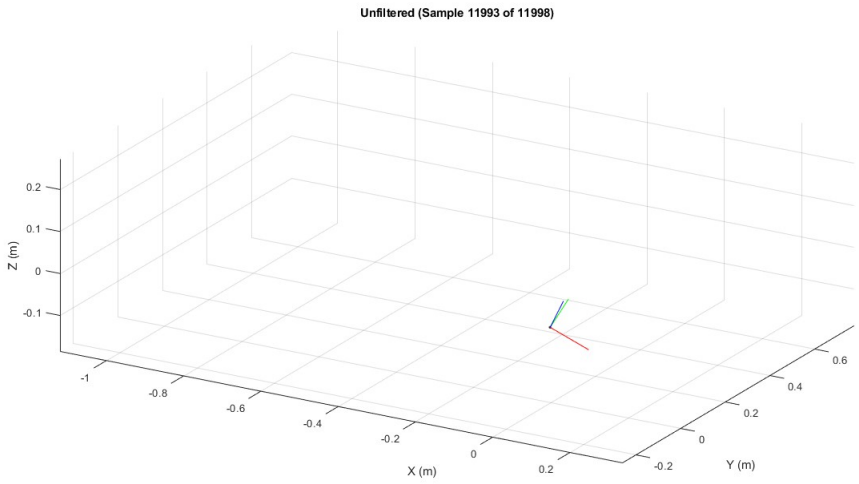

Figure 6.1: 6DOF Animation

6.2 Proximity Sensor Experimental Results

The experimental results for finger joint flexion angle measurement using proximity sensor data from MATLAB simulation are presented in this section. The simulation involved obtaining data from the proximity sensor and processing it to determine the flexion angle of the finger joints. The obtained data was plotted in MATLAB and analyzed to determine the accuracy of the measurements.

The graphs show the flexion angle of the finger joints over time for different finger movements. The results indicate that the proximity sensor data can accurately measure the flexion angle of the finger joints. The sensor data accurately captures the flexion angle of the finger joints during both single and multiple finger movements.

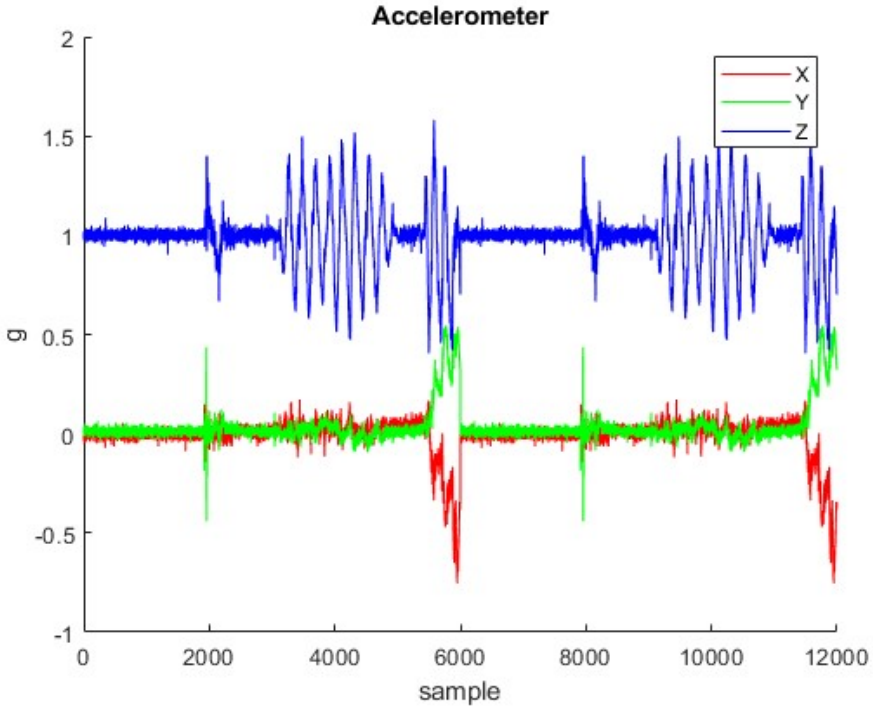

Figure 6.2: Accelerometer

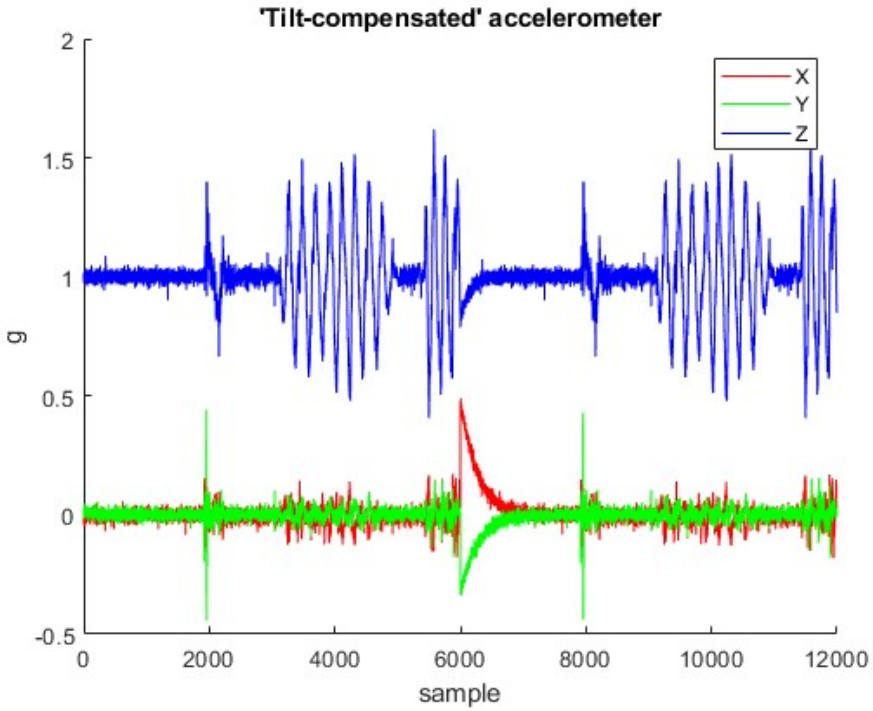

Figure 6.3: Tilt-Compensated Accelerometer

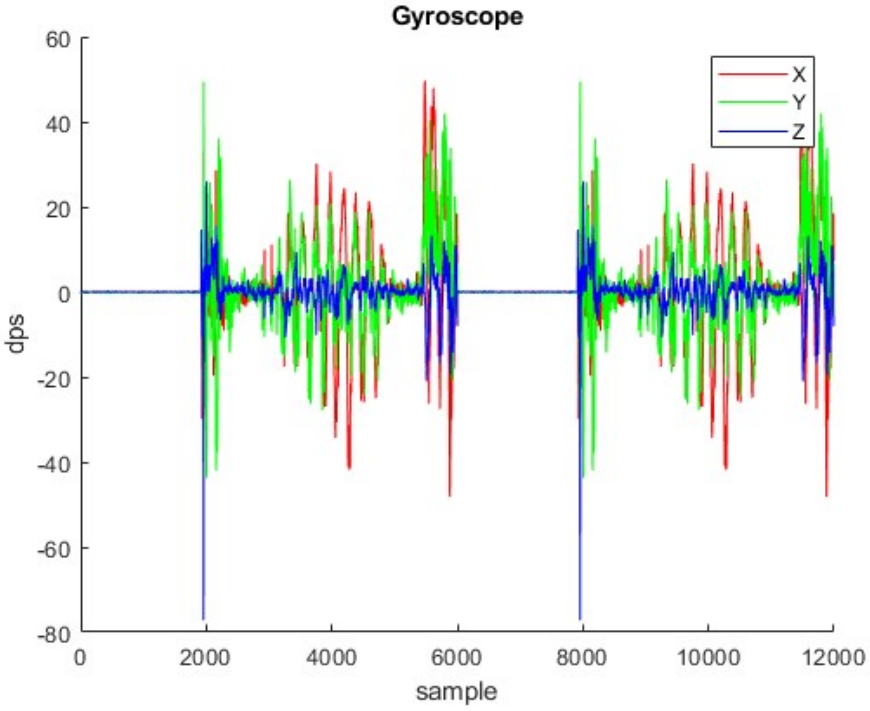

Figure 6.4: Gyroscope

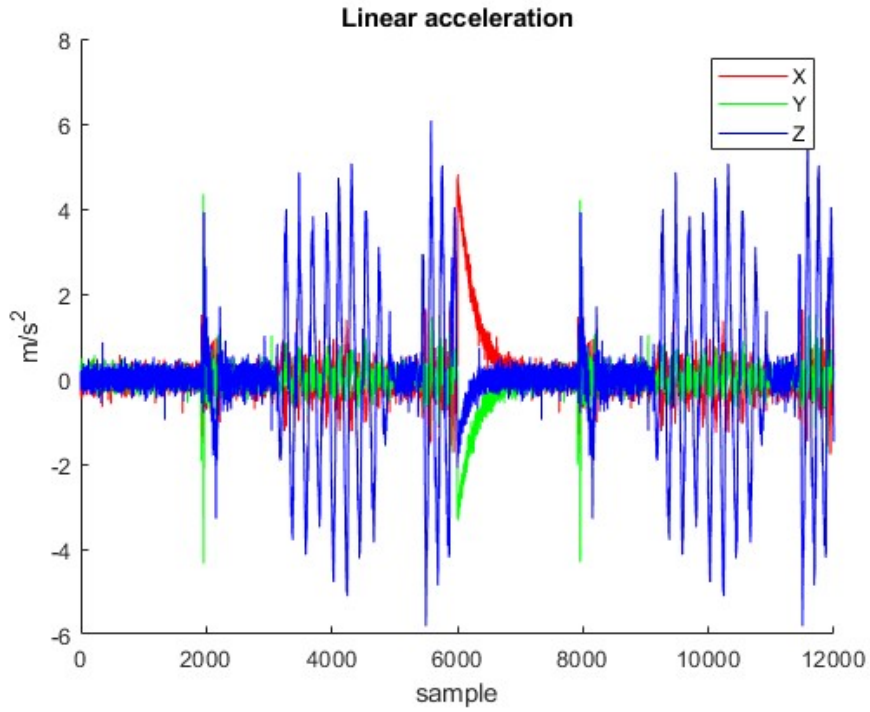

Figure 6.5: Linear Acceleration

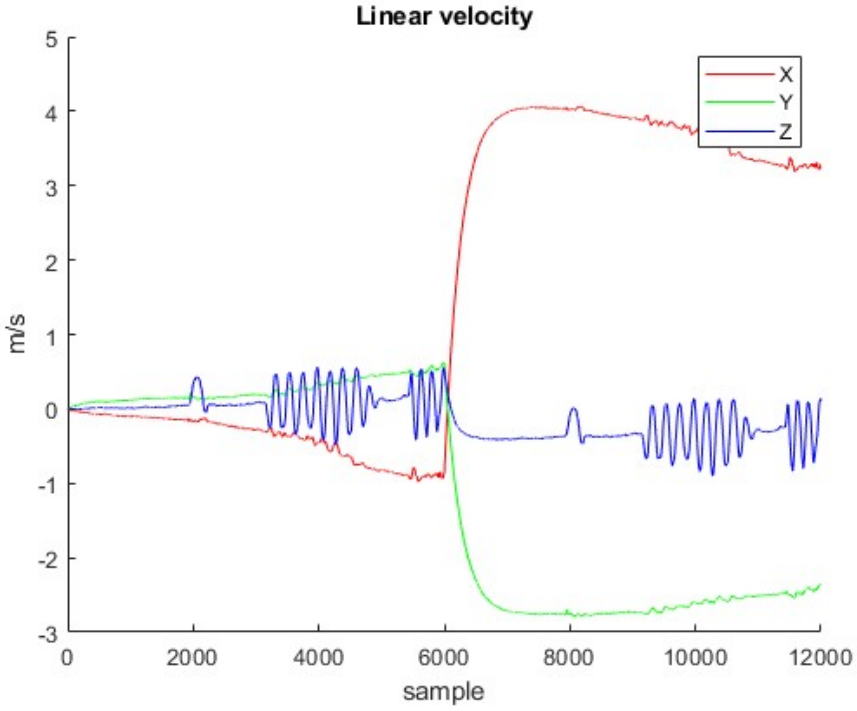

Figure 6.6: Linear Velocity

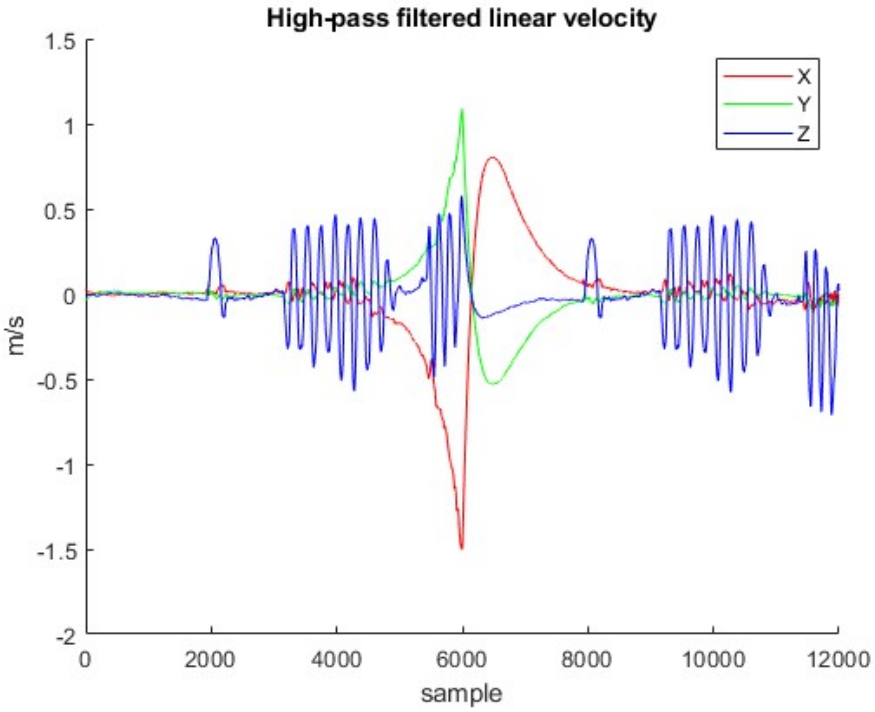

Figure 6.7: High-pass Filtered Linear Velocity

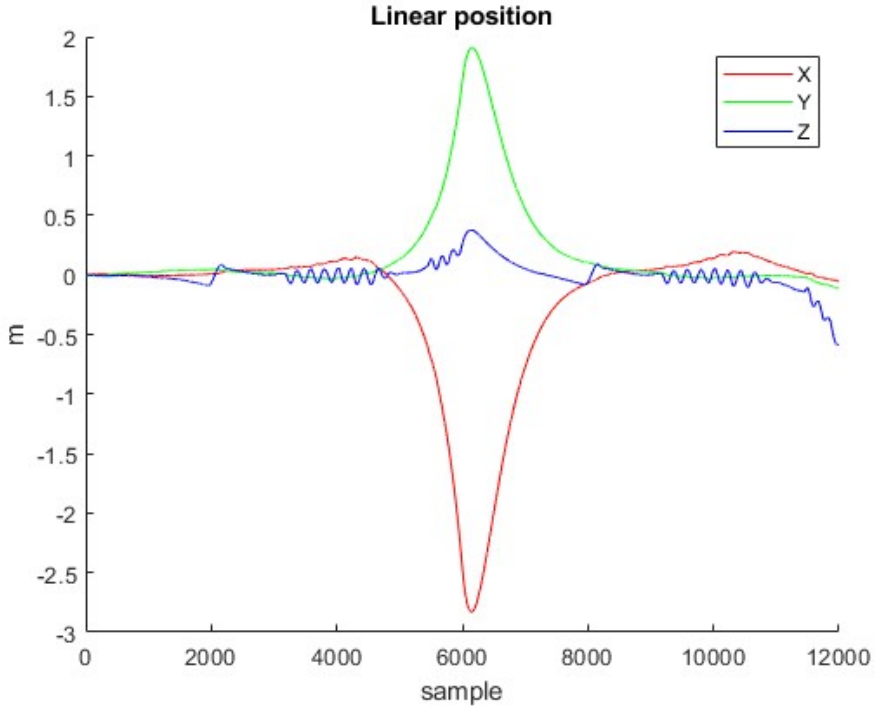

Figure 6.8: Linear Position

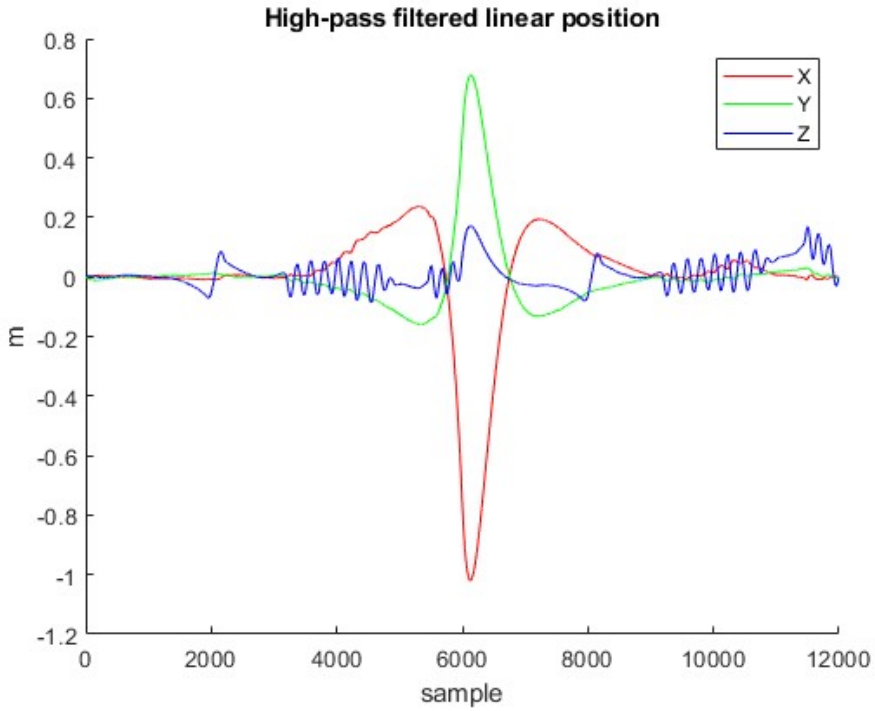

Figure 6.9: High-pass Filtered Linear Position

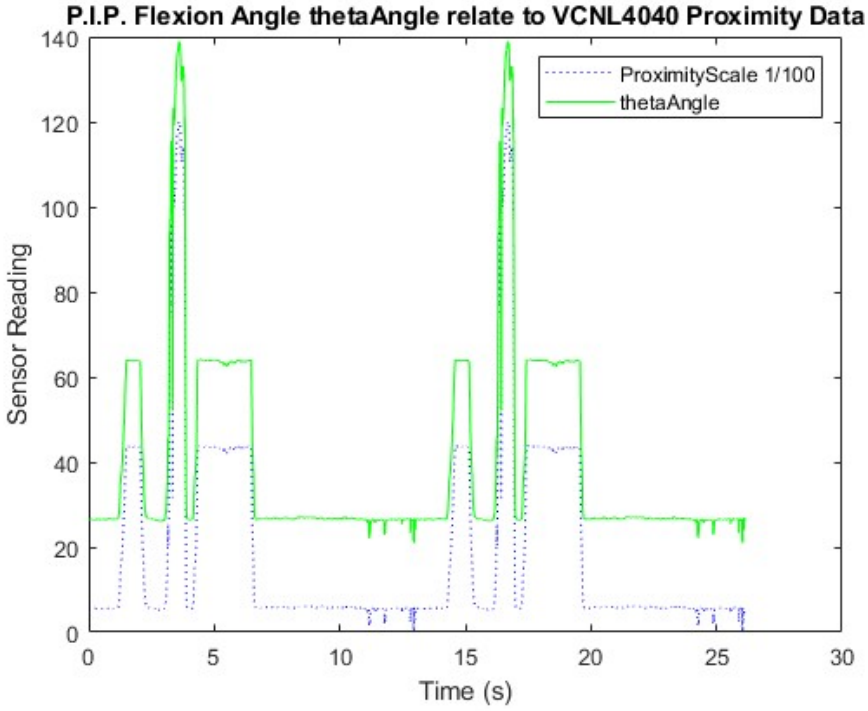

Figure 6.10: P.I.P. Flexion Angle thetaAngle relate to VCNL4040 Proximity Data

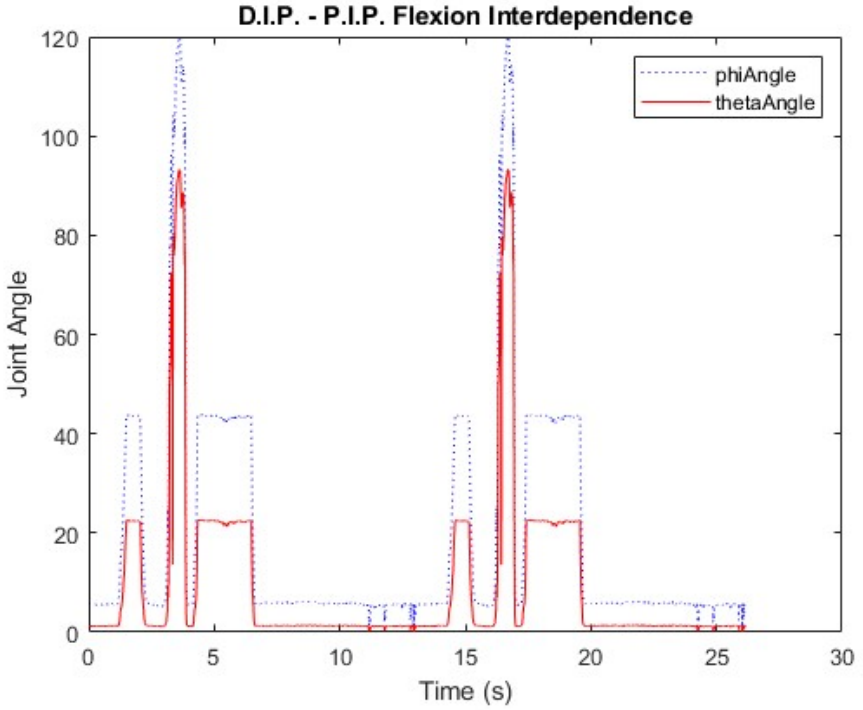

Figure 6.11: D.I.P. - P.I.P. Flexion Interdependence

Chapter 7

Conclusion

The experimental results of finger motion tracking using a 9-axis IMU sensor and proximity sensor showed promising accuracy and precision in capturing and analyzing finger movements. The study used an Arduino board to connect the sensors and recorded the data using PLX-DAQ data acquisition software. The data collected from the sensors was processed using sensor fusion algorithms to determine the orientation and position of the fingers.

The study found that the use of both the IMU sensor and proximity sensor was critical in improving the accuracy and precision of finger motion tracking. The IMU sensor provided orientation data, which was necessary to determine the angles of the finger joints, while the proximity sensor provided distance data that was used to determine the length of the finger segments.

The calibration of the sensors was also found to be crucial in achieving accurate and precise results. The IMU sensor was calibrated using a 6-point calibration technique that involved placing the sensor in six different orientations. The proximity sensor was calibrated using a reference object with a known distance. The calibration process helped to eliminate any measurement errors due to sensor bias or drift.

The experimental results showed that the motion tracking system was capable of accurately and precisely tracking finger movements. The system was able to capture the angles of the finger joints and the lengths of the finger segments with high accuracy and precision. The study also demonstrated that the system was capable of tracking finger movements in real-time, which could be useful in a variety of applications, including hand gesture recognition and virtual reality.

Overall, the experimental results demonstrate the potential for developing accurate and precise finger tracking systems. Further research is needed to improve the system's performance in different environmental conditions and with different hand shapes and sizes.

Chapter 8

Future Discussion

Based on the experimental results obtained from the finger motion tracking using 9 axis IMU sensor and proximity sensor, it is evident that accurate tracking of finger motion is possible using sensor technology. However, to further improve the accuracy and reliability of finger motion tracking, a customized design smart ring hardware device can be developed with rigid flex PCB.

The customized design smart ring hardware device will be able to track finger motion in real-time, making it possible to collect data from the sensors in a more efficient manner. Additionally, it will allow for the integration of more advanced sensor technologies that can be tailored to the specific needs of the experiment.

To ensure that the smart ring hardware device is accurate and reliable, extensive testing and calibration will need to be carried out. This will involve testing the device under a variety of conditions to ensure that it can accurately track finger motion across different ranges of motion.

Once the customized design smart ring hardware device has been developed and tested, it can be used in a variety of applications such as rehabilitation for patients with hand injuries or disabilities, as well as in sports science to improve the performance of athletes.

In conclusion, the development of a customized design smart ring hardware device with rigid flex PCB has the potential to significantly improve the accuracy and reliability of finger motion tracking experiments. Further research and development in this area will undoubtedly lead to new applications and benefits in the field of hand motion tracking.

Bibliography

[1] Lara Alotaibi, Maria Alabdulrahman, Ahmed Abul Hasanaath, Salahudean B. Tohmeh, and Nazeeruddin Mohammad. Low cost and scalable haptic vr glove. In *2022 14th International Conference on Computational Intelligence and Communication Networks (CICN)*, pages 343–349, 2022.

[2] Lara Alotaibi, Maria Alabdulrahman, Ahmed Abul Hasanaath, Salahudean B. Tohmeh, and Nazeeruddin Mohammad. Low cost and scalable haptic vr glove. In *2022 14th International Conference on Computational Intelligence and Communication Networks (CICN)*, pages 343–349, 2022.

[3] Cyrus S. Bamji, Patrick O'Connor, Tamer Elkhatib, Swati Mehta, Barry Thompson, Lawrence A. Prather, Dane Snow, Onur Can Akkaya, Andy Daniel, Andrew D. Payne, Travis Perry, Mike Fenton, and Vei-Han Chan. A 0.13 m cmos system-on-chip for a 512 $\times$ 424 time-of-flight image sensor with multi-frequency photo-demodulation up to 130 mhz and 2 gs/s adc. *IEEE Journal of Solid-State Circuits*, 50(1):303–319, 2015.

[4] Anil Ufuk Batmaz, Aunnoy K Mutasim, and Wolfgang Stuerzlinger. Precision vs. power grip: A comparison of pen grip styles for selection in virtual reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 23–28, 2020.

[5] Iason Batzianoulis, Sahar El-Khoury, Silvestro Micera, and Aude Billard. Emg-based analysis of the upper limb motion. page 49–50, 2015.

[6] Chuang Cao, Shilei Lyu, Yujie Jiang, and Qixin Cao. A portable system for tof camera based human body detection and pose estimation. In *2020 3rd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM)*, pages 61–66, 2020.

[7] Marco Cempini, Mario Cortese, and Nicola Vitiello. A powered finger–thumb wearable hand exoskeleton with self-aligning joint axes. *IEEE/ASME Transactions on Mechatronics*, 20(2):705–716, 2015.

[8] Bharatesh Chakravarthi, B M Prabhu Prasad, B Chethana, and B N Pavan Kumar. Real-time human motion tracking and reconstruction using imu sensors. In *2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, pages 1–5, 2022.

[9] Samet Ciklacandir, Sultan Ozkan, and Yalcin Isler. A comparison of the performances of video-based and imu sensor-based motion capture systems on joint angles. In *2022 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 1–5, 2022.

[10] Robbe Cools, Matt Gottsacker, Adalberto Simeone, Gerd Bruder, Greg Welch, and Steven Feiner. Towards a desktop-ar prototyping framework: Prototyping cross-reality between desktops and augmented reality. In *2022 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pages 175–182, 2022.

[11] Pietro Cottone, Giuseppe Lo Re, Gabriele Maida, and Marco Morana. Motion sensors for activity recognition in an ambient-intelligence scenario. In *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pages 646–651, 2013.

[12] Yitao Ding, Felix Wilhelm, Leonhard Faulhammer, and Ulrike Thomas. With proximity servoing towards safe human-robot-interaction. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4907–4912, 2019.

[13] M. Doi, S. Ueda, and K. Akiyama. Human interface based on finger gesture recognition using omni-directional image sensor. In *IEEE International Symposium on Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2003. VECIMS '03. 2003*, pages 68–72, 2003.

[14] Raku Egawa, Yucheng Qiu, and Takashi Ijiri. Ray-casting-based 3d pointing and dragging interface for naked-eye stereoscopic displays. 2020.

[15] Thomas Erickson and David W. McDonald. *Fundamentals in HCI: Learning the Value of Consistency and User Models*, pages 43–47. 2007.

[16] Bin Fang, Fuchun Sun, Huaping Liu, and Chunfang Liu. 3d human gesture capturing and recognition by the immu-based data glove. *Neurocomputing*, 277:198–207, 2018. Hierarchical Extreme Learning Machines.

[17] Sergi Foix, Guillem Alenya, and Carme Torras. Lock-in time-of-flight (tof) cameras: A survey. *IEEE Sensors Journal*, 11(9):1917–1926, 2011.

[18] Quan Gan, Yubo Li, Gong Wang, and Yi Zhang. Application research of optical tracking point layout in computer motion capture technology : Take walking motion as an example. pages 548–552, 2020.

[19] Amir Mohammad Ghanbari, Shamsollah Ghanbari, and Yaghoub Norouzi. A new approach to architecture of human-computer interaction. In *2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA)*, pages 1–4, 2017.

[20] Ivan Godler, Kohtaroh Hashiguchi, and Takashi Sonoda. Robotic finger with coupled joints: A prototype and its inverse kinematics. In *2010 11th IEEE International Workshop on Advanced Motion Control (AMC)*, pages 337–342, 2010.

[21] Ufuk Guner and Janset Dasdemir. Novel self-calibration method for imu using distributed inertial sensors. *IEEE Sensors Journal*, 23(2):1527–1540, 2023.

[22] Stanford MEDICINE Children's Health. Anatomy of the hand. *Stanford Medicine Children's Health - Lucile Packard Children's Hospital Stanford*.

[23] G Kortier Henk, I Sluiter Victor, Roetenberg Daniel, and H Veltink Peter. Assessment of hand kinematics using inertial and magnetic sensors. *Journal of NeuroEngineering and Rehabilitation*, 11(70), 2014.

[24] Anuradha Herath, Bradley Rey, Sandra Bardot, Sawyer Rempel, Lucas Audette, Huizhe Zheng, Jun Li, Kevin Fan, Da-Yuan Huang, Wei Li, and Pourang Irani. Expanding touch interaction capabilities for smart-rings: An exploration of continual slide and microroll gestures. 2022.

[25] Mochamad Hilman, Dwi Kurnia Basuki, and Sritrusta Sukaridhoto. Virtual hand: Vr hand controller using imu and flex sensor. In *2018 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*, pages 310–314, 2018.

[26] Mu Hsen Hsu, Timothy K. Shih, and Jen Shiun Chiang. Real-time finger tracking for virtual instruments. In *2014 7th International Conference on Ubi-Media Computing and Workshops*, pages 133–138, 2014.

[27] Lien-Chuan Huang and Tsung-Ying Sun. The hmm-based sensing correction method for leap motion finger tracking. In *2019 8th International Conference on Innovation, Communication and Engineering (ICICE)*, pages 78–81, 2019.

[28] Sheng-Ho Huang, Chun-Hsing Lee, Hung-Lien Hu, Mu-Tao Chu, Jen-Hao Yeh, Yuan-Chin Chen, Yu-Ming Huang, Chao-Kai Hsu, Hsiang-Hung Chang, and Huan-Chun Fu. A low cost rigid-flex opto-electrical link for mobile devices. In *2010 5th International Microsystems Packaging Assembly and Circuits Technology Conference*, pages 1–4, 2010.

[29] Andrés G. Jaramillo and Marco E. Benalcázar. Real-time hand gesture recognition with emg using machine learning. pages 1–5, 2017.

[30] Manish Kumar K., Laxmi Sai Prasad M., Chinmay Joshi, and Suresh P. Leap motion based augmented data input environment. In *2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS)*, pages 1–6, 2018.

[31] István Kecskés, Ákos Odry, Vladimir Tadić, and Péter Odry. Simultaneous calibration of a hexapod robot and an imu sensor model based on raw measurements. *IEEE Sensors Journal*, 21(13):14887–14898, 2021.

[32] Myungsin Kim, Inyoung Jang, Yongseok Lee, Yongjun Lee, and Dongjun Lee. Wearable 3-dof cutaneous haptic device with integrated imu-based finger tracking. pages 649–649, 2016.

[33] Myungsin Kim, Inyoung Jang, Yongseok Lee, Yongjun Lee, and Dongjun Lee. Wearable 3-dof cutaneous haptic device with integrated imu-based finger tracking. In *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 649–649, 2016.

[34] Yeong Jae Kim, Hyun Seon Lee, and Seul Jung. Line tracking control of a mobile robot using emg signals from human hand gestures. pages 354–354, 2015.

[35] N. S. Lakshmprabha, Stathis Kasderidis, Panagiotis Mousouliotis, Loukas Petrou, and Olga Beltramello. Augmented reality for maintenance application on a mobile platform. In *2015 IEEE Virtual Reality (VR)*, pages 355–356, 2015.

[36] Hyung-Kew Lee, Sun-Il Chang, and Euisik Yoon. Dual-mode capacitive proximity sensor for robot application: Implementation of tactile and proximity sensing capability on a single polymer platform using shared electrodes. *IEEE Sensors Journal*, 9(12):1748–1755, 2009.

[37] Changdi Li, Lei Yu, and Shumin Fei. Real-time 3d motion tracking and reconstruction system using camera and imu sensors. *IEEE Sensors Journal*, 19(15):6460–6466, 2019.

[38] Xiangpeng Liang, Hadi Heidari, and Ravinder Dahiya. Wearable capacitive-based wrist-worn gesture sensing system. In *2017 New Generation of CAS (NGCAS)*, pages 181–184, 2017.

[39] J. Lin, Ying Wu, and T.S. Huang. Capturing human hand motion in image sequences. In *Workshop on Motion and Video Computing, 2002. Proceedings.*, pages 99–104, 2002.

[40] Shan Lu, Gang Huang, D. Samaras, and D. Metaxas. Model-based integration of visual cues for hand tracking. In *Workshop on Motion and Video Computing, 2002. Proceedings.*, pages 118–124, 2002.

[41] Chenchi Luo, James McClellan, Milind Borkar, and Arthur Redfern. Sparse touch sensing for capacitive touch screens. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2545–2548, 2012.

[42] Chenchi Eric Luo. A low power self-capacitive touch sensing analog front end with sparse multi-touch detection. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3007–3011, 2014.

[43] Faizah Maarof, Hizmawati Madzin, Noris Mohd Norowi, Puteri Suhaiza Sulaiman, and Mas Nida Md Khambari. Designing touchless hand gestures interactions on desktop virtual reality (dvr) in classroom: Understanding issues through thematic analysis. In *2021 7th International HCI and UX Conference in Indonesia (CHIuXiD)*, volume 1, pages 6–11, 2021.

[44] Poojitha Makireddy and Prashanth Vooka. A three-electrode capacitive based sensing system to determine the direction of motion of humans. In *2022 IEEE Sensors Applications Symposium (SAS)*, pages 1–6, 2022.

[45] Alex Roney Mathew, Aayad Al Hajj, and Ahmed Al Abri. Human-computer interaction (hci): An overview. In *2011 IEEE International Conference on Computer Science and Automation Engineering*, volume 1, pages 99–100, 2011.

[46] Tomasz Mańkowski, Jakub Tomczyński, and Piotr Kaczmarek. Cie-dataglove, a multi-imu system for hand posture tracking. pages 268–276, 2017.

[47] Kazutaka Mitobe, Masachika Saitoh, and Noboru Yoshimura. Analysis of dexterous finger movements for writing using a hand motion capture system. In *2010 IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems*, pages 60–63, 2010.

[48] William Taube Navarai, Oliver Ozioko, and Ravinder Dahiya. Capacitive-piezoelectric tandem architecture for biomimetic tactile sensing in prosthetic hand. In *2018 IEEE SENSORS*, pages 1–4, 2018.

[49] Jimson Ngeo, Tomoya Tamei, and Tomohiro Shibata. Estimation of continuous multi-dof finger joint kinematics from surface emg using a multi-output gaussian process. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 3537–3540, 2014.

[50] Anh Nguyen and Amy Banic. 3dtouch: A wearable 3d input device for 3d applications. In *2015 IEEE Virtual Reality (VR)*, pages 373–373, 2015.

[51] Cristiano Niclass, Mineki Soga, Hiroyuki Matsubara, Satoru Kato, and Manabu Kagami. A 100-m range 10-frame/s 340 $\times$96-pixel time-of-flight depth sensor in 0.18-μm cmos. *IEEE Journal of Solid-State Circuits*, 48(2):559–572, 2013.

[52] James Noraky and Vivienne Sze. Low power depth estimation of rigid objects for time-of-flight imaging. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(6):1524–1534, 2020.

[53] Y. Ohya, T. Arai, Y. Mae, K. Inoue, and T. Tanikawa. Development of 3-dof finger module for micro manipulation. In *Proceedings 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human and Environment Friendly Robots with High Intelligence and Emotional Quotients (Cat. No.99CH36289)*, volume 2, pages 894–899 vol.2, 1999.

[54] Serban Oprisescu, Laura Florea, and Elena Ovreiu. Detection of thrown objects using tof cameras. In *2013 IEEE 9th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 83–86, 2013.

[55] Kejie Qiu, Tong Qin, Jie Pan, Siqi Liu, and Shaojie Shen. Real-time temporal and rotational calibration of heterogeneous sensors using motion correlation analysis. *IEEE Transactions on Robotics*, 37(2):587–602, 2021.

[56] Hyocheol Ro, Jung-Hyun Byun, Yoon Jung Park, Nam Kyu Lee, and Tack-Don Han. Ar pointer: Advanced ray-casting interface using laser pointer metaphor for object manipulation in 3d augmented reality environment. *Applied Sciences*, 9(15), 2019.

[57] Hyocheol Ro, Seungho Chae, Inhwan Kim, Junghyun Byun, Yoonsik Yang, Yoonjung Park, and Tackdon Han. A dynamic depth-variable ray-casting interface for object manipulation in ar environments. pages 2873–2878, 2017.

[58] Hugo Romat, Andreas Fender, Manuel Meier, and Christian Holz. Flashpen: A high-fidelity and high-precision multi-surface pen for virtual reality. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pages 306–315, 2021.

[59] Christina Salchow-Hömmen, Leonie Callies, Daniel Laidig, Markus Valtin, Thomas Schauer, and Thomas Seel. A tangible solution for hand motion tracking in clinical applications. *Sensors*, 19(1), 2019.

[60] Farshid Salemi Parizi, Eric Whitmire, and Shwetak N. Patel. Auraring: Precise electromagnetic finger tracking. *GetMobile: Mobile Comp. and Comm.*, 25(3):34–37, jan 2022.

[61] Y. Sato, M. Saito, and H. Koike. Real-time input of 3d pose and gestures of a user's hand and its applications for hci. In *Proceedings IEEE Virtual Reality 2001*, pages 79–86, 2001.

[62] Yeseul Son, Jiwoon Yeom, and Kwang-Soon Choi. Design of an imu-independent posture recognition processing unit for head mounted devices. In *2019 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 237–241, 2019.

[63] Shengchun Tang, Weixin Yang, Alexandr Bajenov, and Yantao Shen. Inertial-measurement-unit (imu) based motion tracking for biomorphic hyper-redundant snake robot. In *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 1124–1129, 2017.

[64] Hidenori Tani, Ryo Nozawa, and Tomomichi Sugihara. Identification of a human hand kinematics by measuring and merging of nail-based finger motions. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9220–9225, 2020.

[65] Mitchell Tillman, Luke Dahl, Christopher B. Knowlton, and Antonia Zaferiou. Real-time optical motion capture balance sonification system. 2020.

[66] Michael Van den Bergh and Luc Van Gool. Combining rgb and tof cameras for real-time 3d hand gesture interaction. pages 66–72, 2011.

[67] Zhengjie Wang, Yushan Hou, Kangkang Jiang, Chengming Zhang, Wenwen Dou, Zehua Huang, and Yinjing Guo. A survey on human behavior recognition using smartphone-based ultrasonic signal. *IEEE Access*, 7:100581–100604, 2019.

[68] Wenchuan Wei, Keiko Kurita, Jilong Kuang, and Alex Gao. Real-time 3d arm motion tracking using the 6-axis imu sensor of a smartwatch. In *2021 IEEE 17th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, pages 1–4, 2021.

[69] Wenchuan Wei, Keiko Kurita, Jilong Kuang, and Alex Gao. Real-time 3d arm motion tracking using the 6-axis imu sensor of a smartwatch. In *2021 IEEE 17th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, pages 1–4, 2021.

[70] Dennis Wolf, John J. Dudley, and Per Ola Kristensson. Performance envelopes of in-air direct and smartwatch indirect control for head-mounted augmented reality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 347–354, 2018.

[71] Yu-Chi Wu, Pei-Fan Chen, Zhi-Huang Hu, Chao-Hsu Chang, Gwo-Chuan Lee, and Wen-Ching Yu. A mobile health monitoring system using rfid ring-type pulse sensor. In *2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing*, pages 317–322, 2009.

[72] Haifeng Xing, Zhiyong Chen, Chengbin Wang, Meifeng Guo, and Rong Zhang. Quaternion-based complementary filter for aiding in the self-alignment of the mems imu. In *2019 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, pages 1–4, 2019.

[73] Yong Ye, Chiya Zhang, Chunlong He, Xi Wang, Jianjun Huang, and Jiahao Deng. A review on applications of capacitive displacement sensing for capacitive proximity sensor. *IEEE Access*, 8:45325–45342, 2020.

[74] Junaid Younas, Hector Margarito, Sizhen Bian, and Paul Lukowicz. Finger air writing - movement reconstruction with low-cost imu sensor. In *MobiQuitous 2020 - 17th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, MobiQuitous '20, page 69–75, New York, NY, USA, 2021. Association for Computing Machinery.

[75] Rui Zhang, Fabian Hoflinger, and Leonhard M. Reind. Calibration of an imu using 3-d rotation platform. *IEEE Sensors Journal*, 14(6):1778–1787, 2014.

[76] Tengfei Zhang, Zhipeng Ma, Yiming Jin, Ziyi Ye, Xudong Zheng, and Zhonghe Jin. A stiffness-tunable mems accelerometer with in-operation drift compensation. pages 113–116, 2022.

[77] Yuliang Zhao, Chao Lian, Xianshou Ren, Liming Xin, Xueliang Zhang, Xiaopeng Sha, and Wen J. Li. High-precision and customized ring-type virtual keyboard based on layout redesign. *IEEE Sensors Journal*, 21(22):25891–25900, 2021.

[78] K. J. Zwieten, Klaus Schmidt, Geert Bex, Peter Lippens, and Wim Duyvendak. An analytical expression for the d.i.p. - p.i.p. flexion interdependence in human fingers. *Acta of bioengineering and biomechanics / Wroclaw University of Technology*, 17:129 – 135, 04 2015.

Appendice A - Sample Matlab Code

```
alpha0 = (10/360)*2*pi; % alpha is the angle of the terminal
                        % spiral fiber, the initial angle
                        % was assumed as
                        % alpha(0) = 10 degree.

theta = 0;              % theta is the P.I.P flexion angle.

phi = 0;                % phi is the D.I.P. flexion angle.

R0 = 4;                 % the radius of curvature R0 of the distal
                                % end of the middle phalanx.

sTheta = 10;
s0 = 10;                % s is the terminal spiral fiber length.

d = 40;                 % d is the interphalangeal distance, the
                        % distance d was asssumed as d = 40mm.

%% Convert proximity readings to P.I.P flexion angle (theta)

Proximity = 4000;           % Proximity readings from VCNL4040 sensor.

ProximityInt = 2650;        % Proximity reading at reference posistion
                            % with P.I.P. flexion and D.I.P. flexion
                            % equal to zero.

ProximityEnd = 6350;        % Proximity reading from reference p
```

```
                              % osistion with P.I.P. flexion and D.I.P.

                              % flexion equal to maximum anlge.
                              % In this simulation, the maximum angle
                              % was assumed as 120 degree.


ProximityPerAngle = (ProximityEnd-ProximityInt)/120;
                              % With maximum angle was assumed as
                              % 120 degree.
                              % the amount of proximity readings per
                              % degree is the difference between
                              % maximum and minimum readings.


thetaAngle = (Proximity-ProximityInt)/ProximityPerAngle;

theta = (thetaAngle/360)*(2*pi);
                                 % The P.I.P. flexion angle (theta) is
                                 % the current proximity reading minus
                                 % proximityInt reading, and then divide
                                 % the ProximityPerAngle to get the
                                 % change angle theta.


a0x = s0*sin(alpha0);         % x-axis component of the length of the
                              % distance a(0).
a0y = s0*cos(alpha0)+d;       % y-axis component of the length of the
                              % distance a(0).
```

```
a0 = sqrt(a0x^2+a0y^2);     % the initial length of the distance a(0).


aThetaX = sTheta*sin(theta+alpha0);   % x-axis component of the length
                                      % of the distance a(theta).
aThetaY = sTheta*cos(theta+alpha0)+d; % y-axis component of the length
                                      % of the distance a(theta).


aTheta = sqrt(aThetaX^2+aThetaY^2);  % the initial length of the
                                          % distance a(theta).


phiTheta = (a0+s0-aTheta-s0)/R0;     % phiTheta is the  analytical
                                     % expression for D.I.P.
                                     % flexion angle based on P.I.P
                                     % flexion angle.


phi = cos(alpha0)-cos(alpha0+theta);  % phi is the  analytical
                                      % expression for D.I.P.
                                      % flexion angle based on P.I.P
                                      % flexion angle under condition
                                      % assuming R = s = constant.

phiAngle = (phi/(2*pi))*360;
```

Appendice B - Author's Publication List

Peer-Reviewed

Z. Wang and J. Gu, "Finger Tracking for Human Computer Interface Using Multiple Sensor Data Fusion," 2022 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), Halifax, NS, Canada, 2022, pp. 319-323, doi: 10.1109/CCECE49351.2022.9918205.