UWB and IMU Fusion Based on Kalman Filter in Mobile Robot Localization System

By

Su Liu

A Thesis

Submitted to the School of Graduate Studies

In Partial Fulfillment of the Requirements

For the Degree of

Master of Applied Science

McMaster University

Master of Applied Science (2023), Mechanical Engineering, McMaster University, Hamilton, Ontario Canada.

Title: UWB and IMU Fusion Based on Kalman Filter in Mobile Robot Localization System

Author: Su Liu

Supervisor: Dr. Fengjun Yan

Abstract

Nowadays, mobile robots are used in a wide variety of fields, such as manufacturing, agriculture, space and underwater exploration, and healthcare. The localization process is crucial to mobile robots and refers mainly to the precise determination of the coordinates where the system is present at a certain moment. Ultra-wideband (UWB) and inertial measurement unit (IMU) are commonly used techniques in localization systems. However, UWB can't avoid Non-line-of-sight (NLOS) propagation errors due to obstacles and IMU always suffers from error accumulation over time. In my research, IMU and UWB are combined optimally in the Kalman filter to overcome their limitations in specific situations. A new method to detect UWB measurement anomaly is proposed based on the difference between the velocity calculated by UWB measurement and IMU data. The complementary filter is used to combine the accelerometer data and gyroscope data to derive the roll and pitch degree. Kalman filter parameters are adjusted in different situations to help the localization system perform better and provide reliable position information to the control system to complete the tracking task.

Keywords: UWB, IMU, NLOS, Kalman Filter, localization

Table of Contents

1. Introduction

1.1 Background

1.1.1 Mobile Robots Review

The mobile robot is defined as an automatic machine highly controlled by software programming that uses sensors and other technologies to move around its environment. Mobile robots are used in a wide variety of fields, such as manufacturing, agriculture, space and underwater exploration, and healthcare. Mobile robots include five main parts: control system, sensors, actuator, power supply, and end effector. The robot control system manages, commands, and directs the movement and functions of various parts of the mobile robot to achieve desired results. The working principle of a robot actuator is to convert energy into physical motion. Robotic actuators typically fall into two categories: linear motion and rotational motion. For the power supply, electrical power is the most common source of energy in mobile robotics. The elements used in mobile robotics to help regenerate power constantly and slow down the depletion of energy are rechargeable batteries, such as Lead-Acid (Pb-Acid) Battery and Nickel-Iron (Ni-Fe) Battery and solar power systems collectors including flat plate collectors, focusing collectors, and passive collectors. Sensors are essential parts of a mobile robot, without which it could not gather information about the environment and itself. Sensors can be generally divided into those for perceiving external information helping robots establish their position, navigate, and avoid

obstacles, and those for perceiving internal state such as battery level, wheel position, degree of tilt, and so on. Based on the types of measurements performed the sensors can be classified into four categories: distance sensors, positioning sensors, environment sensors, and inertial sensors.

The important function of a sensor network is to collect and forward data to the destination. The position information is extraordinarily important among various data and accurate position estimation is a key component for autonomous mobile robots to operate successfully. The localization process[1] is crucial to mobile robots and refers mainly to the precise determination of the coordinates where the system is present at a certain moment. For reliable navigation system, the mobile robot needs to adopt appropriate localization strategies to provide accurate information to the control system. Currently, the widely used localization technologies are infrared, sound, radar, inertial measurement, and ultra-wideband. Each has its advantages and limitations, so applying each method individually can only perform well in specific situations or environments. [2]Incorporating different localization methods in specific ways to eliminate or minimize limitations is necessary to enhance the accuracy performance and ensure the localization system can be reliable in a variety of situations.

1.1.2 Inertial Measurement Unit Introduction

An inertial measurement unit (IMU) is an electronic device that measures and reports the body’s accelerometers, gyroscopes, and sometimes magnetometers

which can be used to calculate attitude, linear velocity, angular rate, and position relative to the global frame.[3] An individual inertial sensor can only sense a measurement along or about a single axis. Three individual inertial sensors must be mounted together into an orthogonal cluster known as a triad to provide a three-dimensional solution. This set of inertial sensors mounted in a triad is commonly referred to as a 3-axis inertial sensor, as the sensor can provide one measurement along each of the three axes. Similarly, an inertial system consisting of a 3-axis accelerometer and a 3-axis gyroscope is referred to as a 6-axis system as it provides two different measurements along each of the three axes for a total of six measurements. In general, IMUs are to be mounted on the center of gravity of the object to be measured. The accelerometers detect the acceleration signals of the object in the carrier coordinate system independent of the three axes, while the gyroscopes detect the angular velocity signals of the carrier relative to the navigation coordinate system, measure the angular velocity and acceleration of the object in three-dimensional space, and use them to solve the attitude of the object.[4] IMUs are mostly used in devices that require motion control, such as automobiles and robots. They are also used in applications where precise displacement derivation using attitude is required, such as inertial navigation equipment for submarines, aircraft, missiles, and spacecraft.

IMU's sample frequency is high, and it is sensitive to changes in speed and angular rates, so it can provide relatively accurate information quickly and it can provide relatively accurate in a short period of time. For example, when the

position of the previous step is already known and the sample frequency is high, IMU data can be used to accurately calculate the position of next step.[5]However, IMU is prone to error that accumulates over time, also known as a drift. Since acceleration is always integrated by guidance systems concerning time to calculate the body's velocity and position, the measurement error will result in a linear error growth in velocity and a quadratic error growth in position. Similarly, the angle velocity error will result in linear error growth in angle calculation.[5]Therefore, it is necessary to use external information like GPS for assistance to realize combined navigation to effectively reduce the problem of error accumulation over time and achieve the highest velocity and position accuracy.[7] To improve reliability, it is also possible to equip each axis with more sensors.

1.1.3 Ultra-wideband Introduction

Ultra-wideband (UWB) is a short-range wireless communication technology that enables precise location tracking by precisely timing how long it takes a radio pulse to travel amongst different devices.[8] UWB is defined as a signal or system where the bandwidth exceeds more than 500MHZ or the fractional bandwidth is greater than 20%. Currently, UWB is capable of supporting a variety of data rates from 110 to 480 Mbps at distances up to 10 meters. Although UWB technology is seen as a new technology, the technology has been around for decades. The original definition of UWB came from the pulse communication technology that emerged in the 1960s, also known as pulse radio technology. Unlike carrier

modulation technology, which is widely used in today's communications, this technology communicates directly with baseband pulses that are jittered at both the rising and falling edges, so it is also known as baseband transmission or carrier-free technology. Most current commercial wireless communication methods use a communication carrier that is a continuous electric wave to which a data signal is added by some kind of modulation. UWB signals, on the other hand, are sent in very short, very fast pulses that are precisely timed to be only a few to tens of picoseconds long each. So this technology can transmit data with less power consumption.[9] Due to the wide operating bandwidth of UWB technology, the signal release is distributed throughout the entire signal bandwidth and the signal speed and capacity are high, so UWB is suitable for the transmission of large amounts of data, especially the transmission of digital multimedia signals. At same time, UWB can work better with other wireless technologies because it is not easy to interfere with others, which is due to the weak UWB signal of decentralized transmission. The system capacity is higher than any other radio technology because of the extremely low duty cycle of the impulse pulses emitted by the UWB radio signal, the high gain of the system, and the strong multipath discrimination capability. Because of the relatively large gain of the spread spectrum processing of UWB signals, even with a low gain omnidirectional antenna, a power of less than 1mw can be used to achieve several kilometers of communication.

There are two main localization techniques UWB usually uses, Time of Flight (TOF) and Time Difference of Arrival (TDoA).[10]

For ToF, the tag sends a poll packet to anchors and records the timestamp as T_{sp}, then the anchor receives the packet and records it as T_{rp}. Then anchor needs some time to receive and generate a response packet. When the anchor sends the response packet, recording the timestamp as T_{sr}, then the tag receives the packet and records T_{rr}. Similarly, the tag needs to wait some time to generate a new response message, when the tag sends a final message and records T_{sf}. Finally, the anchor receives the final message and records the timestamp as T_{rf}.

So the distance between the tag and anchor can be shown as:

$$D = \frac{(T_{rr} - T_{sp}) - (T_{sr} - T_{rp}) + (T_{rf} - T_{sr}) - (T_{sf} - T_{rr})}{4} \quad (1)$$

After the ToF method, the UWB tag can measure the distance between every anchor and itself. There will be 3 corresponding distances named d1~d3. Use each anchor as the center and d1~d3 as the radius to draw circles, then the intersection of three circles is the estimated tag position.[11]

For TDOA, it is localization based on comparing the time difference between each anchor receiving the signal from the tag. The UWB records the timestamp when each of the three anchors receives signal from UWB as T1, T2, and T3. The difference between the distance between tag and anchor1 and the distance

between anchor2 is C*(T1-T2) and the difference between the distance between tag and anchor1 and the distance between anchor3 is C*(T1-T3). According to these two values, two hyperbolas can be drawn, and the intersection of these two hyperbolas is the estimated tag position.[12]

The main difference between the two methods is the TDoA method needs to keep all anchors in sync. Besides, when the tag is on the periphery of the anchor or outside the area, ToF is more accurate than TDoA. Compared with TDoA, the ToF method is more suitable to be applied in places where the environment is complicated. However, concerning power consumption, the ToF method needs tag to send and receive signals several times to finish ranging with every anchor, but the TDoA only needs tag to send one broadcast message, so TDoA can make the battery life longer.

There are many applications of UWB positioning technology, including industrial production, warehousing, logistics, judicial prisons, sports, transportation, airports, petrochemicals, etc.[13]

UWB is one of the most popular indoor positioning technologies, with an accuracy of one centimeter in a line-of-sight (LOS) environment. However, non-line-of-sight (NLOS) occurs when walls or other objects block the UWB signal, which reduces the signal ratio and causes delays in signal transmission.[14] NLOS means two points of communication have an obstructed line of sight and greater than 50% of the Fresnel zone is blocked. Some of the obstructions absorb or

scramble certain radio frequencies, and some others reflect them. Both can limit the use of many types of radio transmission especially when the power budget is limited. Obstacles that typically cause NLOS propagation include trees, buildings, hills, mountains, and sometimes high-voltage power lines.[15] While UWB is an ideal candidate for a localization system, it can't avoid non-line-of-sight errors due to obstacles.[16] However, it is not plausible to eliminate all the obstacles before the mobile robots work in real environments. So UWB also needs to corporate with other techniques to keep high accuracy in a diverse environment.

1.1.4 Kalman Filter Introduction

The most commonly used filter algorithm is the Kalman filter, a state estimator that uses sensor fusion and information fusion to improve the accuracy of the system.[17]

Usually, we have two means to observe the state of a system. One is to derive the state of the next moment by using the state transfer equation of the system and combining it with the state of the previous moment. One is to obtain the state of the system with the help of measurements from an auxiliary system (a measurement system). Both of them have their uncertainties, and the Kalman filter can combine them optimally (weighted average) so that the uncertainty of the state we estimate is less than either of them. The advantage of the Kalman filter does not lie in how small the bias of its estimation is, but in its clever integration of the observed data and the estimated data, and the closed-loop

management of the error, which limits the error to a certain range. In other words, the Kalman filter can constrain the error in an acceptable range over a long period. The most critical parameters in the Kalman filter are Q and R, Q represents the process noise covariance and R represents the measurement noise covariance.[18] Adjustment of these two parameters has a great impact on the Kalman filter result. For the process noise Q value, the smaller the value means we have higher trust in the predicted value of the model and the system converges faster; for the measurement noise R, the larger the value means we have lower trust in the measured value.[19]

Kalman filter has many applications in technology. A common application is for guidance, navigation, and control of vehicles, aircraft, and spacecraft. In addition, the Kalman filter is a widely used concept in time series analysis for fields such as signal processing and econometrics. Kalman filter is also one of the main algorithms in the field of robot motion planning and control, and they are sometimes included in trajectory optimization.[20] The Kalman filter can also be used to model the control of motion by the central nervous system. Because of the time delay between issuing motor commands and receiving sensory feedback, the use of Kalman filters supports the actual model used to estimate the current state of the motor system and issue updated commands.

1.2 Thesis Contribution

To make our mobile robot location system more accurate, I combine IMU and UWB data in the Kalman filter to limit the position estimation error in a certain range. The linear acceleration, angular velocity, and absolute orientation quaternion are extracted from IMU and tag coordinates are extracted from UWB. To obtain an optimal set of parameters in our experiment environment, I adopt the fmincon method, a function to minimize the objective function to determine parameters in the Kalman filter and use repeated experiments in the same environment as samples.

Before using the IMU data in Kalman Filter, it is pre-processed to eliminate static error due to gravity influence and reduce noise by the mean filter algorithm.

The Non-Line-of-Sight (NLOS) propagation occurs when walls or other objects block the UWB signal transmission between anchors and tag, it will degrade the accuracy of the UWB position system greatly. I take advantage of IMU's high reliability in a short period to compare it with the UWB position data to detect the NLOS propagation, which means the NLOS propagation happens if the difference between the velocity measured by IMU and the velocity calculated by UWB data is too large. When the UWB position measurement is detected to be not accurate, then the signal is sent to adjust parameter of the Kalman Filter to reduce the weight of UWB in the estimated position and rely more on IMU acceleration data and keep a period.

For determining the degree of tilt and shaking of the mobile robot according to the pitch and roll angle of the mobile robot, I apply the accelerometer data and gyroscope data to the complementary filter, taking slow-moving signals from the accelerometer and fast-moving signals from the gyroscope and combining them. So the idea is to pass gyroscope signals through a high-pass filter and accelerometer data through a low-pass filter. When the mobile robot is detected shaking and bumping significantly, the level we trust the predicted value will decrease, so the parameters need to change with it.

Overall, the article optimizes the localization system of our mobile robot and combines the data from IMU and UWB to overcome their limitations on localization to perform better in various environments especially when the UWB measurement is not accurate due to NLOS. It provides basic reliable position information to the mobile robot control system to complete the tracking task.

2. Literature Review

Nowadays, mobile robot has been widely used in different fields. The localization process is crucial to mobile robots and UWB and IMU are the two most promising technologies for mobile robot localization systems.

However, since Time of arrival estimation in UWB is highly conditioned by the presence or absence of line of sight, the UWB measurement will severely degrade when NLOS occurs. Peng Dai analyzed the ranging results caused by NLOS along with the obstacle moving. In his experiment, the two UWB ranging

nodes are fixed at two points, and the distance between the two nodes is 130cm. Different types of obstacles move between the points and the measurement of the distance is shown below.

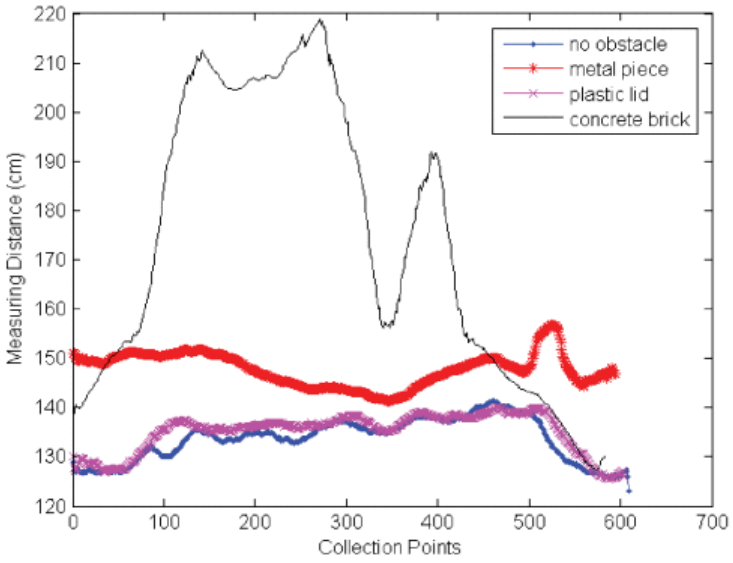

Figure 1 Obstacles moving from A to B[21]

From his experiment result, it can be concluded that obstacles can cause obvious error in distance measurement, especially the concrete brick. NLOS identification and mitigation methods are divided into two categories, direct path estimation based method and statistics based method.

Zhuoqi Zeng proposed a method to use channel impulse response to detect NLOS in the real field test. The CIR c(t) is the sum of all the received pulses and could be described with Equation (2):

$$c(t) = \sum_{k=1}^{K} a_k \delta(t - \tau_k) \quad (2)$$

Where K is the total number of the multipath components, a_k and τ_k represent the amplitude and time delay of the k arrived path. The CIR samples show different characteristics in LOS and NLOS situations, which can be used to detect NLOS propagation. The CIR under LOS and NLOS with people and water block are shown below.

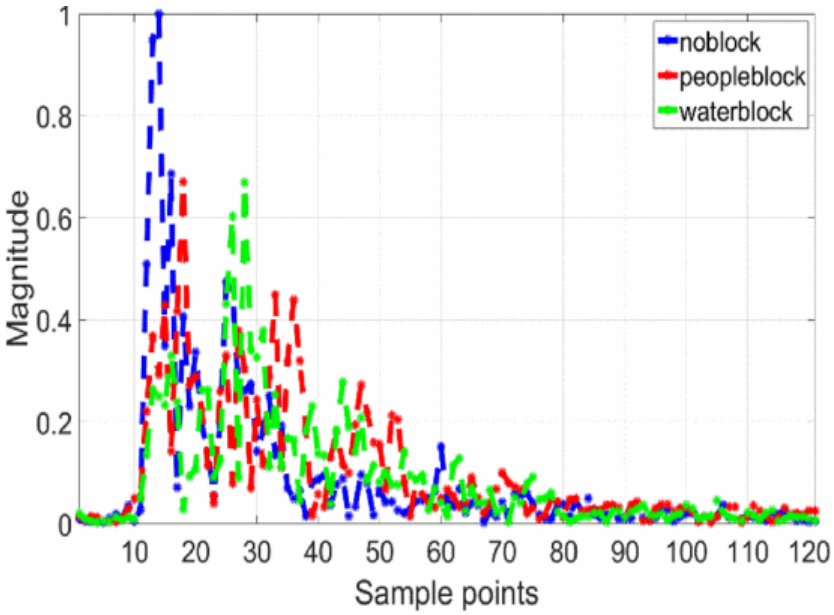

Figure 2 CIR under LOS and NLOS with people, water block[22]

Different features like rise time, rise time, mean excess delay, and maximum amplitude are calculated to be used in SVM to train data to detect NLOS propagation.

Meanwhile, IMU suffers from accumulation error, also known as drift. Guoqiang Xu classified the IMU error into three kinds: dynamic drift, static drift, and

stochastic drift.[23] The first two errors can be filtered and eliminated by discrete filters but the last one is hard to process so IMU can't be used individually to provide high accuracy in the navigation system.

Shanwen Guan proposed a noise removal algorithm based on LSTM.[24] The low-cost commercial IMUs are very noisy, so integrating IMU acceleration data directly may lead to worse positioning data. LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. The data with the timestep length is used as the sample feature, and the data at a position after the timestep is used as the sample label.

Alvin et al. proposed a method to fuse the acceleration data from IMU and the 2D coordinates from UWB anchors to increase the robustness and accuracy of the location.[25] The method adds the fused displacement calculated by IMU displacement and UWB displacement to the last known position to produce a precise location estimate as Equations (3) and (4) show.

$$\Delta xFused = \frac{\Delta xIMU\sigma_{IMU}^2 + \Delta xIPS\sigma_{IPS}^2}{\sigma_{IMU}^2 + \sigma_{IPS}^2} \tag{3}$$

$$\Delta yFused = \frac{\Delta yIMU\sigma_{IMU}^2 + \Delta yIPS\sigma_{IPS}^2}{\sigma_{IMU}^2 + \sigma_{IPS}^2} \tag{4}$$

Where ΔIMU is the IMU displacement for IMU and ΔIPS is the UWB displacement, σ_{IMU} is the variance of IMU measurement and σ_{UWB} is the variance of UWB measurement.

Xudong Yue represented the BP-EKF method to process IMU data to reduce the error caused by IMU using EKF in order to eliminate high-order Terms of Tyler expansion.[26] Backpropagation (BP) method is a widely used algorithm for training feedforward artificial neural networks. The algorithm uses the BP neural network to predict the error between the actual value and the filtered value to correct the EKF error.

Fengbo Wu developed a new algorithm using the difference between the IMU positioning system data information and the UWB two-way ranging information as the measurement[27], assigning weights separately to remove outliers, and then using the Kalman filter to perform downhole positioning and attitude angle calculation. The article uses the estimated UWB position result of the previous step as the predicted value and the current UWB position is included in the observation in the Kalman filter for optimization and IMU estimation.

H. Benzerrouk proposed the use of modern algorithms developed with a modified version of the Extended Kalman Filter[28], Sigma Point Kalman Filter, and Cubature Kalman filter to replace the standard filters.

To solve the problem that the IMU initial state is not stationary in practical application and IMU installation direction is difficult to measure, Shuaikang Zhang presented a rotation and motion state initialization algorithm and provided a novel tightly-coupled IMU and multiple UWB tags fusion framework based on a graph optimization model. The system framework is shown below.

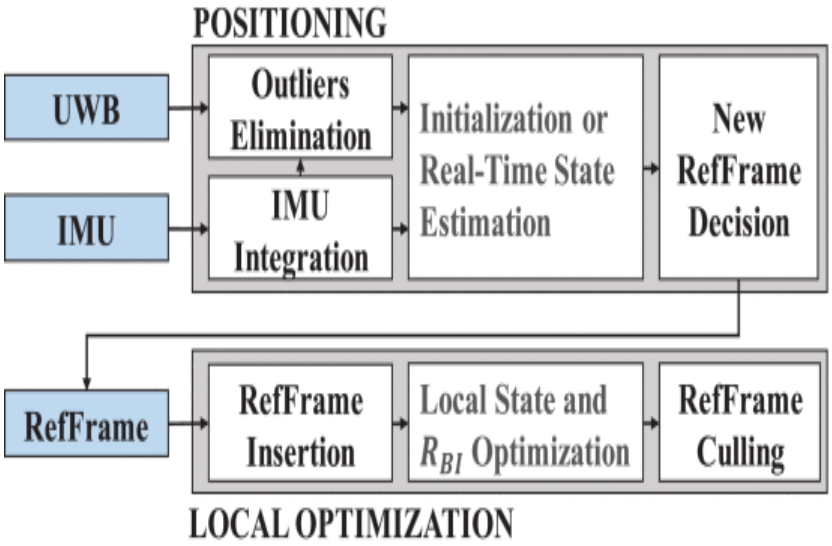

Figure 3 Real-time positioning and local optimization diagram[29]

From figure 3, the system contains a local optimization thread and a real-time positioning thread. They are separate to prevent real time process from being influenced by local optimization thread. The local thread can give real time positioning constraint reference.

The existing literatures have proposed various methods based on Kalman Filter to eliminate error caused by IMU and UWB measurement and initialization. My research proposes a new method to detect UWB anomaly by comparing the velocity calculated by UWB and the velocity calculated by IMU based on the fact that IMU calculated state is relatively accurate in a short period. Then my fusion algorithm will combine UWB and IMU data optimally also based on Kalman Filter according to the detection result. The algorithm is less complex and takes full

advantage of UWB's ability to be accurate enough under normal condition and IMU's ability to calculate accurate velocity and position in a short period.

3. Experimental Platform

3.1 Raspberry Pi

Raspberry Pi is a series of small single-board computers. For the operating system, the Raspberry Pi Foundation provides Raspberry Pi OS, a Debian-based Linux distribution. The Raspberry Pi I use in my project to interface IMU and UWB is Raspberry Pi 4 Model B. My Raspberry Pi board and its GPIO pinout are shown below.

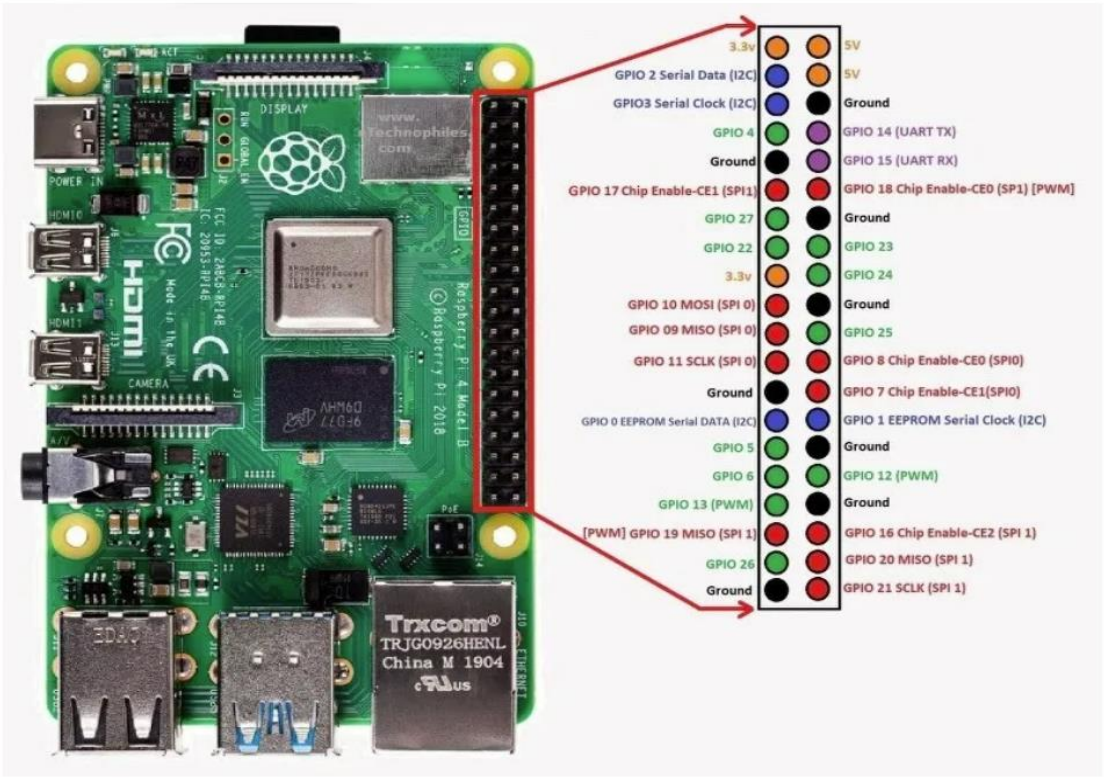

Figure 4 Raspberry Pi GPIO pinout

3.2 Inertial Measurement Unit BNO055

The BNO055 smart sensor board is a compact board that provides a triaxial 14-bit accelerometer, an accurate close loop triaxial 16-bit gyroscope, a triaxial geomagnetic sensor, and a 32-bit microcontroller running the BSX3.0 Fusion software.

The data outputs and their brief explanation are shown below.

Data	Explanation

Absolute Orientation(Euler Vector 100HZ)	Three-axis orientation data based on a 360° sphere
Absolute Orientation (Quaternion, 100Hz)	Four-point quaternion output for more accurate data manipulation
Angular Velocity Vector (100Hz)	Three axes of ‘rotation speed’ in rad/s
Acceleration Vector (100Hz)	Three axis of acceleration (gravity + linear motion) in m/s^2
Magnetic Field Strength Vector (20Hz)	Three axis of magnetic field sensing in micro-Tesla (uT)
Linear Acceleration Vector (100Hz)	Three axis of linear acceleration data (acceleration minus gravity) in m/s^2
Gravity Vector (100Hz)	Three axis of gravitational acceleration (minus any movement) in m/s^2
Temperature (1Hz)	Ambient temperature in degrees Celsius

Table 1 IMU data outputs and explanation

The BNO055 hardware board is shown below.

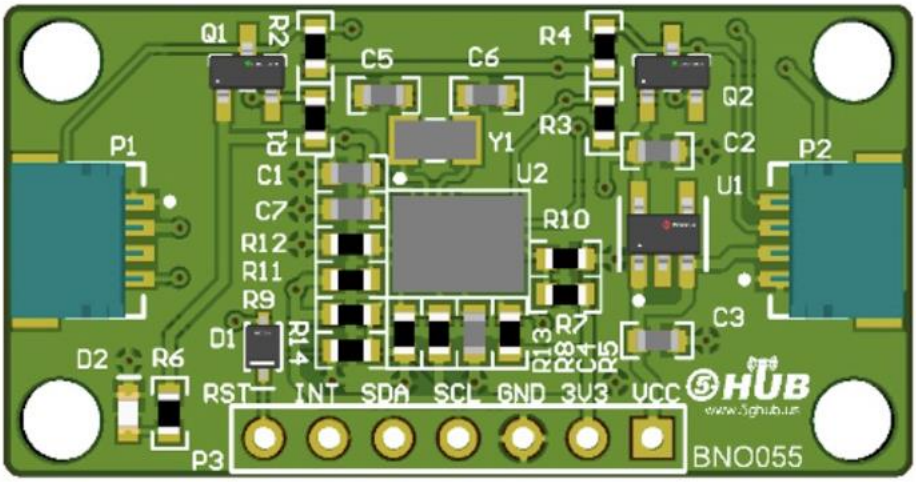

Figure 5 BNO055 hardware board

The way of connecting wires from BNO055 to Raspberry Pi is Vin – Power; GND – Ground; SDA -GPIO23; SCL – GPIO24.

Serial Data(SDA) is used to transfer data that takes place through this pin and Serial Clock(SCL) is used to carry the clock signal to synchronize the conversation between devices.

3.3 Ultra-wideband Decawave1001

The DWM1001-DEV Development Board is shown below.

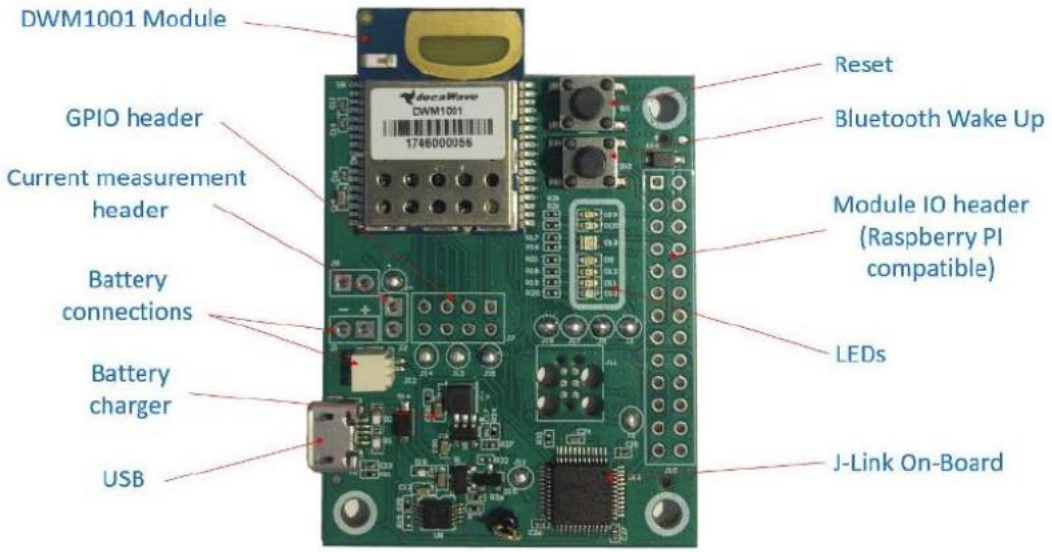

Figure 6 DWM1001-DEV Development Board

For the system setup, I select four of the boards as anchors and one board as a tag. The anchors are powered by rechargeable batteries, the tag is connected to Raspberry Pi by USB. Then an android smartphone is used to configure and show localization.

3.4 Robot Operating System

ROS is an open-source meta-operating system for robots. It provides the services expected of an operating system, including hardware abstraction, underlying device control, implementation of common functions, inter-process messaging, and package management. It also provides the tools and library functions needed to acquire, compile, write, and run code across computers. In my project, the ROS system is used to export IMU and UWB data to txt files.

3.5 Data Export Process

The data need to be exported and transferred to excel form to be processed in Matlab. The first step is to initialize. In IMU, I initialize the driver, open the I2C bus, load calibration offset and radius data, set operation mode to IMU, get calibration status data, and if the Pi successfully gets the accelerometer data, magnetometer data, gyroscope data, Euler angles data, and quaternions data. In UWB, I check if the serial port is open and if the anchors are created successfully. If anything is wrong, the Pi will terminate the program operation and reports the error. The errors usually occur when the wires from IMU to Pi are not connected properly or the anchors in UWB are out of work.

After initialization of these two devices, the next step is to create various messages to pack IMU and UWB data, such as Marker message to pack tag positions, IMU messages to pack acceleration, orientation, angular velocity information, and magnetic field message to pack magnetic field information provided by IMU. Then the third step is to export these messages to corresponding topics and then publish these topics. After that, the launch file is created to roslaunch the node. When running the program, the Raspberry Pi can subscribe to the topics and save all of them or certain topics of them as .bag files. After that, I play the bag and save the topics I need like tag and imu data in the txt files in Pi. Finally, I can extract the key information in these txt files and create excel files to be processed in Matlab. The ROS provides a convenient way to publish and subscribe to information. According to the data type needed in

Matlab, I can choose different messages to process the data conveniently and effectively.

3.6 Experiment Device Setup

The four UWB anchors are fixed on the four corners of the experiment site, and the UWB tag and the IMU are connected to Raspberry Pi powered by the portable charger. The monitor, mouse, and keyboard are connected to the Raspberry Pi. The setup figure is shown below.

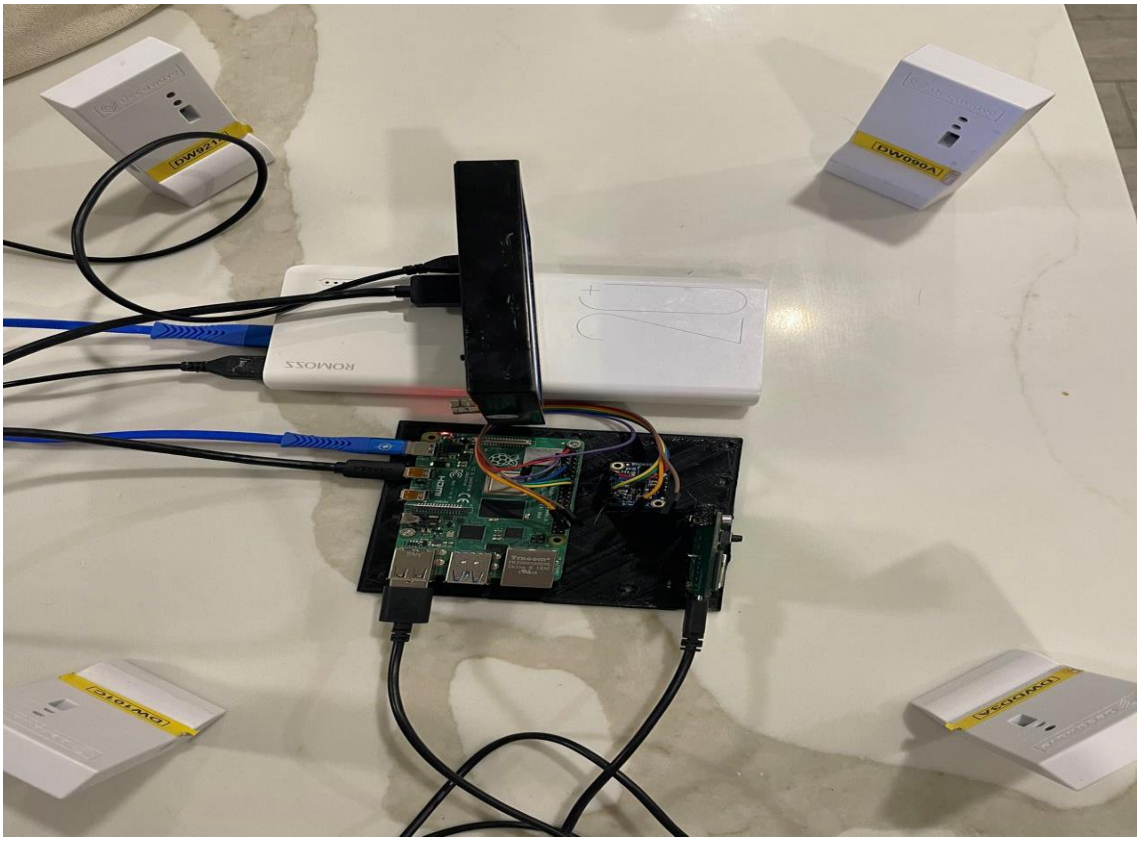

Figure 7 The experiment device setup

4. IMU and UWB Error and Limitation Analysis

4.1 IMU Error Analysis

In chapter 2 I mentioned that IMU and UWB have their own limitations, which lead to error and location inaccuracy in certain situations.

To analyze the accumulated drift of IMU, I conducted a stationary experiment, which means keeping the device stationary and recording the accelerometer data. The raw accelerometer value is shown below.

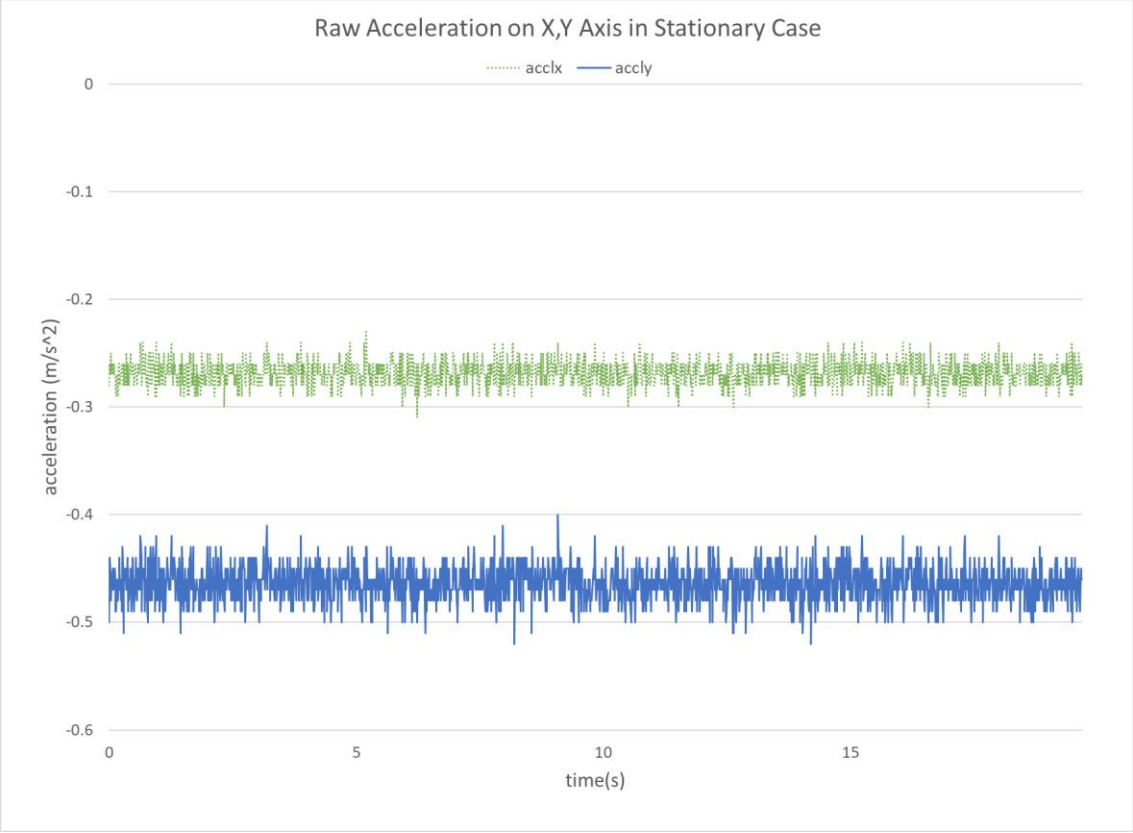

Figure 8 The raw accelerometer data from IMU in the stationary case

As figure 8 shows, the acceleration is closer to zero, but when the accelerometer data is used to derive the position information, the small error still results in a huge error after a long time of accumulation. The x position and y position deduced only from IMU acceleration data are shown below.

Figure 9 The x and y position calculated by IMU acceleration data

As I mentioned in chapter 2, since acceleration is always integrated by guidance systems with respect to time to calculate the body's velocity and position, the measurement error will result in a linear error growth in velocity and a quadratic error growth in position. So even after eliminating the gravity influence on the accelerometer, the acceleration value still results in position drift.

Besides, the angular velocity around three axes are shown below.

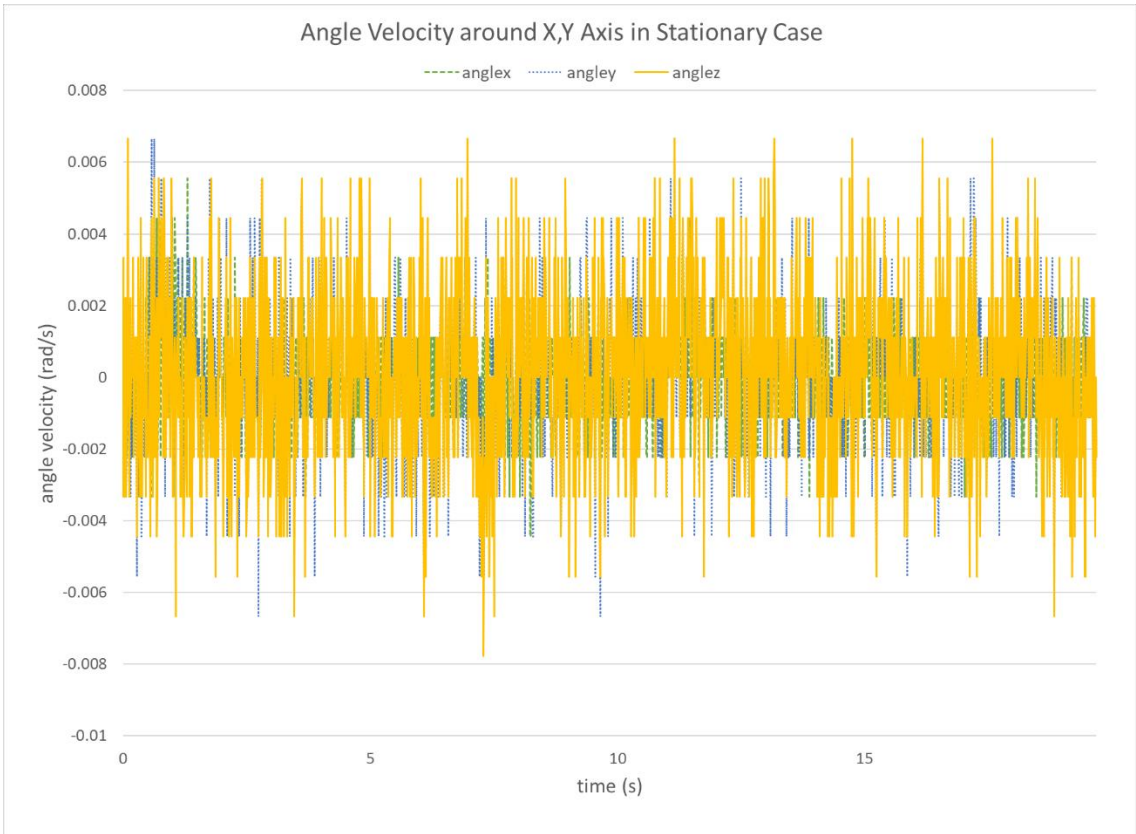

Figure 10 Angle velocity around x and y axis from IMU

From figure 10, the pitch, raw, and yaw angular velocity are close to zero when the device keeps stationary. But if only the angular rate data is used to calculate the pitch, raw, and yaw angle of our mobile robot, the small error of angular velocity will accumulate to an unacceptable value which results in huge errors in the mobile robot's posture.

4.2 UWB Error Analysis

To verify the accuracy of Decawave 1001, I first chose an open and flat ground next to the laboratory. The experiment scenario is shown below.

Figure 11 UWB accuracy test experiment site

As figure 11 shows, the four anchors are fixed on the four corners of the 6m*6m square. The coordinates are marked on the two sides of the square to obtain the coordinate information of each point inside the square assisted with the laser pointer. Then I fix the UWB at the coordinate (1,0) and record the UWB data.

Taking the average of the x-axis error and y-axis error as the measurement error, the error graph is as follows.

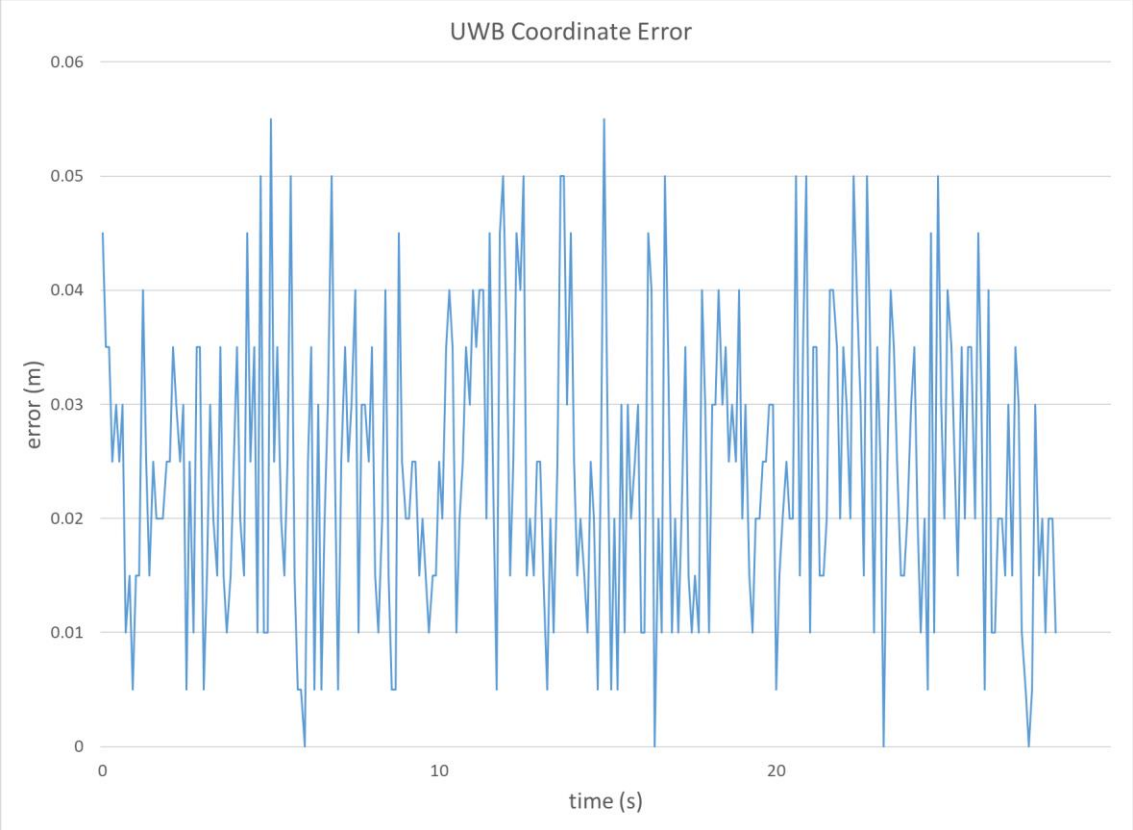

Figure 12 UWB coordinate error without NLOS propagation

In figure 12, it can be roughly estimated that the error of UWB in an open field is within 6cm. It corresponds with the description of decawave1001 that the error of decawave1001 is within 10cm.

However, in the presence of obstacles in the environment, Non-Line-of-Sight will produce inaccurate results. Then I completed the UWB accuracy test with NLOS propagation. The experimental site is a 3*4 rectangle and the anchors are fixed

on the four corners of this rectangle. Two of the anchors are in the living room and two of the anchors are in the bedroom. The tag is fixed in (0.8,1). A plastic bag is added as an obstacle and placed between the tag and anchor.

Taking the average of the x-axis error and y-axis error as the measurement error, the error graph is as follows.

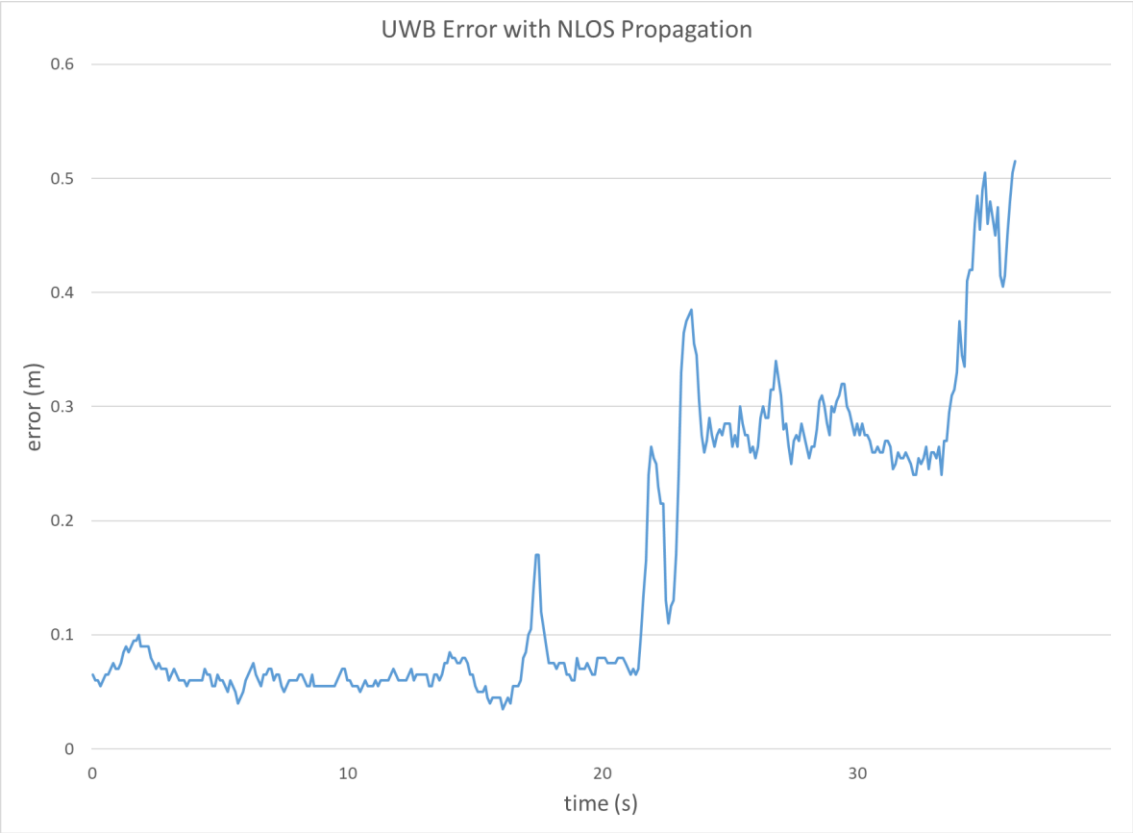

Figure 13 UWB measurement error with NLOS propagation

Figure 13 shows that even though the anchors are separated by the wall, the tag can provide normal and acceptable position to us and the error is within 10cm. It is because only two or three anchors are enough for providing position

information to tag to analyze and give an accurate position. But when the plastic bag is added, which means obstacles greatly affect the tag signal reception, the error will increase significantly to about 0.5m.

5. Filtering Algorithm Design

5.1 Data Pre-processing

To eliminate the static error due to gravity and various other factors which I mentioned in chapter 3, I decide to extract the mean value of the acceleration of the device on the x, and y axes of the first 300 times in the stationary state in a flat field. In the subsequent data processing, the linear acceleration and angular velocity minus the mean value I measured. The after-process acceleration and angular velocity data in the stationary experiment are shown below.

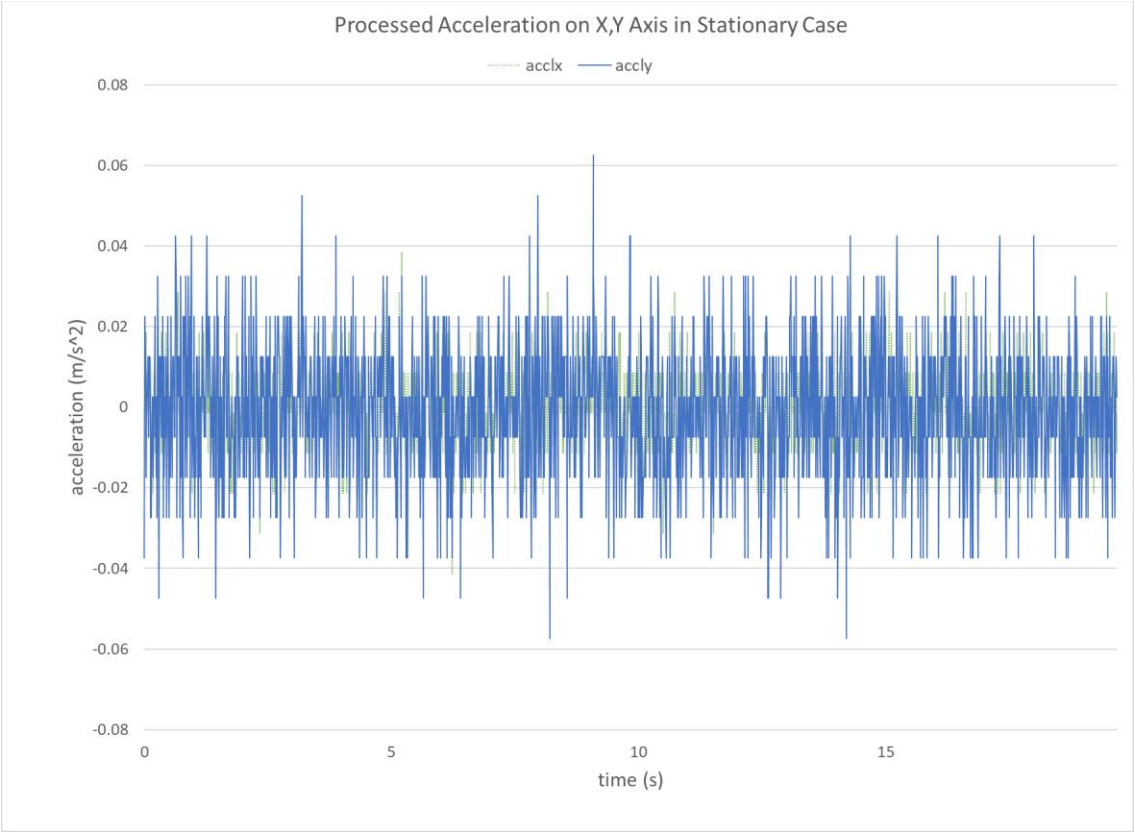

Figure 14 Processed acceleration on x,y axis in stationary case after eliminating the influence of gravity

From figure 14, after adopting the above method to eliminate static error, the acceleration data becomes more reasonable, and acceleration on the x axis and y axis are closer to zero than before.

To reduce the effect of oscillation of angular velocity data, I adopt the mean filter which is widely used in image processing. It is a simple, intuitive, and easy way to implement the method of smoothing images, reducing the amount of intensity variation between one pixel and the next. The main function of the mean filter

applying to IMU data is to replace each sample's value with the mean value of its neighbors. For example,

$$anglevelx(i) = \frac{1}{2}anglevelx(i) + \frac{1}{4}anglevelx(i-1) + \frac{1}{4}anglevelx(i-2) \quad (5)$$

The angular velocity data graph after the mean filter when the device is still is shown below.

Figure 15 Mean filter processed angle velocity around x,y axis in stationary case

From figure 15, the oscillation of angle velocity becomes smaller and the static error decreases significantly, so the mean filter is effective for the angular velocity data process.

5.2 Kalman Filter to Fuse IMU and UWB Data

The current consideration is to estimate the current state of the system and there are two known quantities, an estimate of the previous state and a measurement of the current state, both of which have a certain amount of noise, what needs to be done is to combine these two.

The first step is to predict state estimation:

$$x_{k+1|k} = Ax_{k|k} + Bu_k \tag{6}$$

In the first step, the position and velocity of the mobile robot are described by the linear state space.

$$x_k = \begin{bmatrix} x & y \\ \dot{x} & \dot{y} \end{bmatrix} \tag{7}$$

Where $\dot{x}$ is the velocity, the derivative of position with respect to time.

We assume the mobile robot is in uniformly accelerated linear motion in T.

$$x_{k+1} = x_k + v_k T + \frac{1}{2} a T^2 \tag{8}$$

$$v_{k+1} = v_k + aT \tag{9}$$

So

$$A = [1\ T; 0\ 1] \tag{10}$$

$$B = \left[\frac{1}{2} * T^2; T\right] \tag{11}$$

Where T = 0.01s, which is the IMU sample time.

For a, which is the system's input, can be expressed as:

$$a_k = [accx\ accy] * [cos\alpha\ sin\alpha; sin\alpha\ cos\alpha] \tag{12}$$

Where accx and accy are acceleration on the x and y axes from IMU data and α represents the angle of the mobile robot. The α can be obtained from the orientation quaternions described later.

The second step is to predict the error covariance.

$$P_{k+1|k} = AP_{k|k}A^T + Q \tag{13}$$

Where P represents the covariance matrix for an estimate, a measure of the estimated accuracy for the estimated state. This matrix is used to propagate the state estimate and state error covariance matrix appropriately. P value determines the initial convergence rate, and generally, a small value is set at the beginning to obtain a faster convergence rate. As the Kalman filter iterates, the value of P changes continuously, and when the system enters the steady state, the value of P converges to a minimum estimated variance matrix, and the Kalman gain at this time is also optimal, so this value only affects the initial convergence speed.

The Q matrix represents process noise for the system model. The system model is an approximation. Throughout the life of a system state, that system model fluctuates in its accuracy. Therefore, the Q matrix is used to represent this uncertainty and adds to the existing noise on the state. The value of Q and R which will be introduced later is crucial to the effect of the Kalman filter. I adopt the NMPC method to determine the value of Q and R and I will elaborate on the method later.

The third step is to compute the Kalman gain.

$$K_{k+1} = P_{k+1|k}H^T\left(HP_{k+1|k}H^T + R\right)^{-1} \tag{14}$$

The Kalman filter computes a Kalman gain for each new measurement that determines how much the input measurement will influence the system state estimate. In other words, when a noisy measurement comes in to update the system state, the Kalman gain will trust its current state estimate more than this new inaccurate information. This concept is the root of the Kalman filter algorithm and why it works. It can recognize how to properly weigh its current estimate and the new measurement information to form an optimal estimate. The Kalman filter uses the state-to-measurement matrix, H, to convert the system state estimate from the state space to the measurement space.

$$H = [1\ 0] \tag{15}$$

R matrix represents the measurement noise matrix, which means that the larger the R, the less trusting the measured value. Observability of a control system is

the ability of the system to determine the internal states of the system by observing the output in a finite time interval when input is provided to the system.

After determining the A matrix and H matrix, it can be confirmed whether the system is observable. The observability matrix O is shown below.

$$O = \begin{bmatrix} H \\ HA \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & T \end{bmatrix} \tag{16}$$

The determinant of the observability matrix O is T, a non-zero value, so the system is completely observable.

The fourth step is to update state estimation.

$$x_{k+1|k+1} = x_{k+1|k} + K_{K+1}\left(z_{k+1} - Hx_{k+1|k}\right) \tag{17}$$

After the Kalman gain is computed, it is used to weight the measurement appropriately in two computations. The first computation is the new system state estimate. The second computation is the system state error covariance. Z represents the position from UWB measurement.

The last step is to update the error covariance for the next step.

$$P_{k+1|k+1} = (I - K_{K+1}H)P_{k+1|k} \tag{18}$$

Since the raw IMU acceleration data can cause obvious and huge drift, which results in the linear growth of error of velocity. If only applying the mean filter and static error elimination method, the IMU error is still not small enough to be used in the Kalman filter to improve accuracy when the UWB suffers from NLOS

propagation error. So after every step, I calibrate the velocity according to the absolute orientation provided by the gyroscope.

$$V_x = \sqrt{V_x^2 + V_y^2} * cos\alpha \tag{19}$$

$$V_y = \sqrt{V_x^2 + V_y^2} * sin\alpha \tag{20}$$

Where α is the orientation angle around the z axis, also known as the yaw angle. In every step of the Kalman filter, the velocity direction of the mobile robot is calibrated to correspond with the orientation data which is relatively accurate and free from accumulation error.

5.3 Fmincon Method to Determine Q and R in Kalman Filter

The fmincon function in Matlab can find the minimum of the objective function, the function here to determine Q and R is the difference between the true position and Kalman filter estimated position. I collect three sets of data and record the true position of some timing points in the field shown below.

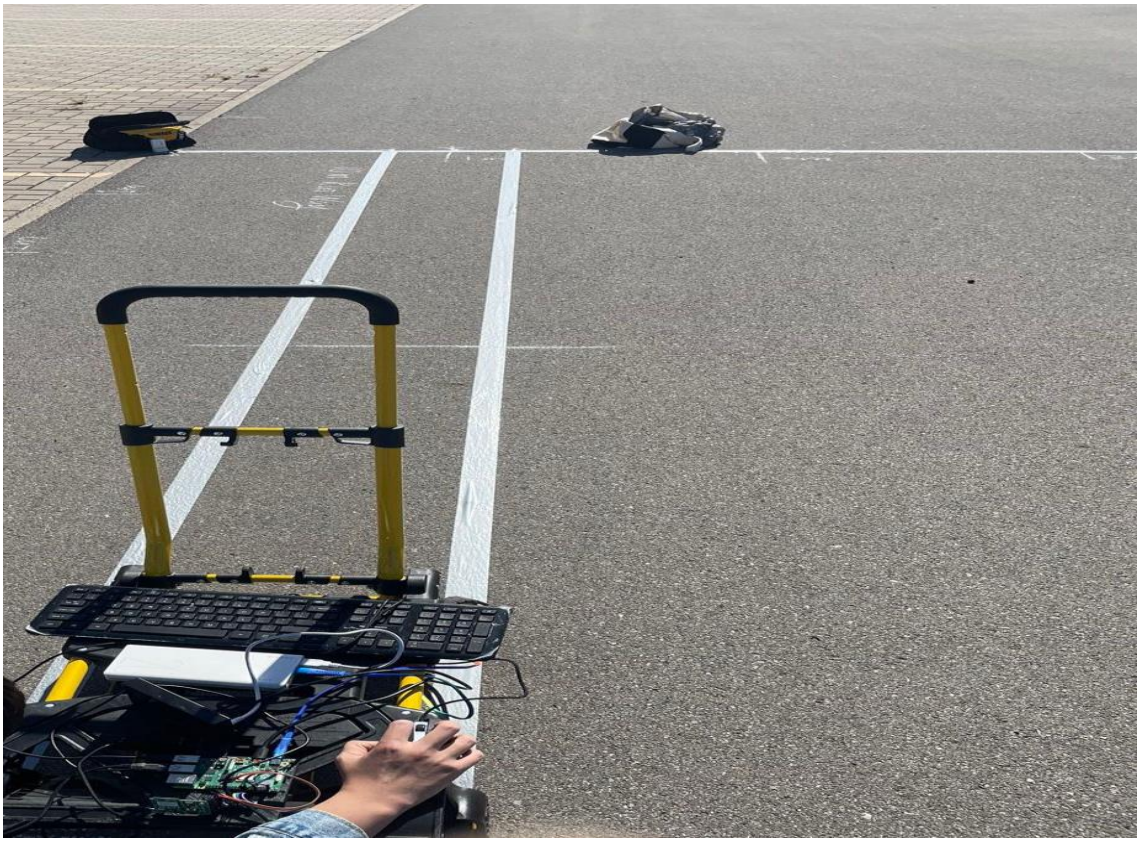

Figure 16 Experiment site to determine Kalman filter parameters

The device is fixed on the flat cart and the four anchors are fixed on the four corners of the 6m*6m square. The coordinates are marked on the two sides of the square to obtain the coordinate information of each point inside the square assisted with the laser pointer. Then I fix the UWB at the coordinate (1,0) first as the start point and then push the flat cart smoothly from the start point (1,0) to the endpoint (1,5). Every time the cart passes through the coordinate points (1,1),(1,2),(1,3),(1,4),(1,5), I press the stopwatch to record time. So the time of the cart passing through certain points can be known. Extract the corresponding position estimated by the Kalman filter at the same time as the stopwatch records

and get the error by using the estimated position minus the true position to be used in the objective function. I repeat the experiment three times.

The objective function is shown below.

$$obj = (x_1 - 1)^2 + \cdots + (x_5 - 1)^2 + (y_1 - 1)^2 + \cdots + (y_5 - 5)^2 \quad (21)$$

Where (x_n, y_n) represents the coordinate estimated by the Kalman filter when the cart passes through (1,n). The inputs of the objective function to get the (x_n, y_n) are Kalman filter parameters, UWB position data, IMU data, and stopwatch records.

For the fmincon algorithm, I choose the 'interior point' algorithm. It handles large, sparse problems, as well as small dense problems. The algorithm satisfies bounds at all iterations and can recover from NaN or Inf results. It is a large-scale algorithm.

The iteration results are shown below.

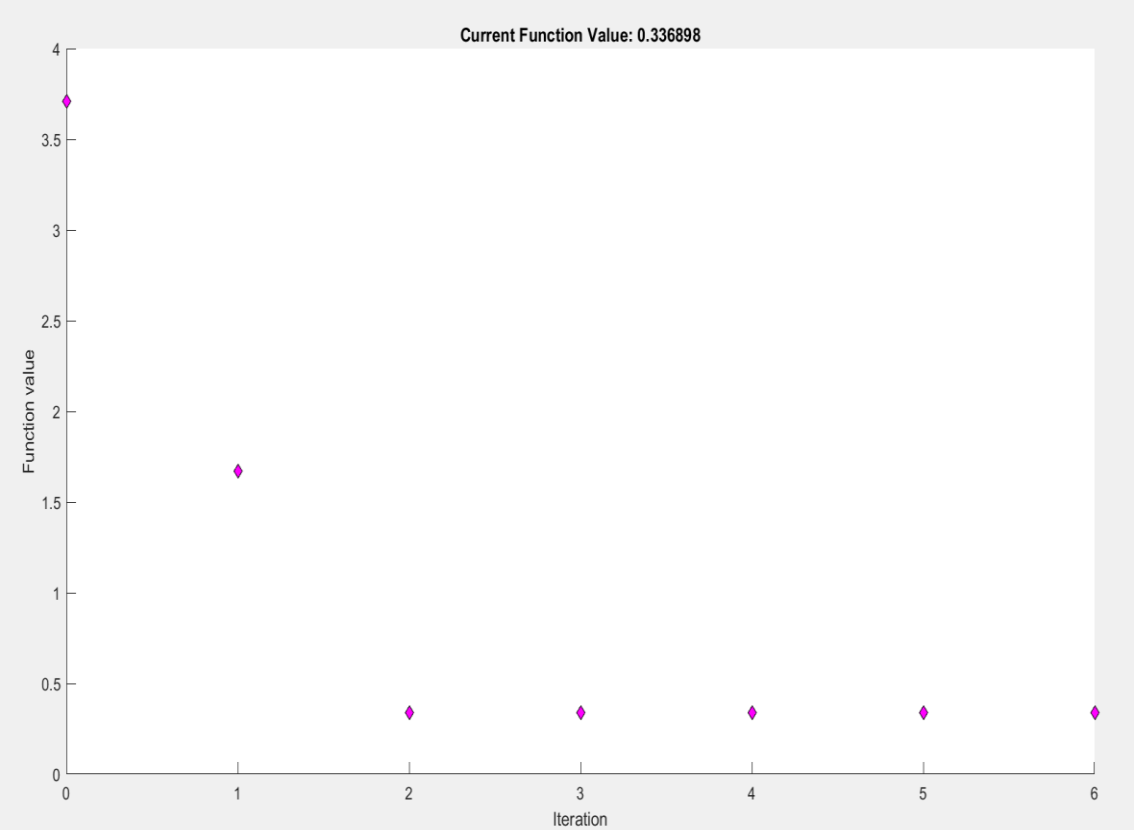

Figure 17 Fmincon iteration result

And the optimum solution R = 0.011358 Q=23.9714.

R is much smaller than Q since in the environment without obstacles because the UWB can provide accurate position data with the error less than 6cm.

The Q and R here is the optimum solution to the flat, no obstacles, and smoothly moving scenario. When the mobile robot does not move smoothly or the tilt of the mobile robot becomes larger, the trust in IMU data to estimate position needs to decrease and the Q needs to increase. When obstacles appear in front of the mobile robot, the UWB signal transmission will be certainly influenced so the trust

in UWB to get the position needs to decrease and the R needs to increase. In the last chapter, I have discussed that UWB accuracy decreases a lot, from 6cm to nearly 50 cm, so the adjustment of R needs to be large enough to prevent inaccurate data to influence the filter effect.

5.4 Complementary Filter for Tilt Degree

The idea behind the complementary filter is to combine the slow moving signal from the accelerometer with the fast moving signal from the gyroscope. The accelerometer provides a good tilt indicator under static conditions. The gyroscope provides a good tilt indicator under dynamic conditions. However, each method has its own limitation in certain situations, accelerometer becomes inaccurate when the device is tilted and moving translationally at the same time and when they are subject to sudden bumps or taps. Minuscule noise in the signal will eventually multiply to skew the calculated rotation angle by gyroscope.

Therefore, the idea is to pass the accelerometer signal through a low-pass filter and the gyroscope signal through a high-pass filter and combine them to get the final rate. The low-pass filters' task is to filter out signals that are too high and only let frequencies under a selected number pass through, high-pass filters do the opposite. The key point is that the frequency response of the low-pass and high-pass filters adds up to 1 at all frequencies.

First, by using the gravitational accelerometer data, the roll and pitch can be calculated. Formulas used to get the pitch and roll are shown below,

$$angle\ roll = atan\theta\left(\frac{GAcc_y}{\sqrt{GAcc_x^2 + GAcc_z^2}}\right) \quad (22)$$

$$angle\ pitch = -atan\theta\left(\frac{GAcc_x}{\sqrt{GAcc_y^2 + GAcc_z^2}}\right) \quad (23)$$

The result of this is multiplied by (180/Pi) to get a [-90 90] range.

Second, by using the gyroscope data,

$$\theta_{angle} = \theta_{angle} + w_{gyro} * dt \quad (24)$$

The mathematical model used for the complementary filter can be represented as below.

$$\theta_{angle} = \alpha * \left(\theta_{angle} + w_{gyro} * dt\right) + (1 - \alpha) * \theta_{acc} \quad (25)$$

θ_{angle} represents the calculated angle(pitch/roll), α is a filter weight constant, w_{gyro} is the angular velocity from the gyroscope, dt is the sampling time, and θ_{acc} is the calculated pitch or roll from the accelerometer data.

The flow chart of the complementary filter is shown below.

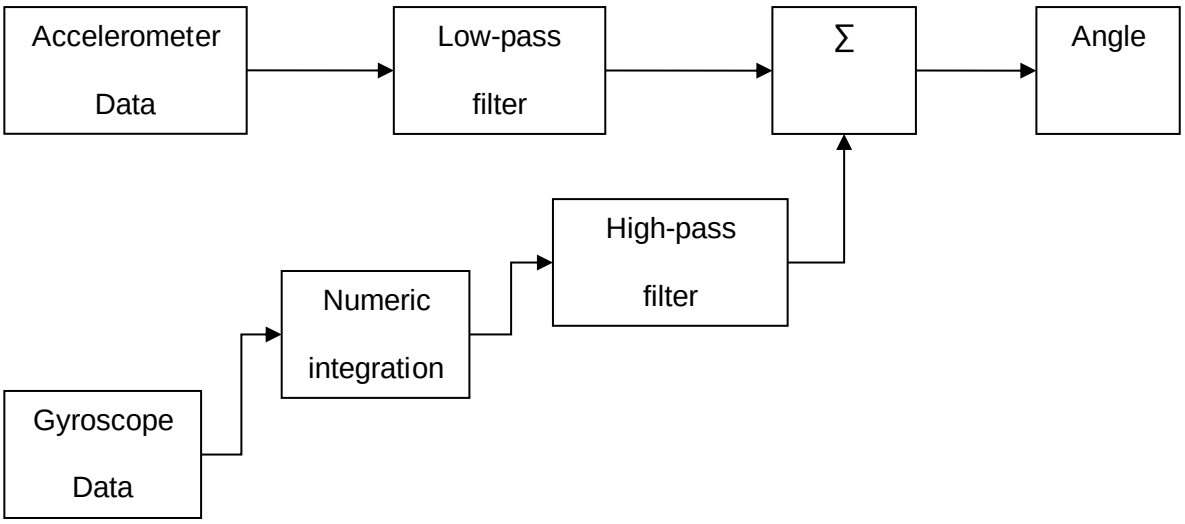

Figure 18 Complementary filter flow chart

The test experiment for the complementary filter is divided into two steps, In the first step, the device keeps stationary and flat on the table for the first 10 seconds, then it starts to rotate around the y-axis, returns to the original state when it turns to ninety degrees, then repeat the rotation twice. In the second step, the device keeps stationary and flat on the table for the first 10 seconds, then it starts to rotate around the y-axis, return to the original state when it turns to ninety degrees, then repeats the rotation twice.

The roll and pitch graph for the two steps are shown below.

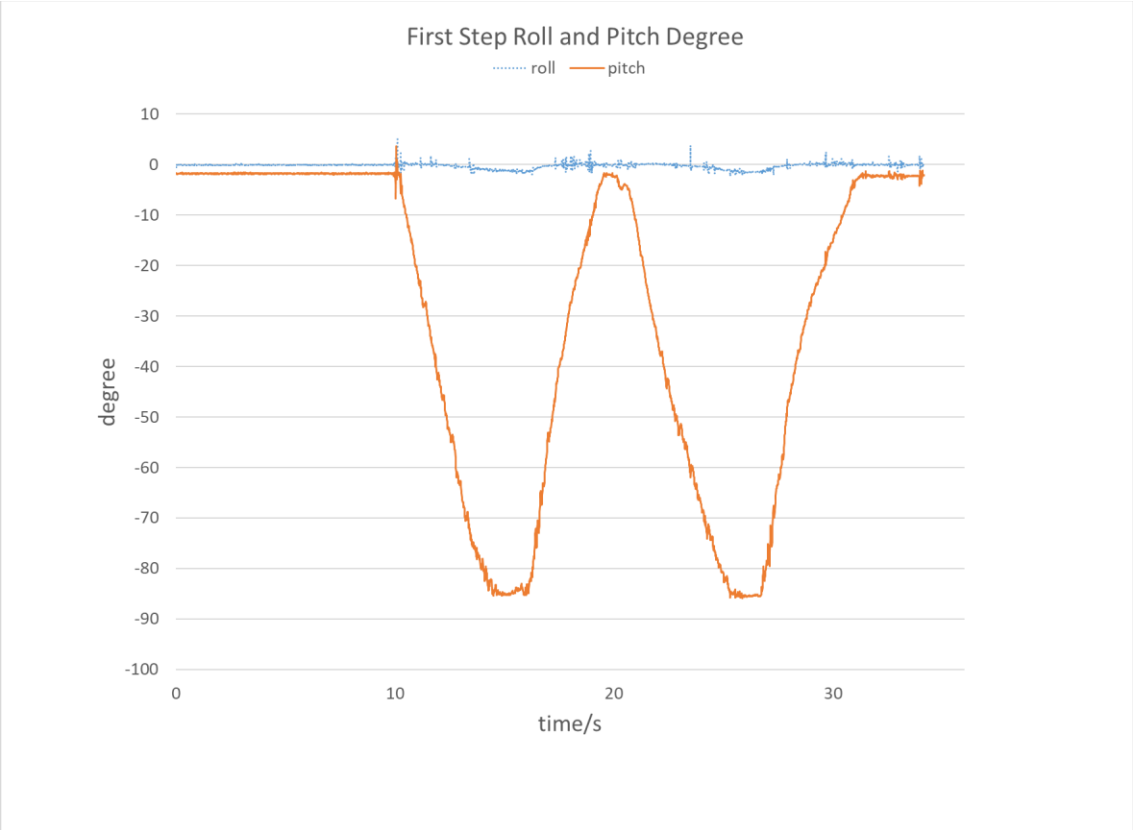

Figure 19 First step roll and pitch degree

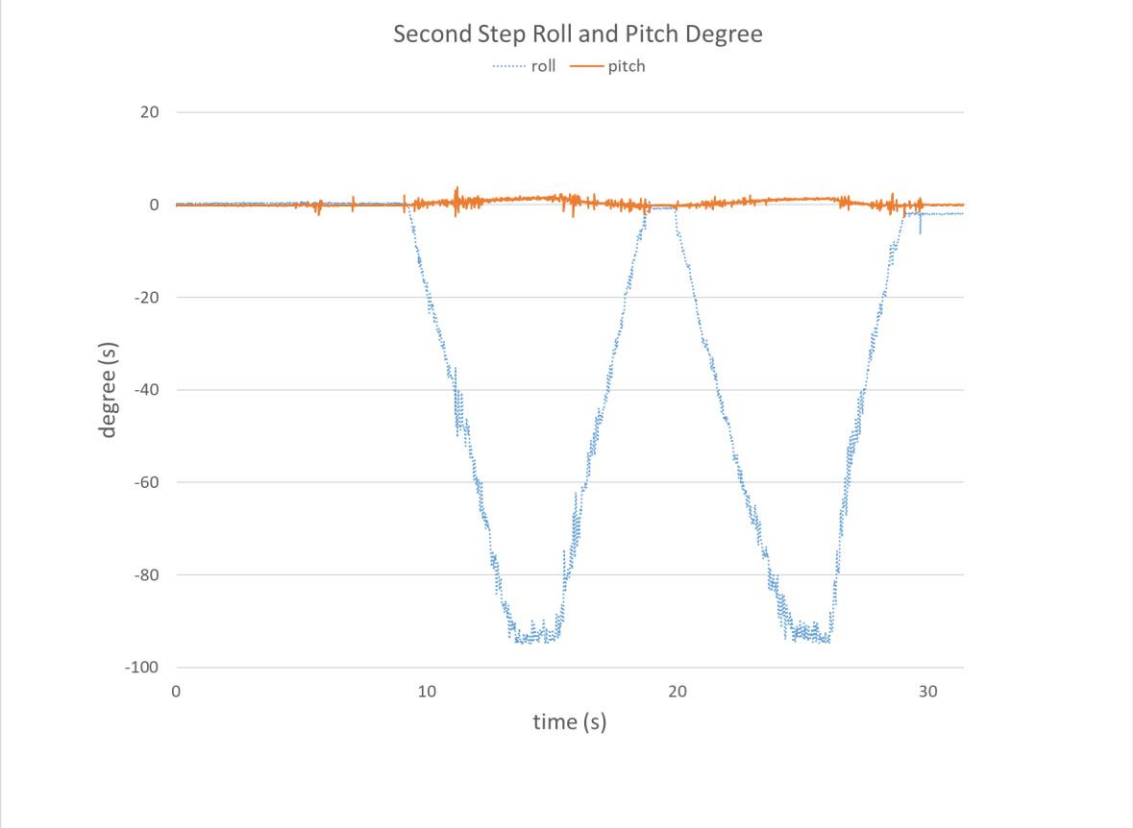

Figure 20 Second step roll and pitch degree

As figure 19 and 20 show, as the device rotates around the y axis, the pitch value will gradually rise close to 90 degrees and the roll value will keep stable around the original value. As the device rotates around the x axis, the roll value will gradually rise close to 90 degrees and the pitch value will keep stable around the original value. So the estimated tilt degree through the complementary filter is basically close to the actual tilt degree.

Regarding the selection of the filter weight constant, since the error of degree calculated by the gyroscope increases with the accumulation of time and the method using acceleration data is highly subject to sudden bumps and becomes

inaccurate when the device moves translationally when it is tilted, the work time and the general condition of the road need to be considered. If the mobile robot needs to keep in work mode for a long time, then the mobile robot needs to rely more on accelerometer. On the other hand, if the work time is short but the surface is uneven, the gyroscope is more reliable.

The purpose of the complementary filter is to determine the tilt of the mobile robot, according to which Q in Kalman filter will be changed since the state estimation of position and velocity is based on the assumption that the mobile robot drives on a smooth road and the device is not tilted. So if the tilt angle is large, the accuracy of state estimation will certainly decrease, which means the trust in state estimation in the Kalman filter need to decrease.

5.5 Data Analysis to Detect UWB Measurement Anomaly

Non-line-of-sight (NLOS) occurs when the obstacles move between the anchors and causes obvious error in UWB measurement as mentioned in UWB error analysis and literature review. When the NLOS occurs, the accuracy of UWB measurement degrades significantly, so the trust in it needs to decrease, which means R needs to increase and Q needs to decrease in the Kalman filter. There are two kinds of methods to detect NLOS propagation in experiments, the first one is to depend only on the camera and computer vision to detect the obstacles and derive the distance from the obstacles. However, the effect of obstacles on UWB accuracy varies greatly with different materials and different position of the

obstacles, so the system must be complicated if only relying on computer vision to determine NLOS propagation. The second option is to analyze the UWB data and compare it with IMU data. Since the IMU only suffers from accumulation error, so in a short period, the velocity calculated by IMU data is relatively accurate. If the velocity calculated by UWB measurements differs too much from the velocity calculated by IMU acceleration and gyroscope, it can be confirmed that the UWB anomaly occurs.

$$V_{imux}(i) = V_x(i-1) + a_x * T \tag{26}$$

$$\mathrm{V_{imuy}(i) = V_y(i-1) + a_x * T} \tag{27}$$

$$\Delta Vx = \frac{[xmeasure(i) - xmeasure(i-1)]}{T} - Vimux \tag{28}$$

$$\Delta Vy = \frac{[ymeasure(i) - ymeasure(i-1)]}{T} - Vimuy \tag{29}$$

$$\Delta V = \sqrt{\Delta Vx^2 + \Delta Vy^2} \tag{30}$$

Where xmeasure and ymeasure are the x and y coordinate measured by UWB and V_{imu} is the velocity of mobile robots calculated by IMU acceleration data combined with the velocity of the previous step. T is 0.1s, which is the sample frequency of UWB. If ΔV exceeds the set threshold, the NLOS occurs. Once ΔV is detected to be higher than the threshold, the R in Kalman Filter is raised and held for the next 2.5 seconds.

Regarding the selection of the threshold for UWB anomaly detection , I conduct test experiments to observe the difference between the velocity calculated by UWB position measurements and velocity calculated by IMU accelerometer and gyroscope data. Theoretically, since the UWB measurement error is within certain range under normal condition and the velocity calculated by IMU data in a short period is accurate, the difference of velocity calculated by these two ways is within certain range under normal condition.

Figure 23 shows the velocity difference between the two methods without obstacle interference.

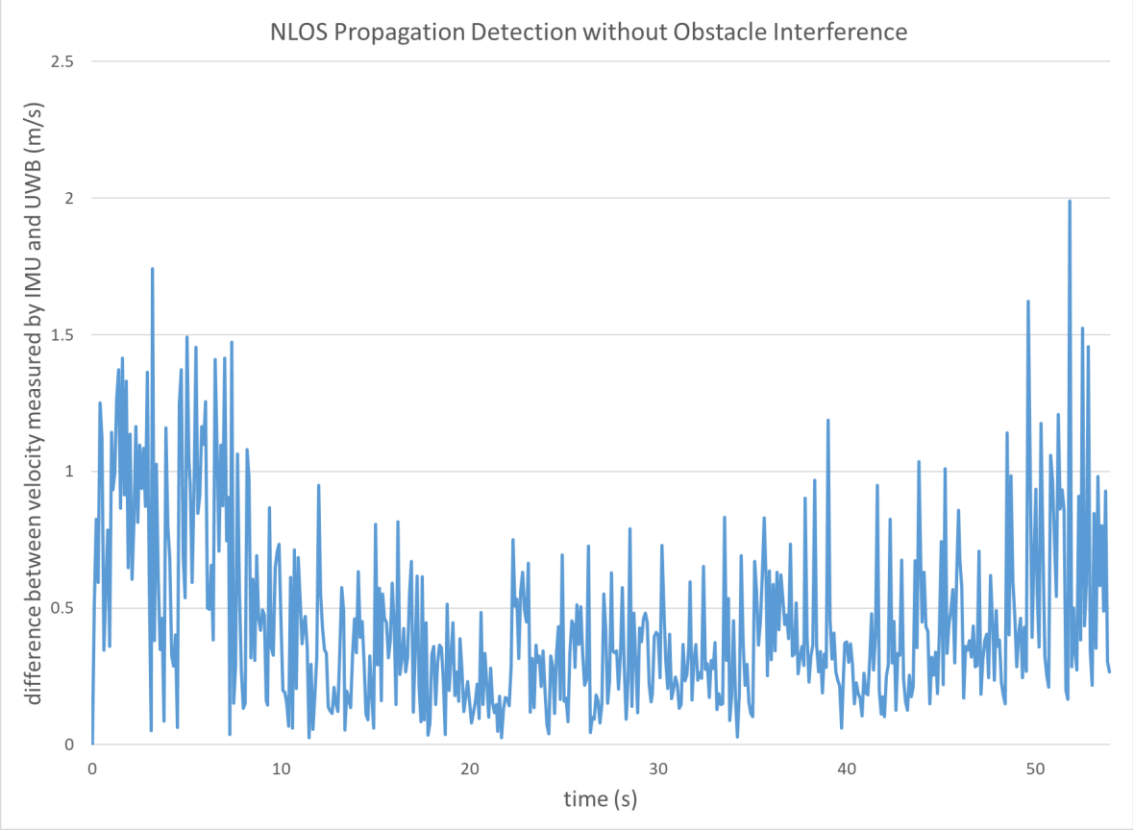

Figure 21 The velocity difference between the two methods without obstacle interference

As figure 21 shows, the maximum of velocity difference between UWB and IMU calculation methods is 1.96 m/s, so the threshold can be set close to and slightly larger than 1.96 m/s

6. Experiment

The experiment site is a 6m*6m square. The four UWB anchors are fixed on the four corners of the 6m*6m square and the device moves from point (1,0) to point (1,5) in straight line , turns 90 degrees and then moves to point (6,5). At second 6 and 36, the obstacle is added and removed after 8 seconds. There are two ways to evaluate the algorithm effect, the first one is to observe the extent to which the estimated trajectory deviates from the true trajectory. The second one is to use the stopwatch to record timestamp when the device passes point (1,0), (1,1), (1,2), (1,3), (1,4), (1,5),(2,5), (3,5), (4,5), (5,5), (6,5). Comparing these points with the coordinates at the same time after the fusion algorithm, the effect of the fusion algorithm can be estimated.

The result of pitch and roll of the device is shown below.

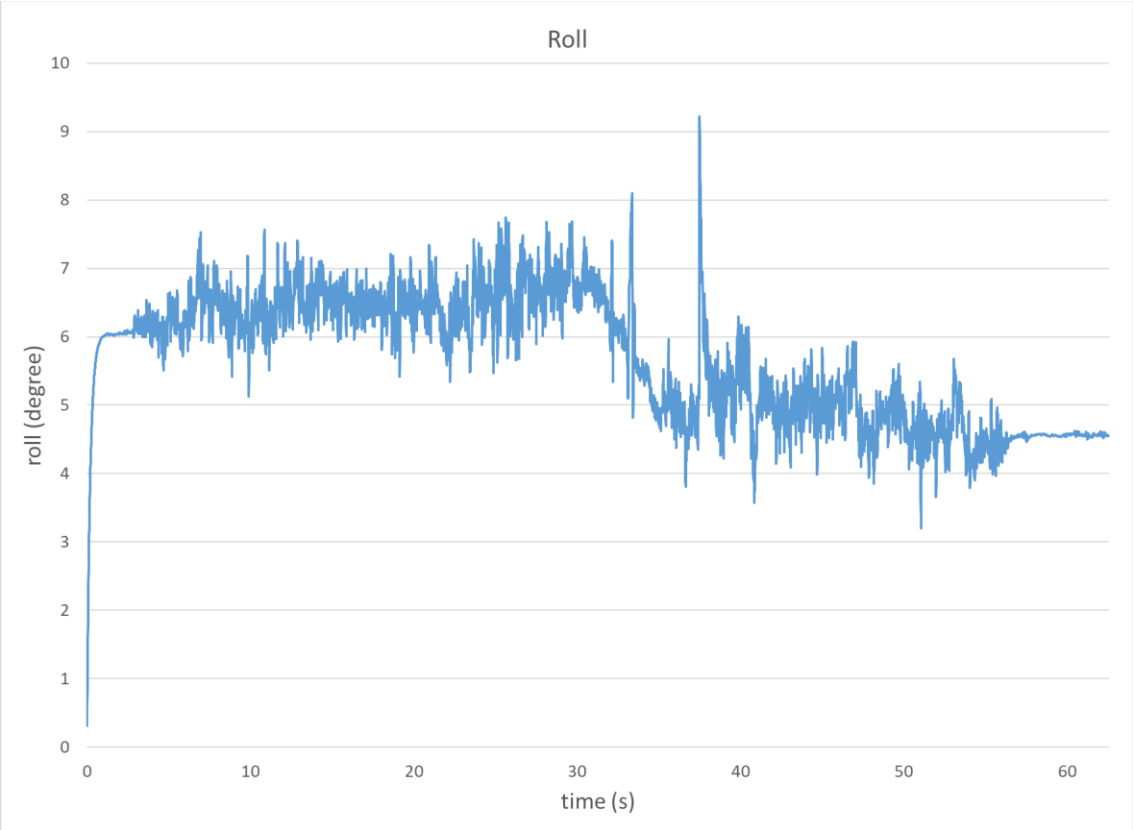

Figure 22 Roll degree of the device

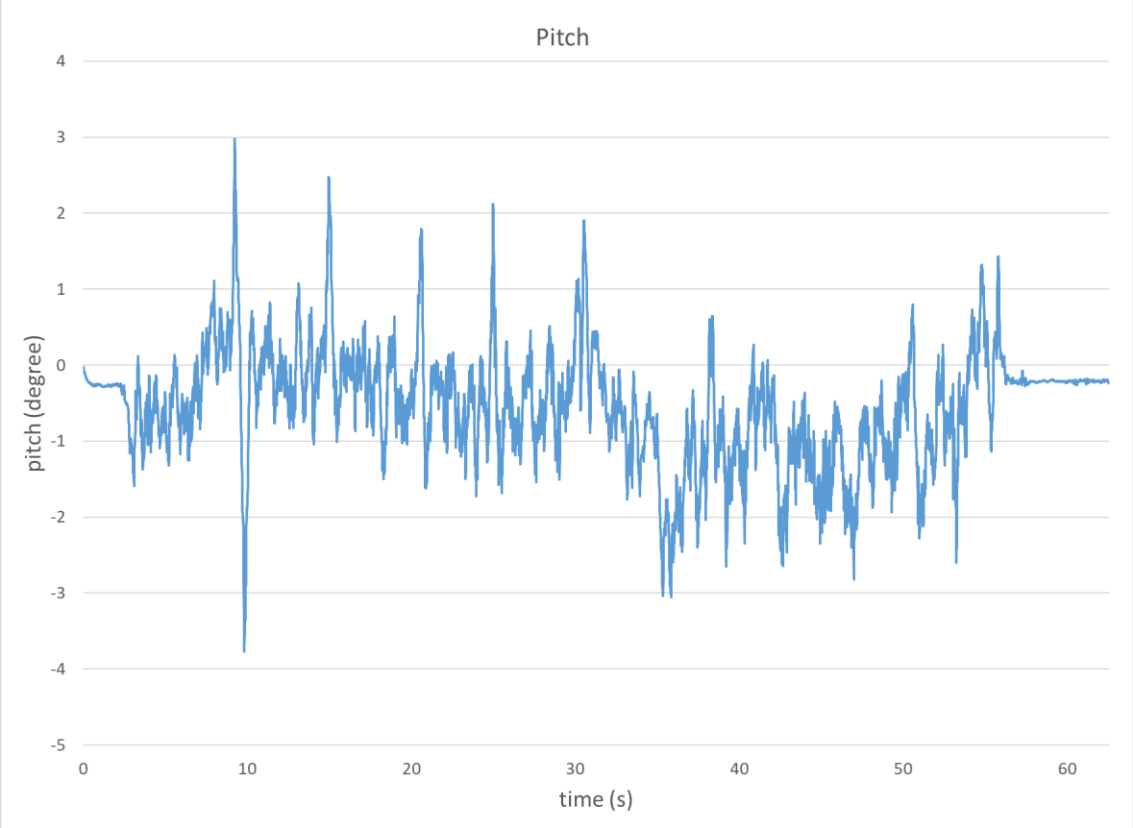

Figure 23 Pitch degree of the device

From figure 22 and 23, the vibration magnitude is very slight, in accordance with the real situation because the road is relatively flat. When pitch and roll degree are large, the Q needs to decrease to prevent inaccurate IMU data to influence the localization system.

The orientation of the device calculated by quaternion data from IMU is shown below.

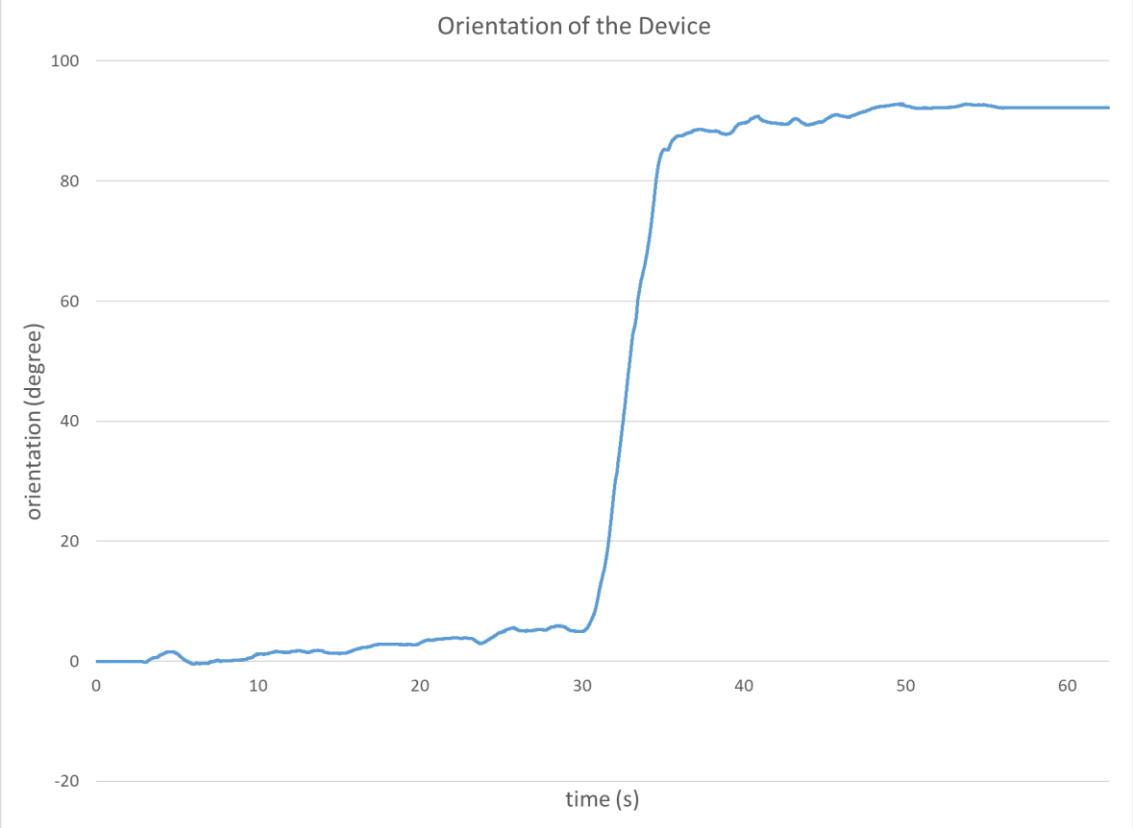

Figure 24 Orientation of the device

From figure 24, the orientation data corresponds with the actual situation as the orientation of the device keeps close to zero in the first half of the trajectory and increases to 90 degree in the second half of trajectory.

The result of the difference between velocity calculated by IMU and UWB is shown below.

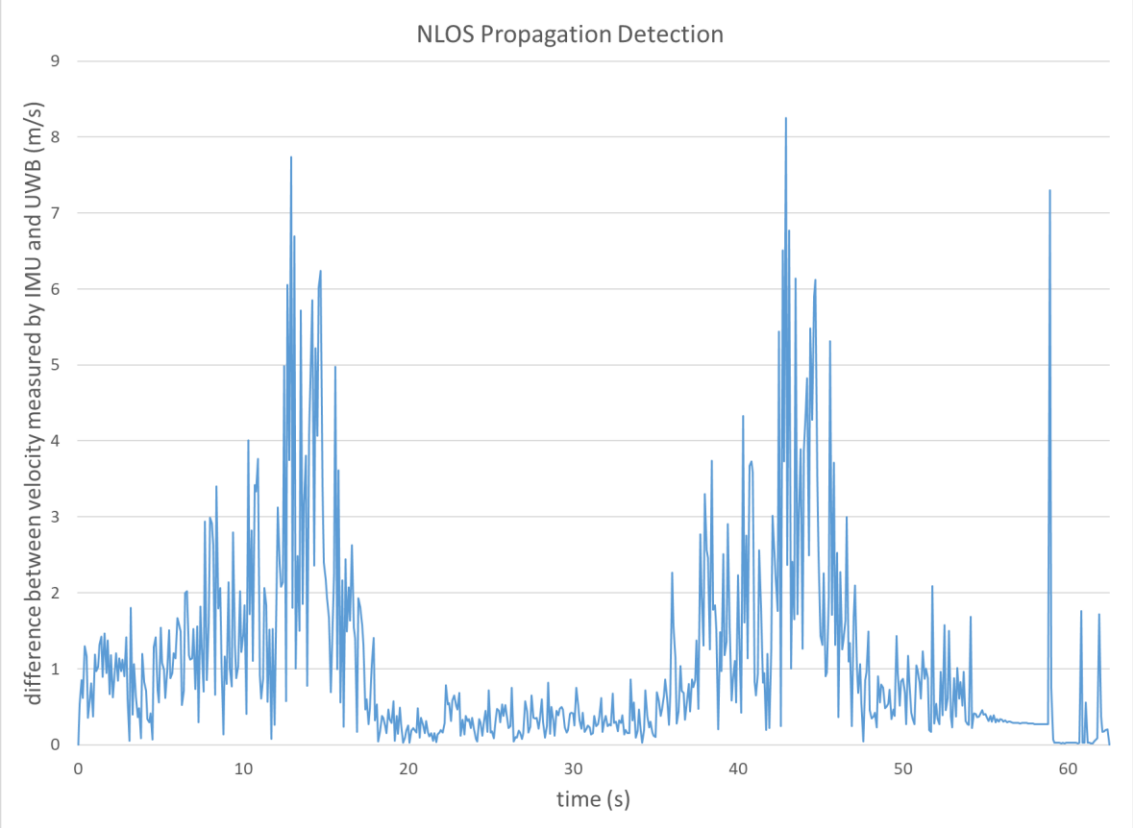

Figure 25 Difference between velocity calculated by UWB and IMU

Figure 25 shows that the velocity difference between UWB and IMU keeps close to zero before obstacle is added, but increases obviously after 6s and 36s, which are exactly the time the obstacle is added. It proves the NLOS propagation detection method is effective. To further analyze the accuracy of the detection algorithm, I compare it with the UWB measurement data. Since in the first half of the experiment, the device's x coordinate remains unchanged and the device's y coordinate remains unchanged during the second half, it is easy to tell from the UWB x,y coordinate graph whether UWB is accurate or not.

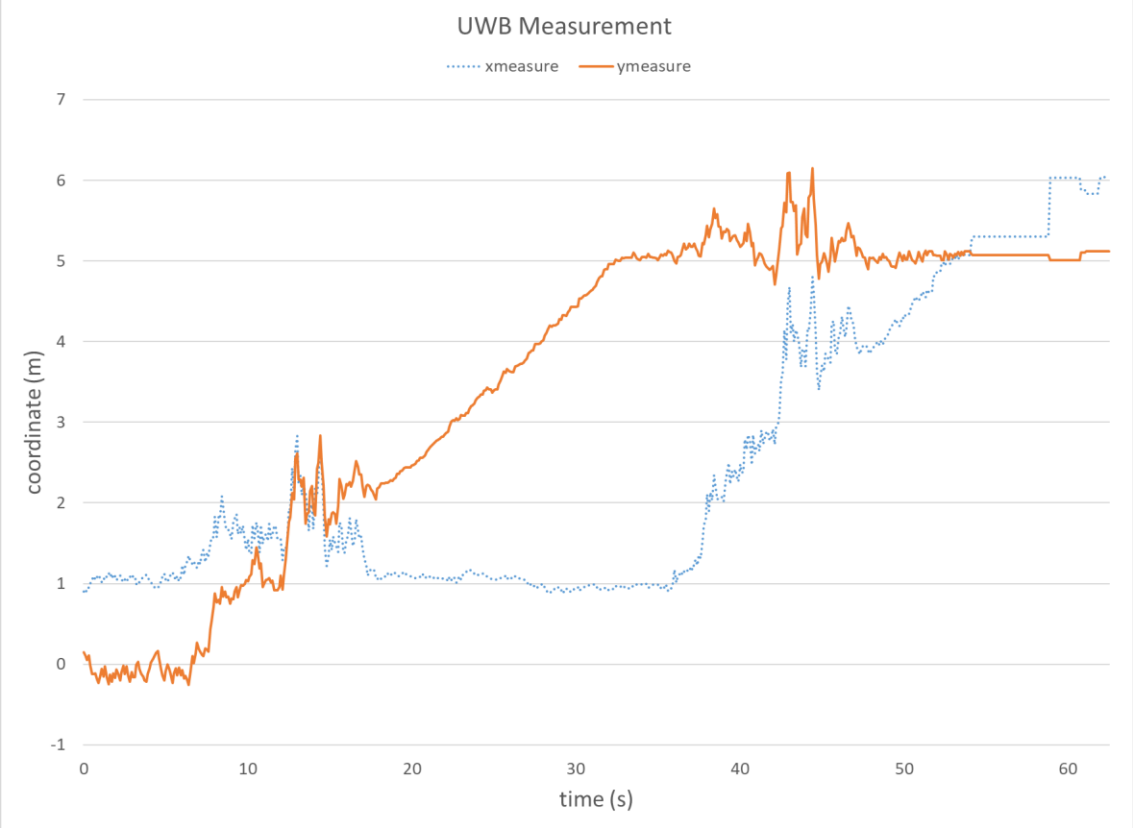

Figure 26 UWB measurement x,y coordinate

As figure 26 shows, the anomaly of UWB measurement matches the NLOS propagation detection graph. At 6s and 36s, the NLOS propagation influences the UWB measurement accuracy, so the velocity difference starts to increase. At 59s, the velocity difference increases suddenly because UWB moves close to the edge of the site, the abrupt change in UWB measurement can be observed. Overall, the detection value will increase when the UWB measurement is not accurate.

The signal to change Kalman filter parameter according to the NLOS propagation detection and the set threshold is shown below.

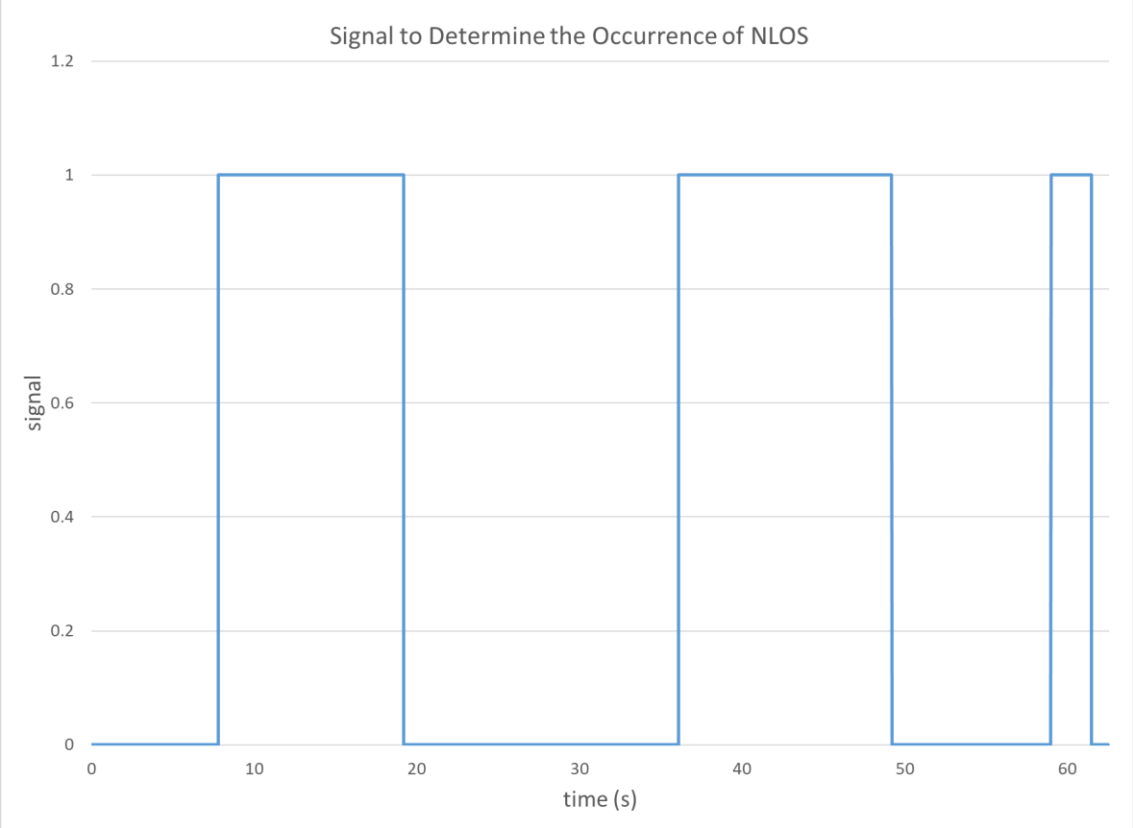

Figure 27 Signal to detect NLOS

As shown in figure 27, when the difference is larger than the threshold we set to detect NLOS propagation, the signal turns from zero to one and hold for 2.5 seconds. When the signal equals to one, the Q is turned smaller and R is turned larger.

The true trajectory, estimated result and the UWB measurement are shown below.

Figure 28 Estimated trajectory, UWB measured trajectory and true trajectory

To compare the error between the UWB measured position and the true position and the error between the estimated position and the true position, it is not possible to record the true position of all sample time, so firstly I calculate the distance from each measured point and estimated point to the true trajectory to compare the extent of deviation.

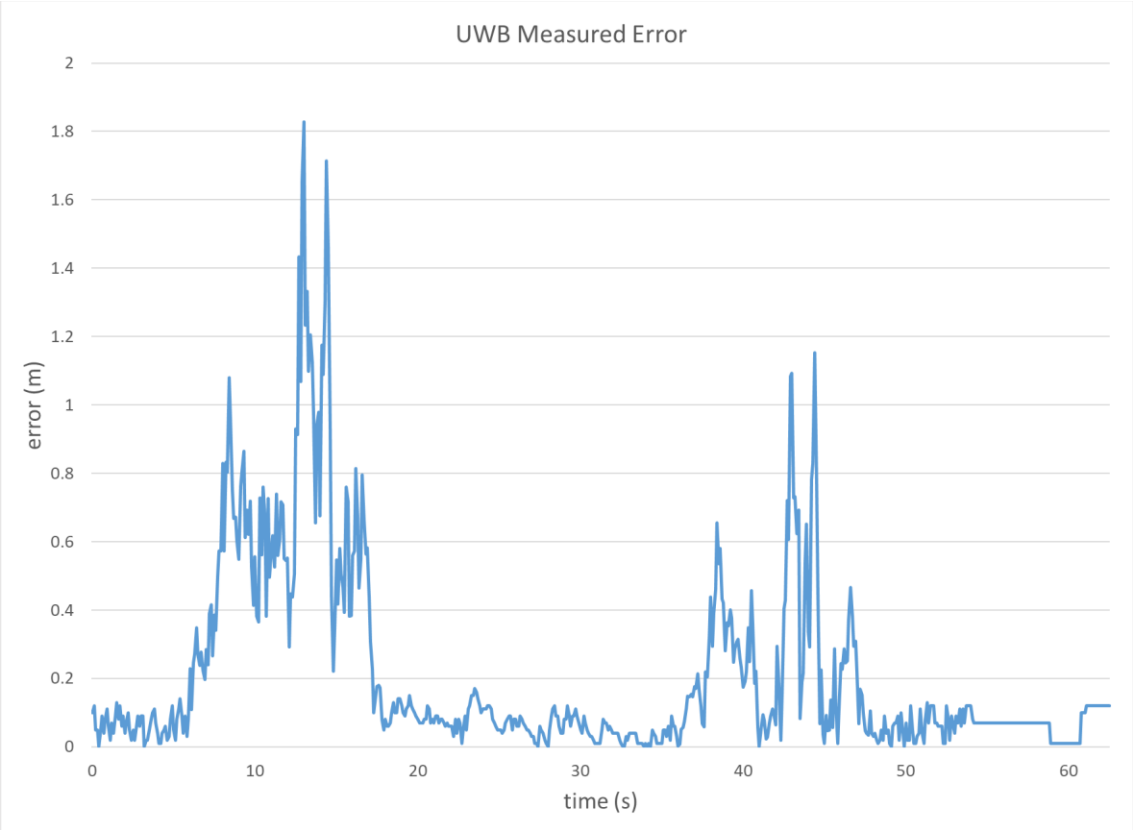

Figure 29 UWB Measured Error

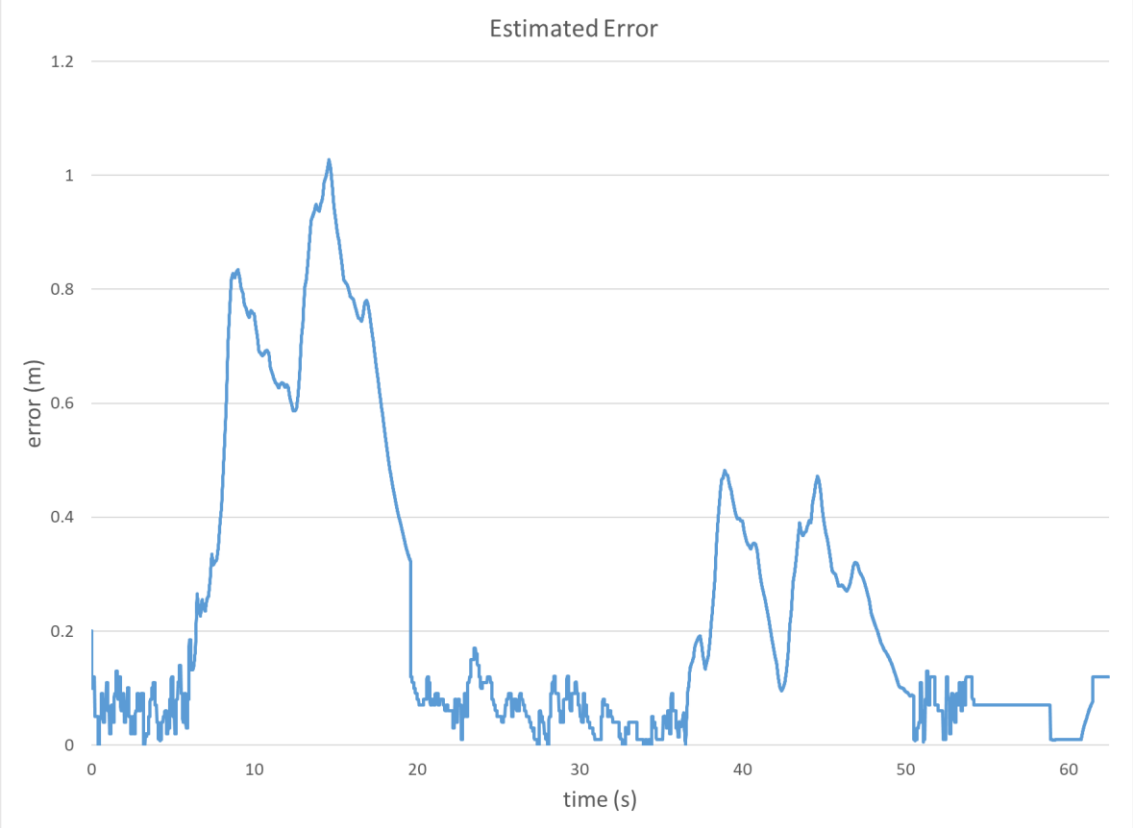

Figure 30 Esimated Error

From figure 29 and 30, the maximum distance that the UWB measurement deviates from the real trajectory is 1.83m, while the maximum distance that the estimated position deviates from the real trajectory is only 1.02m.

In addition, I use a stopwatch to record the time of the device passing some specific points. From the stopwatch, the timestamp of the device passing the certain points are recorded, and these points are compared with the estimated position and UWB measured position at the same time.

Time	True position	Estimated position	UWB measured position	Estimation error	UWB error
8.69s	(1,1)	(1.827,0.745)	(1.9188,0.710)	0.865	0.963
14.74s	(1,2)	(2.011,1.876)	(2.563,2.740)	1.019	1.729
20.11s	(1,3)	(1.071,2.469)	(1.082,2.662)	0.536	0.348
24.85s	(1,4)	(1.061,3.415)	(1.060,3.413)	0.588	0.590
30.08s	(1,5)	(0.951.4.430)	(0.961,4.440)	0.572	0.561
38.43s	(2,5)	(1.883,5.376)	(2.576,5.758)	0.394	0.952
41.96s	(3,5)	(2.618,5.161)	(2.728,4.703)	0.415	0.403
45.77s	(4,5)	(3.819,5.279)	(4.671,5.604)	0.333	0.903
50.16s	(5,5)	(4.470,5.009)	(4.472,5.010)	0.530	0.528
58.17s	(6,5)	(6.031,5.010)	(6.032,5.012)	0.033	0.034

Table 2 The estimated, measured position at certain time recorded by the stopwatch

From table 2, comparing the UWB measurement error and estimation error in 14.74 second, 38.43 second and 45.77 second, during the time NLOS propagation occurs, the estimation error is obviously less than the UWB measurement error.

The trajectory calculated only by IMU data is shown below.

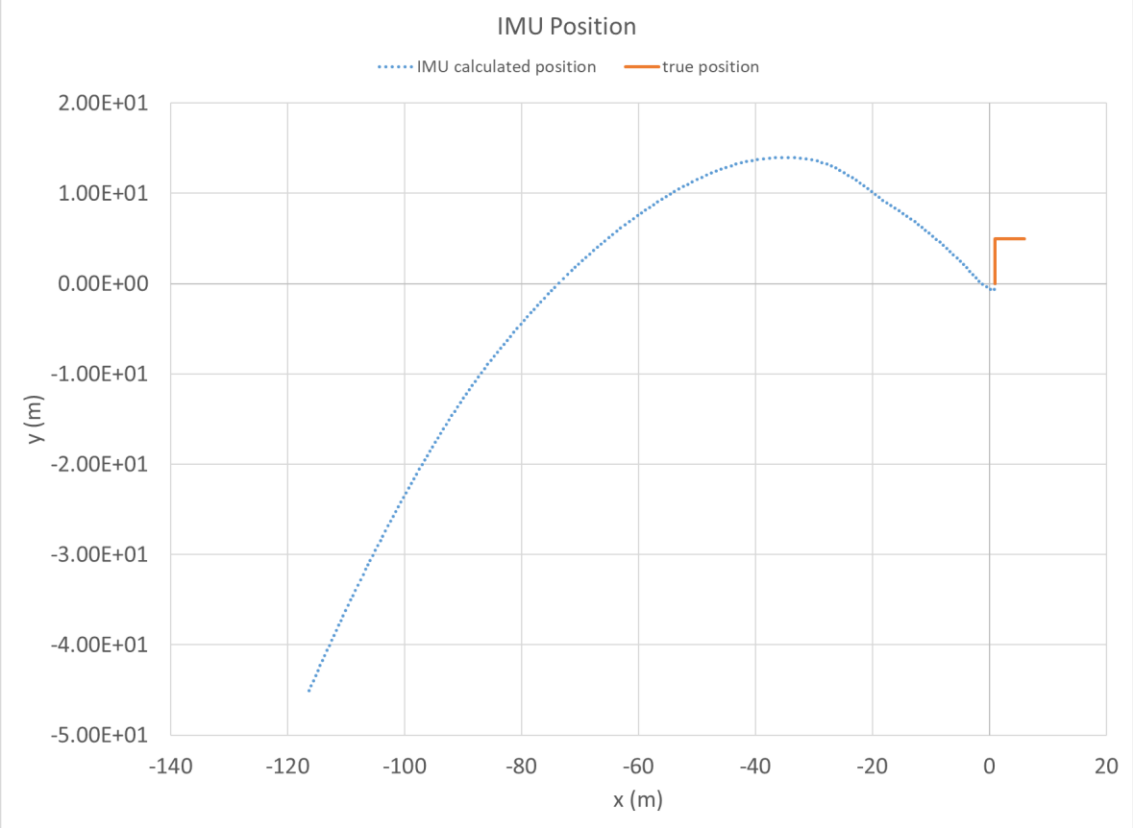

Figure 31 Trajectory calculated only by IMU

From figure 31, it can be derived that the IMU drift over long time accumulation is huge. After finishing the experiment, the error on y axis comes to about 50m even after data pre-processing, from which it can be proved that the IMU error accumulation is huge and non-negligible.

The threshold selection of the detection algorithm is important. When the threshold is too high, Kalman filter can't adjust parameters in time when error of UWB measurement is too large. The estimated position will keep following the UWB measurement rather than depending more on IMU data to calculate position. The experiment result when the threshold is too large is shown below.

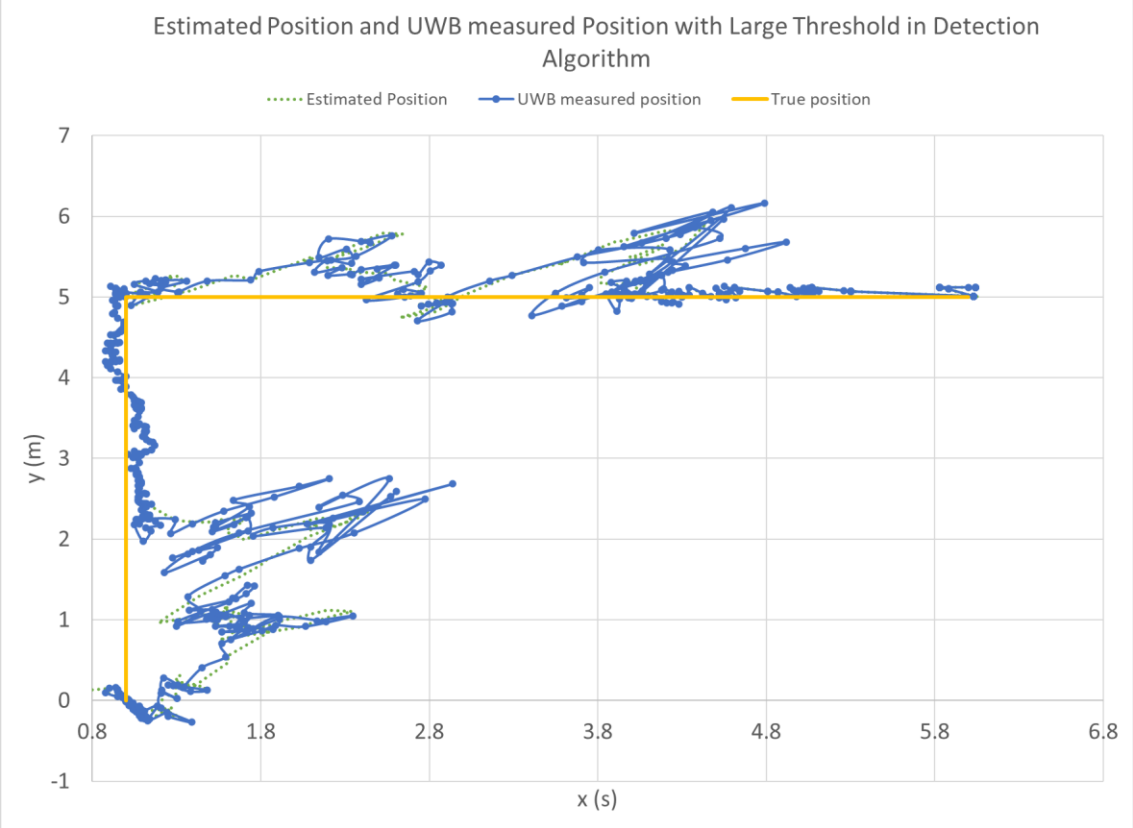

Figure 32 Estimated trajectory, UWB measured trajectory and true trajectory with large threshold

As figure 32 shows, since the threshold is too large, the algorithm becomes less sensitive to UWB measurement error and can't adjust parameters in time. The distance of deviation from the true trajectory becomes larger significantly.

On the other hand, when the velocity difference threshold is too small, which means the Kalman filter is too sensitive to UWB measurement error, the estimated position will abandon too much UWB measurement data and rely too much on IMU. As figure 34 shows, the error of position calculated only by IMU is large. Besides, every time the algorithm detects the difference of velocity between the velocity calculated by IMU data and the velocity calculated by UWB

position data exceeds the threshold, the signal will hold for the later 2.5s. So if the threshold is too small, the algorithm will be more complex and slower. The experiment result when the threshold is too small is shown below.

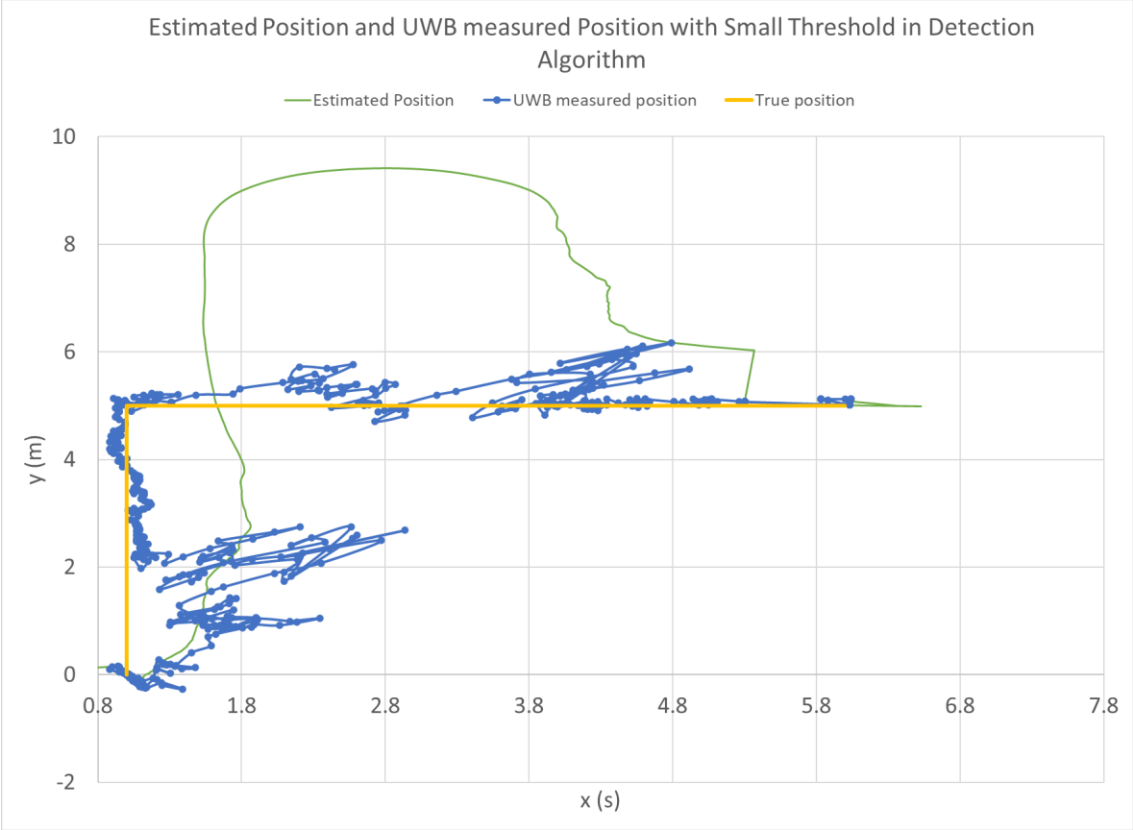

Figure 33 Estimated trajectory, UWB measured trajectory and true trajectory with small threshold

As figure 33 shows, since the threshold is too small, the localization system chooses to rely too much on IMU data to calculate position even when UWB is relatively accurate. Because IMU suffers from accumulation error, the position accuracy is low.

7. Conclusion

While UWB and IMU are both ideal techniques in localization system, non-line-of-sight propagation caused by obstacles placed between the anchor and tag causes huge error in position measurement by UWB and the inertial measurement unit suffers greatly from long-time error accumulation.

The UWB measurement anomaly detection algorithm can detect NLOS propagation and other anomalies in UWB according to the difference between the velocity calculated by UWB measurements and the velocity calculated by IMU data, then it can give a signal to adjust Kalman filter parameters. The result of the smoothness of the trajectory estimated by complementary filter corresponds with the actual situation.

The estimated position by Kalman filter combining the processed IMU and UWB data is much more accurate than the position measured or estimated by UWB or IMU individually. On the one hand, the estimated trajectory matches the true trajectory more accurately, the maximum distance between the estimated trajectory and the true trajectory is shorter than the maximum distance between the UWB measured trajectory and the true trajectory. On the other hand, comparing the true point at certain time recorded by the stopwatch with the estimated point and UWB measured points at same time, the estimated points by Kalman filter are closer to the true point obviously than the points measured only by UWB especially when NLOS occurs. The threshold selection will greatly

influence the algorithm effect and it needs to be set appropriately according to the experiment under normal condition without NLOS interference.

Overall, the algorithm can make the localization system more reliable and provide more accurate position information.

Reference

[1] Y. Li and C. Shi, "Localization and Navigation for Indoor Mobile Robot Based on ROS," 2018 Chinese Automation Congress (CAC), Xi'an, China, 2018, pp. 1135-1139, doi: 10.1109/CAC.2018.8623225.

[2] D. Zhang, F. Xia, Z. Yang, L. Yao and W. Zhao, "Localization Technologies for Indoor Human Tracking," 2010 5th International Conference on Future Information Technology, Busan, Korea (South), 2010, pp. 1-6, doi: 10.1109/FUTURETECH.2010.5482731.

[3] W. Wei, K. Kurita, J. Kuang and A. Gao, "Real-Time 3D Arm Motion Tracking Using the 6-axis IMU Sensor of a Smartwatch," 2021 IEEE 17th International Conference on Wearable and Implantable Body Sensor Networks (BSN), Athens, Greece, 2021, pp. 1-4, doi: 10.1109/BSN51625.2021.9507012.

[4] J. Guo, J. Du and D. Xu, "Navigation and Positioning System Applied in Underground Driverless Vehicle Based on IMU," 2018 International Conference on Robots & Intelligent System (ICRIS), Changsha, China, 2018, pp. 13-16, doi: 10.1109/ICRIS.2018.00012.

[5] I. Toy, A. Durdu and A. Yusefi, "Improved Dead Reckoning Localization using IMU Sensor," 2022 International Symposium on Electronics and Telecommunications (ISETC), Timisoara, Romania, 2022, pp. 1-5, doi: 10.1109/ISETC56213.2022.10010239.

[6] X. Liu, Q. Zhou, X. Chen, L. Fan and C. -T. Cheng, "Bias-Error Accumulation Analysis for Inertial Navigation Methods," in IEEE Signal Processing Letters, vol. 29, pp. 299-303, 2022, doi: 10.1109/LSP.2021.3129151.

[7] L. Chen and H. Hu, "IMU/GPS based pedestrian localization," 2012 4th Computer Science and Electronic Engineering Conference (CEEC), Colchester, UK, 2012, pp. 23-28, doi: 10.1109/CEEC.2012.6375373.

[8] R. S. Kshetrimayum, "An introduction to UWB communication systems," in IEEE Potentials, vol. 28, no. 2, pp. 9-13, March-April 2009, doi: 10.1109/MPOT.2009.931847.

[9] E. L. Mokole and T. K. Sarkar, "Introduction to ultrawideband theory/technology/systems," 2016 International Conference on Electromagnetics in Advanced Applications (ICEAA), Cairns, QLD, Australia, 2016, pp. 768-771, doi: 10.1109/ICEAA.2016.7731512.

[10] R. Mazraani, M. Saez, L. Govoni and D. Knobloch, "Experimental results of a combined TDOA/TOF technique for UWB based localization systems," 2017 IEEE International Conference on Communications Workshops (ICC Workshops), Paris, 2017, pp. 1043-1048, doi: 10.1109/ICCW.2017.7962796.

[11] L. Schmid, D. Salido-Monzú and A. Wieser, "Accuracy Assessment and Learned Error Mitigation of UWB ToF Ranging," 2019 International

Conference on Indoor Positioning and Indoor Navigation (IPIN), Pisa, Italy, 2019, pp. 1-8, doi: 10.1109/IPIN.2019.8911769.

[12] R. Ye and H. Liu, "UWB TDOA localization system: Receiver configuration analysis," 2010 International Symposium on Signals, Systems and Electronics, Nanjing, China, 2010, pp. 1-4, doi: 10.1109/ISSSE.2010.5607083.

[13] Zhi Ning Chen, "UWB antennas: Design and application," 2007 6th International Conference on Information, Communications & Signal Processing, Singapore, 2007, pp. 1-5, doi: 10.1109/ICICS.2007.4449887.

[14] Zhi Ning Chen, "UWB antennas: Design and application," 2007 6th International Conference on Information, Communications & Signal Processing, Singapore, 2007, pp. 1-5, doi: 10.1109/ICICS.2007.4449887.

[15] X. -h. Li and T. Zhang, "Research on Improved UWB Localization Algorithm in NLOS Environment," 2018 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), Xiamen, China, 2018, pp. 707-711, doi: 10.1109/ICITBS.2018.00183.

[16] Zhi Ning Chen, "UWB antennas: Design and application," 2007 6th International Conference on Information, Communications & Signal Processing, Singapore, 2007, pp. 1-5, doi: 10.1109/ICICS.2007.4449887.

[17] "IEE Colloquium on 'Kalman Filters: Introduction, Applications and Future Developments' (Digest No.27)," IEE Colloquium on Kalman Filters:

Introduction, Applications and Future Developments, London, UK, 1989, pp. 0_1-.

[18] Jianguo Yan, Dongli Yuan, Xiaojun Xing and Qiuling Jia, "Kalman filtering parameter optimization techniques based on genetic algorithm," 2008 IEEE International Conference on Automation and Logistics, Qingdao, China, 2008, pp. 1717-1720, doi: 10.1109/ICAL.2008.4636432.

[19] R. Bao and Y. Li, "Kalman filtering design for agricultural vehicle state estimation," 2011 International Conference on Electric Information and Control Engineering, Wuhan, China, 2011, pp. 4252-4255, doi: 10.1109/ICEICE.2011.5777685.

[20] L. Zhang, Z. Chen and X. Zhang, "A novel varible gain unscented kalman filter and its application in the integrated navigation system," Proceedings of the 10th World Congress on Intelligent Control and Automation, Beijing, China, 2012, pp. 1160-1165, doi: 10.1109/WCICA.2012.6358056.

[21] P. Dai, Y. Yang, C. Zhang, X. Bao, H. Zhang and Y. Zhang, "Analysis of Target Detection Based on UWB NLOS Ranging Modeling," 2018 Ubiquitous Positioning, Indoor Navigation and Location-Based Services (UPINLBS), Wuhan, China, 2018, pp. 1-6, doi: 10.1109/UPINLBS.2018.8559809.

[22] Z. Zeng, S. Liu and L. Wang, "NLOS Detection and Mitigation for UWB/IMU Fusion System Based on EKF and CIR," 2018 IEEE 18th

International Conference on Communication Technology (ICCT), Chongqing, China, 2018, pp. 376-381, doi: 10.1109/ICCT.2018.8600094.

[23] G. Xu and X. Meng, "The MEMS IMU Error Modeling Analysis Using Support Vector Machines," 2009 Second International Symposium on Knowledge Acquisition and Modeling, Wuhan, China, 2009, pp. 335-337, doi: 10.1109/KAM.2009.287.

[24] S. Guan and X. Luo, "Fusing Ultra-wideband Range Measurements with IMU for Mobile Robot Localization," 2021 11th International Conference on Intelligent Control and Information Processing (ICICIP), Dali, China, 2021, pp. 107-111, doi: 10.1109/ICICIP53388.2021.9642157.

[25] A. Marquez, B. Tank, S. K. Meghani, S. Ahmed and K. Tepe, "Accurate UWB and IMU based indoor localization for autonomous robots," 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), Windsor, ON, Canada, 2017, pp. 1-4, doi: 10.1109/CCECE.2017.7946751.

[26] X. Yue et al., "Research on Fusion and Location Method of Indoor Targeted Spraying Robots Based on UWB\IMU," 2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 2022, pp. 1487-1492, doi: 10.1109/ITOEC53115.2022.9734704.

[27] F. Wu and Z. Liu, "Research on UWB / IMU Fusion Positioning Technology in Mine," 2020 International Conference on Intelligent

Transportation, Big Data & Smart City (ICITBS), Vientiane, Laos, 2020, pp. 934-937, doi: 10.1109/ICITBS49701.2020.00207.

[28] H. Benzerrouk and A. V. Nebylov, "Robust IMU/UWB integration for indoor pedestrian navigation," 2018 25th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS), St. Petersburg, Russia, 2018, pp. 1-5, doi: 10.23919/ICINS.2018.8405844.

[29] S. Zheng, Z. Li, Y. Liu, H. Zhang and X. Zou, "An Optimization-Based UWB-IMU Fusion Framework for UGV," in IEEE Sensors Journal, vol. 22, no. 5, pp. 4369-4377, 1 March1, 2022, doi: 10.1109/JSEN.2022.3144660.