Sensor Fusion-Based Indoor Navigation in GNSS-Denied Environments

by

Ali SHAKERIAN

MANUSCRIPT-BASED
THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE IN
PARTIAL FULFILLEMENT OF THE REQUIREMENTS FOR A MASTER'S
DEGREE WITH THESIS IN ELECTRICAL ENGINEERING
M.A.Sc.

MONTREAL, FEBRUARY 13, 2024

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

Ali Shakerian, 2024

This Creative Commons licence allows readers to download this work and share it with others as long as the author is credited. The content of this work can’t be modified in any way or used commercially.

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

René Jr Landry, Thesis Supervisor
Electrical Engineering at École de technologie supérieure

Wael Jaafar, President of the Board of Examiners
Software Engineering and IT at École de technologie supérieure

Ricardo Izquierdo, Member of the jury
Electrical Engineering at École de technologie supérieure

THIS THESIS WAS PRENSENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND PUBLIC

FEBRUARY 5, 2024

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

FOREWORD

The pursuit of knowledge often leads researchers to delve into complex challenges, seeking innovative solutions that might benefit the broader community. It was with this objective that I commenced my Master's degree at the LASSENA Lab, being introduced to the already burgeoning IBNAV project.

The IBNAV project, initiated by earlier LASSENA Lab students and researchers, aimed to address the challenges of indoor navigation, especially in environments where GPS signals are compromised or unavailable. The timeliness of such an endeavor is evident. In recent years, the significance of indoor navigation has rapidly expanded, catering not just to individuals in intricate indoor settings, but also playing a pivotal role in drone navigation. With the increasing integration of drones into commercial sectors and the emergence of augmented reality, the demand for precise and reliable indoor navigation systems is expected to grow exponentially.

This thesis encompasses my efforts to enhance the IBNAV project. While inheriting a solid foundation, the ambition was clear: to augment its performance, refine its functionalities, and ensure it keeps pace with current technological demands. The project's structure, spanning hardware design, software development, communication protocols, and advanced navigation algorithms, presented a multifaceted challenge.

The journey, however, was not without its trials. Among the myriad challenges were issues of drift, the urgency of real-time responses, ensuring seamless communication, and the overarching aim of translating this research into a feasible commercial product. Each hurdle, though daunting, offered an invaluable learning experience, pushing the boundaries of my understanding and skills.

I owe a deep debt of gratitude to my supervisor whose expertise and guidance were instrumental in navigating through these challenges. The foundational work and experiences shared by preceding students of the project were crucial in shaping my approach and understanding of the broader context.

In this thesis, I present the culmination of rigorous research, iterative problem-solving, and unwavering perseverance. It is my hope that the insights and advancements documented herein contribute significantly to the field of indoor navigation and serve as a reference for future researchers and practitioners in the domain. I invite you to delve into the pages that follow, to explore the intricacies of the IBNAV project, and to join in the vision of a future where indoor navigation systems, like the one detailed here, redefine our interaction with both the physical and digital realms.

ACKNOWLEDGMENTS

First and foremost, I wish to express my profound gratitude to my supervisor, Professor René Jr Landry. His unwavering support, invaluable guidance, and insightful feedback have been instrumental throughout my research journey. His expertise and mentorship not only enriched this thesis but also significantly shaped my academic and professional growth.

To my parents, whose unwavering belief in my capabilities served as a beacon of motivation and strength, I offer my heartfelt thanks. Their love, encouragement, and sacrifices have been the bedrock upon which my academic pursuits have been built.

I would also like to extend my appreciation to my colleagues at LASSENA Lab. Collaborating with and learning from them has been both an honor and a pleasure. Their dedication, shared knowledge, and camaraderie greatly enhanced my research experience.

Lastly, to my friends, who have stood by me through every high and low, offering encouragement, understanding, and much-needed respite, I am deeply thankful. Their support, both academic and personal, has been invaluable in bringing this thesis to fruition. This work is not just the product of solitary effort but is shaped and made richer by the contributions, faith, and support of all these individuals.

Navigation en intérieur basée sur la fusion de capteurs dans des environnements dépourvus de gnss

Ali SHAKERIAN

RÉSUMÉ

La navigation en intérieur revêt une importance primordiale dans le paysage technologique en constante évolution d'aujourd'hui, en particulier dans des environnements où les méthodes conventionnelles telles que le Système de Navigation par Satellite Global (GNSS) ne sont pas opérationnelles. Le projet ibNav représente une initiative révolutionnaire dans le domaine des systèmes de navigation en intérieur, répondant à l'évolution du paysage technologique où les systèmes de navigation par satellite global traditionnels (GNSS) atteignent souvent leurs limites. Cette thèse de master plonge dans les développements complexes au sein du projet ibNav, aboutissant à une exploration exhaustive de ses principes fondamentaux, de ses méthodologies de calibrage innovantes et de ses applications potentielles dans le monde réel. Au cœur de cette recherche se trouve le perfectionnement de la version ibNav 6.1, impliquant un débogage minutieux et l'introduction d'algorithmes avancés de traitement des données, garantissant la robustesse opérationnelle du système. Tout aussi essentielle était la gestion de la Station de Contrôle au Sol (GCS) pour la visualisation des données et la cartographie des trajectoires, ainsi que l'optimisation du serveur RiCF pour faciliter la communication fluide avec l'ensemble des capteurs intégrés. Des efforts méticuleux ont également été dirigés vers le calibrage des capteurs, garantissant des normes de précision élevées, voire une amélioration de celles-ci. La thèse éclaire la contribution essentielle du projet à la résolution des problèmes liés à l'absence de GNSS, en mettant l'accent sur l'intégration de la technologie de la blockchain pour renforcer l'intégrité et la sécurité des données au sein du système ibNav. De plus, elle explore l'application novatrice de l'algorithme de l'ajustement ellipsoïdal des moindres carrés pour calibrer le cadre des capteurs du projet, améliorant considérablement la précision. Une approche interdisciplinaire est introduite, combinant des données de capteurs en temps réel avec des réseaux neuronaux, démontrant le potentiel non seulement pour la reconnaissance de l'activité humaine, mais aussi pour élever l'adaptabilité et la polyvalence des systèmes comme ibNav. Cette exploration approfondie des composants essentiels d'ibNav forme la base du développement de trois articles de recherche distincts, chacun abordant une dimension unique du projet et fournissant collectivement une compréhension holistique des principes d'ibNav, des techniques de calibrage et du potentiel d'application. Tout au long de cette thèse, l'interrelation entre ces articles devient évidente, formant un récit cohérent qui commence par les principes fondamentaux du système ibNav, plonge dans les subtilités du calibrage des capteurs et aboutit à l'exploration des applications dans le monde réel grâce à l'intégration des données avec les réseaux neuronaux. En conclusion, cette thèse constitue une exploration approfondie du projet ibNav, comblant le fossé entre ses principes fondamentaux, ses méthodologies de calibrage avancées et ses implications prometteuses dans le domaine plus vaste des systèmes de navigation en intérieur. Au fur et à mesure que les chapitres suivants se dévoileront, ils éclaireront davantage l'importance de ces contributions et leur potentiel pour révolutionner la technologie de navigation en intérieur.

Mots-clés: navigation, gnss refusé, unité de mesure inertielle (umi), positionnement, apprentissage automatique, calibration, cartographie

Sensor fusion-based indoor navigation in gnss-denied environments

Ali SHAKERIAN

ABSTRACT

Indoor navigation holds paramount importance in today's ever-evolving technological landscape, particularly in environments where conventional methods such as the Global Navigation Satellite System (GNSS) are non-operational. The ibNav project represents a groundbreaking endeavor in the realm of indoor navigation systems, responding to the evolving landscape of technology where traditional Global Navigation Satellite Systems (GNSS) often fall short. This Master's thesis delves into the intricate developments within the ibNav project, culminating in a comprehensive exploration of its foundational principles, innovative calibration methodologies, and potential real-world applications. Central to this research is the refinement of the ibNav 6.1 version, involving exhaustive debugging and the introduction of advanced data processing algorithms, ensuring the system's robust operationality. Equally critical was the management of the Ground Control Station (GCS) for data visualization and trajectory mapping, as well as the optimization of the RiCF server to facilitate seamless communication with the array of integrated sensors. Meticulous efforts were also directed towards sensor calibration, guaranteeing high accuracy standards, if not enhancement. The thesis elucidates the project's core contribution in addressing GNSS-denied environments, emphasizing the integration of blockchain technology to fortify data integrity and security within the ibNav system. Furthermore, it explores the novel application of the Least Square Ellipsoid Fitting Algorithm to calibrate the project's sensor framework, enhancing precision significantly. An interdisciplinary approach is introduced, merging real-time sensor data with neural networks, showcasing the potential not only for human activity recognition but also for elevating the adaptability and versatility of systems like ibNav. This comprehensive exploration of ibNav's core components forms the foundation for the development of three distinct research papers, each addressing a unique dimension of the project and collectively providing a holistic understanding of ibNav's principles, calibration techniques, and application potential. Throughout this thesis, the interrelation between these papers becomes evident, forming a cohesive narrative that begins with the foundational principles of the ibNav system, delves into the intricacies of sensor calibration, and culminates in the exploration of real-world applications through data integration with neural networks. In conclusion, this thesis serves as an in-depth exploration of the ibNav project, bridging the gap between its foundational principles, advanced calibration methodologies, and its promising implications within the broader field of indoor navigation systems. As the subsequent chapters unfold, they will further elucidate the significance of these contributions and their potential to revolutionize indoor navigation technology.

Keywords: navigation, gnss-denied, inertial measurement unit (imu), positioning, machine learning, calibration, mapping

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF ABREVIATIONS AND ACRONYMS

AUV	Autonomous Underwater Vehicle
AI	Artificial Intelligence
API	Application Programming Interface
AR	Augmented Reality
BLE	Bluetooth Low Energy
CAPSLSTM	Capsule Long Short-Term Memory
CLI	Command Line Interface
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CSV	Comma-Separated Values
DL	Deep Learning
EKF	Extended Kalman Filter
ECDSA	Elliptic Curve Digital Signature Algorithm
ESP32	Espressif System's Programmable System-on-Chip 32
ESP8266	A low-cost WiFi module
FPGA	Field-Programmable Gate Array
GCS	Ground Control Station
GPU	Graphics Processing Unit
GUI	Graphical User Interface
GNSS	Global Navigation Satellite System
HAR	Human Activity Recognition

HTTP/2	Hypertext Transfer Protocol version 2
IDF	IoT Development Framework
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IP	Internet Protocol
IPFS	InterPlanetary File System
ISO	International Organization for Standardization
JWT	JSON Web Tokens
LBS	Location-Based Services
LED	Light Emitting Diode
LiDAR	Light Detection and Ranging
LRP	Learning Rate
MARG	Magnetic Angular Rate and Gravity
MB	Megabyte
MEMS	Micro Electromechanical Systems
MQTT	Message Queuing Telemetry Transport
OS	Operating System
PDR	Pedestrian Dead Reckoning
PCB	Printed Circuit Board
POL	Proof Of Location
PNT	Positioning Navigation and Timing
PSO	Particle Swarm Optimization

RAD/s	Radians per second
RAM	Random Access Memory
ReLU	Rectified Linear Unit
RFID	Radio-Frequency Identification
RGB	Red Green Blue
RMSProp	Root Mean Square Propagation
RMSE	Root Mean Square Error
RPC	Remote Procedure Call
RSSI	Received Signal Strength Indicator
RSS	Received Signal Strength
SLAM	Simultaneous Localization and Mapping
SVM	Support Vector Machines
TPU	Tensor Processing Unit
UAV	Unmanned Aerial Vehicle
UART	Universal Asynchronous Receiver/Transmitter
UE	User Equipment
UWB	Ultra-Wideband
USB	Universal Serial Bus
WiFi	Wireless Fidelity
WLAN	Wireless Local Area Network
ZUPT	Zero-Velocity Update
1D	One-Dimensional

2D	Two-Dimensional
3D	Three-Dimensional

INTRODUCTION

The ever-evolving landscape of technology necessitates the development of efficient indoor navigation systems, especially in circumstances where conventional methods like the Global Navigation Satellite System (GNSS) are non-operational. The ibNav project emerges within this framework, aiming to offer precise navigation solutions in indoor settings through an intricate assembly of sensors and embedded systems.

The pivotal work undertaken in the scope of this Master's thesis focuses on various enhancements and critical innovations within the ibNav project. Key among these was the role played in refining the ibNav 6.1 version, which encompassed a comprehensive set of tasks: from rigorous debugging to the introduction of advanced data processing algorithms. Another significant element of this research was the Ground Control Station (GCS), requiring adept management of data visualization and trajectory mapping. Equally challenging was the optimization of the RiCF server, responsible for ensuring smooth communication between the ibNav system and its array of sensors. In addition, meticulous efforts were directed towards sensor calibration processes, ensuring that the system maintained, if not elevated, its accuracy standards. This overarching approach not only ensured the rigorous operationality of the system but also established a solid foundation for future research endeavours.

The foundational and applied work on the ibNav project laid the groundwork for the development of three research papers, each addressing a unique dimension of the project:

- "**A Secure ZUPT-Aided Indoor Navigation System Using Blockchain in GNSS-Denied Environments**": This paper encapsulates the core elements of the ibNav system, focusing on its operation within GNSS-denied arenas. Notably, it delves into the integration of blockchain technology, highlighting its significance in bolstering data integrity and security in such navigation systems.
- "**Calibration of 3-Axis Low-Cost Magnetometer Using the Least Square Ellipsoid Fitting Algorithm**": A deeper dive into the calibration intricacies of the ibNav's sensor

framework is presented here. It elucidates the innovative use of the Least Square Ellipsoid Fitting Algorithm, showcasing its efficacy in enhancing the platform's precision.

- "**Real-Time Sensor-Embedded Neural Network for Human Activity Recognition**": This research introduces an interdisciplinary approach, merging real-time sensor data with neural networks. By doing so, it underscores the potential of such an integration, not only for human activity recognition but also for enhancing the adaptability and versatility of systems like ibNav.

The interrelation of these papers isn't incidental. Instead, they collectively provide a comprehensive understanding of the ibNav system, starting from its foundational principles (as seen in the first paper) to its intricate calibration techniques (highlighted in the second paper) and culminating in its application potential, merging data and neural networks (demonstrated in the third paper).

To conclude, this thesis endeavors to provide an in-depth exploration into the ibNav project, bridging the gaps between foundational principles, intricate calibration methodologies, and potential real-world applications. As the subsequent chapters unfold, they will further elucidate the significance of the contributions made and their implications in the broader realm of indoor navigation systems.

CHAPTER 1

IBNAV OVERVIEW

1.1 Introduction to ibNav

In the contemporary military landscape, advances in technology have shaped strategies, operations, and capabilities. Among these technological domains, the realm of positioning, navigation, and timing (PNT) stands paramount. The increasing awareness of vulnerabilities associated with Global Positioning Systems (GPS) in adversarial scenarios necessitates innovative solutions to counter potential signal degradation or outright denial. Recognizing these challenges, research initiatives have been undertaken to develop alternative systems that can operate efficiently in the absence of traditional GPS signals. One such prominent project aims to ensure military personnel are equipped with an advanced system offering precise 3D positioning and real-time environmental mapping, regardless of GPS reliability or availability. A set of rigorous performance benchmarks characterize this project's objectives. Primarily, the envisaged system is designed to operate autonomously from satellite-based navigation for a minimum duration of one hour. Within this operative window, the system is mandated to maintain a navigation error margin below 30 meters and a timing discrepancy under one microsecond. While these represent foundational benchmarks, the overarching ambition extends to achieving an indefinite operation, mirroring the precision and reliability of conventional "GPS-like" frameworks. The design philosophy of the developed equipment adheres firmly to the principles of the SWaP-C (Size, Weight, Power, and Cost) paradigm. Such an approach ensures not only global operability across diverse weather conditions and terrains but also optimizes for military portability and logistical feasibility. A distinctive feature of this system is its ability to initiate operations using pre-mapped locations. Following this initialization phase, the system showcases an impressive degree of operational independence, facilitating seamless contextual transitions without the need for manual operator intervention. An added layer of sophistication is its capacity for ad hoc environmental mapping, reinforcing its utility in dynamic combat scenarios. Incorporating cutting-edge technologies, research teams have emphasized a modular and adaptive design framework. Central to this innovative approach is the integration of inertial-based solutions with modern LiDAR and camera technologies, crafting a comprehensive PNT solution that holds the potential to redefine military positioning paradigms. In summation, this project stands as a testament to the military's foresight, exemplifying proactive innovation in anticipation of evolving challenges. By advancing solutions that promise to revolutionize positioning and navigation, the stage is set to redefine warfare paradigms for the modern era.

1.2 Ibnav Hardware Components

This section delineates the various components and hardware employed in the project, elucidating their specific roles and functionalities within the system. In the intricate fabric of this research endeavor, an array of distinct hardware and components have been meticulously integrated, encompassing elements such as the ibnav 6.1 module, LiDAR systems, camera modules, the Raspberry Pi microcontroller, and a dedicated timing board. Each of these constituents plays a pivotal role, reflecting tailored functionalities designed to serve specific objectives within the project's scope. A comprehensive overview of the architectural blueprint of the ibnav initiative can be discerned from Figure 1.1.

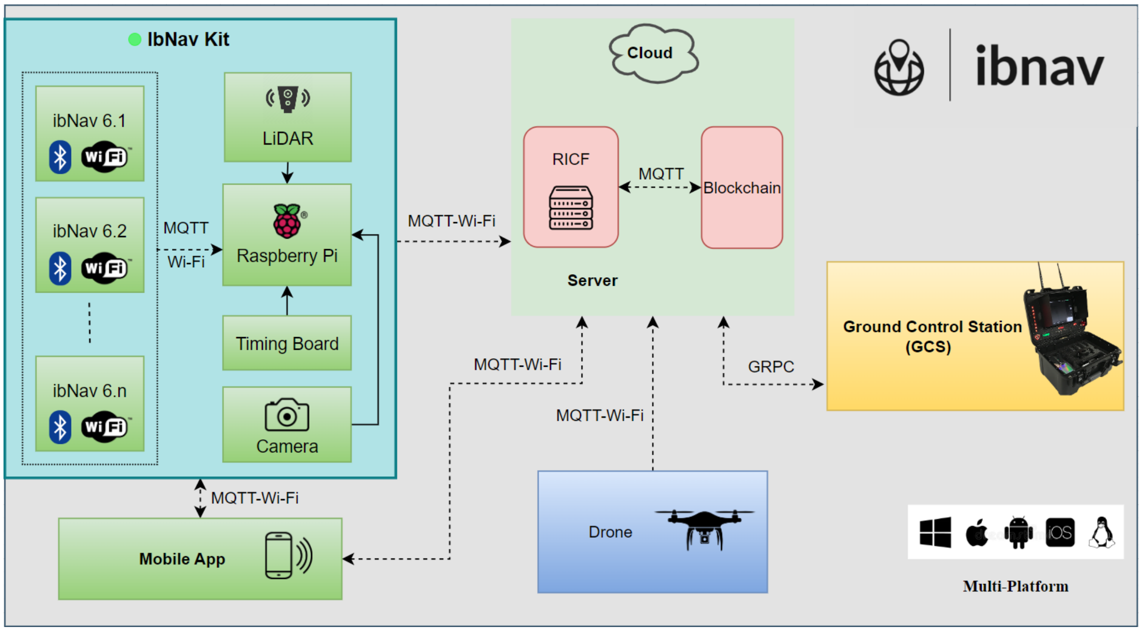


Figure 1.1 General Architecture of ibNav

1.2.1 IbNav 6.1 Sensor

The ibNav 6.1 is a state-of-the-art motion capture sensor system, detailed in a comprehensive hardware design report. This recent version exhibits multiple aspects of its design, emphasizing its advanced nature and precision enhancements over its predecessor, IbNav 6.0. Notably, several key components such as the EEPROM memory, UART to USB bridge, Bus Multiplexer, and the SD Card circuits have been retained from the previous iteration, showcasing their reliability and efficiency in the system. However, with a view to improve functionality and ease of use, the GPIO expander in version 6.1 has undergone modifications. These alterations are primarily to integrate control mechanisms for RGB LEDs and to introduce a more streamlined approach in the routing process. Figure 1.2 illustrates the Front and Back of the ibNva 6.1 PCB design.

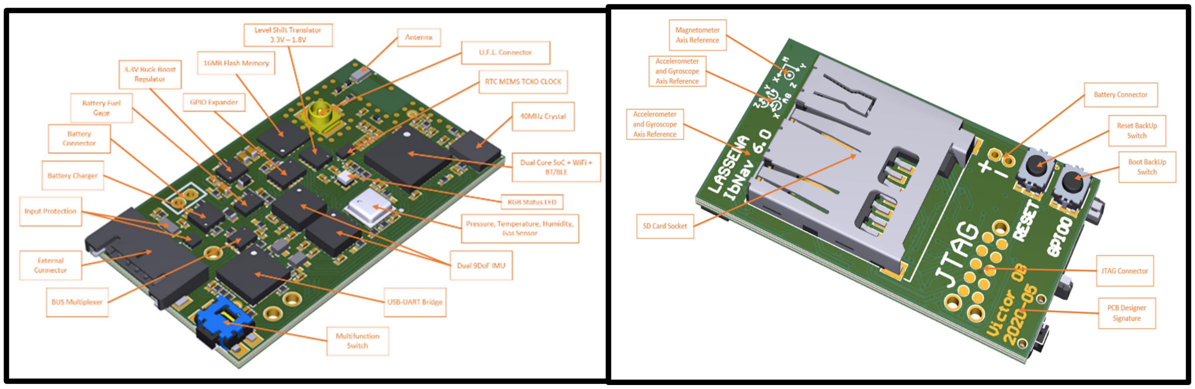


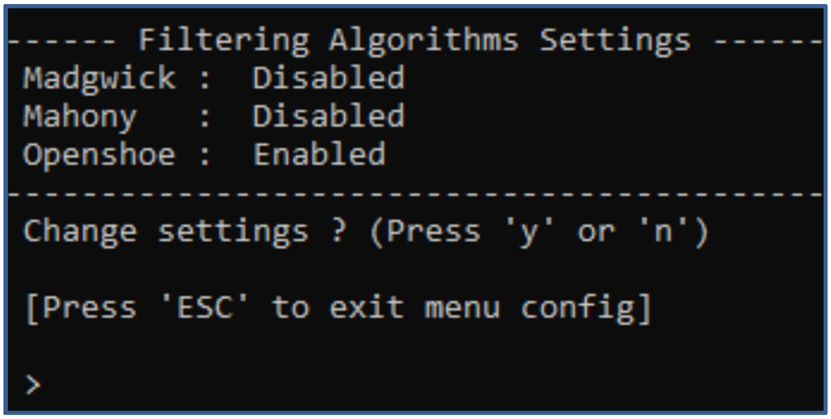
Figure 1.2 PCB Design of ibNav 6.1 Sensor

1.2.1.1 Implemented Algorithms

The IbNav sensor is an intricate piece of hardware that incorporates the renowned ESP32 chip. This chip has been programmed to execute three distinct algorithms essential for advanced motion tracking and navigation: Madgwick, Mahony, and OpenShoe as seen in Figure 1.3. Madgwick's algorithm, developed by Sebastian Madgwick, is an innovative sensor fusion method primarily used in the realm of Inertial Measurement Units (IMUs). It synergistically combines readings from accelerometers, gyroscopes, and magnetometers to derive accurate orientation estimates. One of the hallmarks of this algorithm is its computational efficiency, which is crucial for real-time applications where resources are constrained.

The Mahony algorithm, often paralleled with Madgwick's method, is another Quaternion-based complementary filter technique designed to integrate data from IMUs or MARG (Magnetic, Angular Rate, and Gravity) sensors. While both algorithms aim at the same end goal, the distinctions between them predominantly pertain to their internal computational methodologies and parameter tuning. This has led to nuanced preferences among developers, contingent on the specifics of the application at hand.

On the other hand, the OpenShoe algorithm is deeply entrenched in the domain of inertial navigation. Its primary objective is to deduce the position and orientation of a device, often utilized in footwear applications, using only data from inertial sensors. The algorithm proves invaluable in scenarios where traditional positional references, such as GPS, are either unreliable or altogether unavailable. In essence, the ESP32 chip within the IbNav sensor, through its integration of these three algorithms, stands as a testimony to the advancements in the field of motion tracking and sensor fusion.



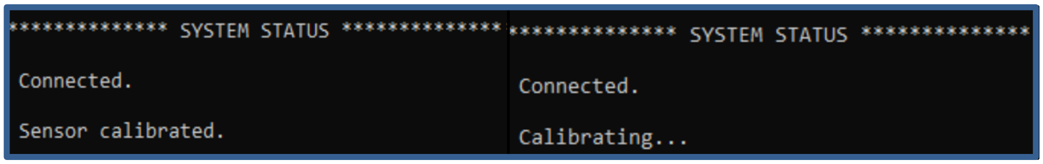
```
------ Filtering Algorithms Settings ------
 Madgwick :  Disabled
 Mahony   :  Disabled
 Openshoe :  Enabled
-------------------------------------------
 Change settings ? (Press 'y' or 'n')

 [Press 'ESC' to exit menu config]

 >
```

Figure 1.3 Terminal view of algorithm Selection during the monitoring process

Based on the specific requirements of the test, whether for motion capture or navigation, one of the aforementioned algorithms can be selected to operate as the optimal choice.

1.2.1.2 Calibration

In the pursuit of achieving accurate motion capture and navigation data, the calibration of the sensor used in this study becomes paramount. The sensor houses a combination of a magnetometer, gyroscope, barometer, and accelerometer. To initiate any experimental test, a task-specific program is meticulously crafted and then transferred to the ESP32 chip utilizing the Espressif IDF.



```
************** SYSTEM STATUS ************** ************** SYSTEM STATUS **************
Connected.                                  Connected.
Sensor calibrated.                          Calibrating...
```

Figure 1.4 Specifications of the ICM-20948

Following this initialization, the sensor undergoes a rigorous calibration process as shown in Figure 1.4. It is imperative to ensure the precision of each component within the sensor for the integrity of the resultant data. Upon the successful culmination of this calibration, the sensor seamlessly connects to the Raspberry Pi's Wi-Fi network to facilitate further operations. It's worth noting that the calibration technique, especially of the magnetometer, is of significant interest. Detailed insights into this process can be found in Chapter 3, which features the paper titled 'Calibration of 3-Axis Low-Cost Magnetometer Using the Least Square Ellipsoid Fitting Algorithm.' This contribution elucidates the nuances of the calibration algorithm and the underlying principles ensuring its robustness.

1.2.1.3 Connection and Test

Upon successful completion of the sensor calibration, an integral feature designed to augment user experience and ensure consistent operation is activated. This involves the real-time monitoring of the battery's state-of-charge (SOC) which is visually represented using an LED indicator. This LED undergoes periodic updates at 30-second intervals, serving as a dynamic gauge for the battery's health. A green illumination of the LED (as shown in Figure 1.5) signifies a battery SOC above 50%, symbolizing optimal operation. Should the SOC hover between 50% and 25%, the LED transitions to a yellow color (refer to Figure 33), indicating a moderate battery level and potential requirement for charging in the near future. When the battery SOC diminishes to below 25%, the LED adopts a red hue (depicted in Figure 1.5), alerting the user to the critical battery level and the imminent need for recharging.

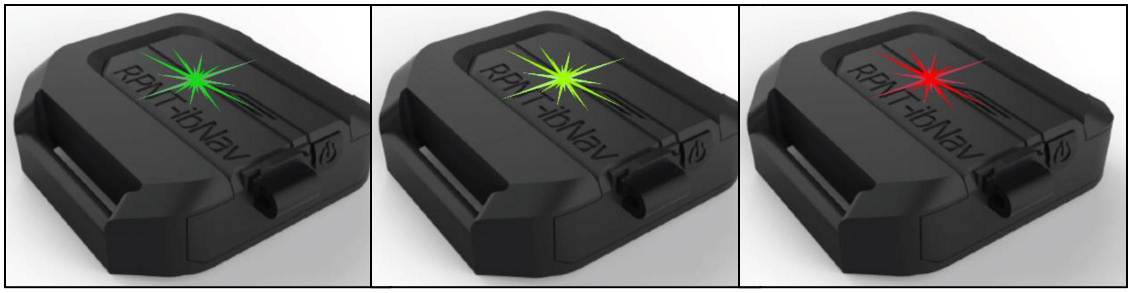


Figure 1.5 LED status of different battery levels of ibNav 6.1

Following the meticulous calibration of the sensor, its applications bifurcate into two distinct yet valuable domains. The primary application is oriented towards biomechanics and movement analysis. In this realm, the sensor is strategically affixed to various anatomical landmarks on the human body as shown in Figure 1.6, each accompanied by unique tags. This configuration facilitates comprehensive body motion capture, enabling users to derive intricate kinematic data and discern subtle nuances in human movement. The secondary application harnesses the principles of inertial navigation. By affixing the sensor to a user's foot, the Openshoe algorithm is employed, transforming the unit into a potent tool for navigation. This method capitalizes on the intricacies of foot dynamics, leveraging stride-based computations to deliver accurate navigation solutions, especially valuable in environments where traditional GPS-based navigation might be compromised or unavailable.

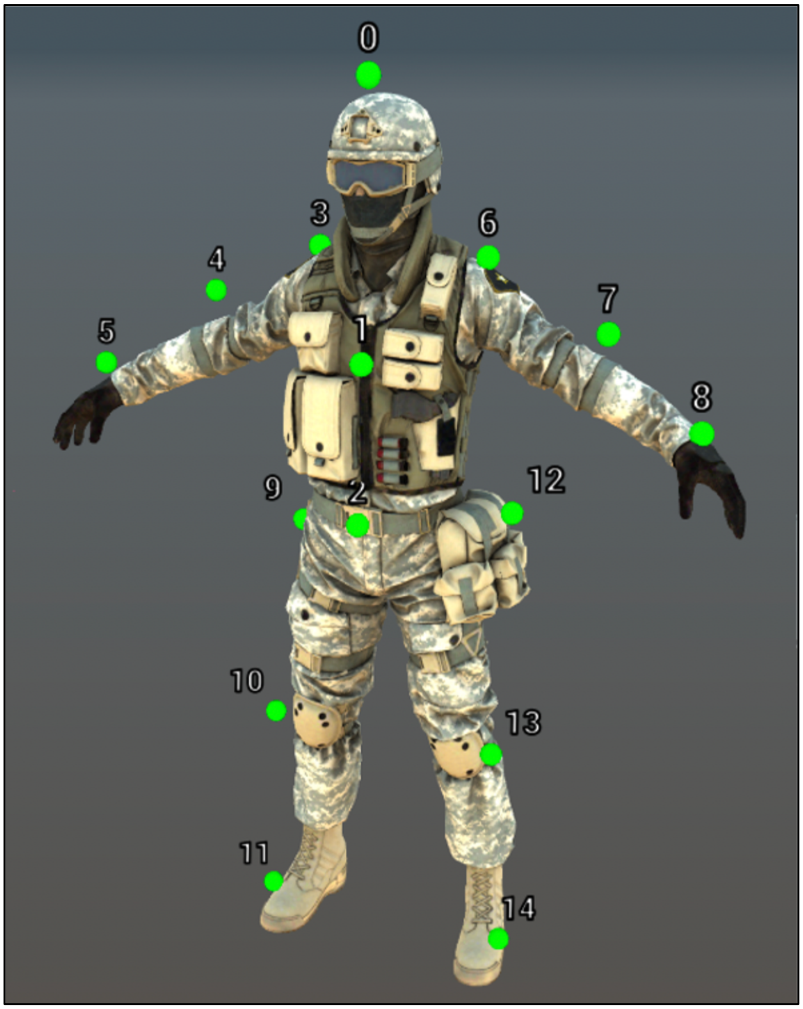


Figure 1.6 Placement location of ibNav 6.1 sensors
on the body

1.2.2 Light Detection and Ranging (LiDAR)

LiDAR, an acronym for Light Detection and Ranging, represents a cutting-edge remote sensing methodology that employs pulsed laser technology to ascertain distances by measuring the time it takes for emitted light to return after reflecting off an object. This procedure generates a dense collection of spatial data points, which, when analyzed, can produce intricate three-dimensional representations and high-definition maps. Such precision in measurement positions LiDAR at an advantage relative to many traditional measurement modalities.

Within the scope of this study, the application of LiDAR is multifaceted, encompassing areas such as intricate spatial mapping and the fusion of sensors for enhanced data interpretation. A notable methodology involves the amalgamation of data procured from the Openshoe and ZUPT algorithms with the data sets retrieved from the LiDAR devices. This synthesis is aimed at augmenting the veracity and robustness of navigational data across varying operational scenarios.

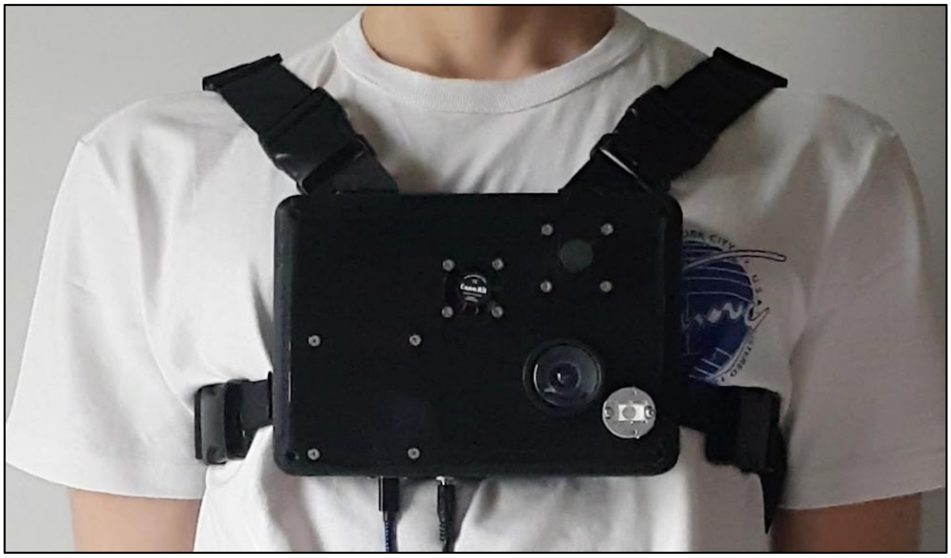


Figure 1.7 Chest-Mounted ibNav kit

Beyond its primary mapping capabilities, the data extracted from LiDAR systems is relayed to a dedicated server and pertinent applications to facilitate real-time analytical tasks. This data is subjected to advanced computational processes, prominently Simultaneous Localization and Mapping (SLAM), as well as both two-dimensional and three-dimensional spatial rendering. This research incorporated two specific LiDAR systems for its pursuits: the Leddartech M16 LSR (Figure 1.9), distinguished by its 16-segment array, and the VU8 (Figure 1.10), which features an 8-segment configuration. These LiDAR sensors has been used in the ibNav kit along other hardware used in the ibNav kit. The decision to deploy either of these systems is contingent upon specific project requirements, balancing fiscal constraints against the level of accuracy mandated by the research objectives. A detailed comparison of these two LiDARs can be found in Table 1.1.

Table 1.1 Comparison of M16 vs VU8

Specification	LeddarTech M16 LSR	LeddarTech VU8
Number of Segments	16	8
Wavelength (nm)	905	905
Power supply (VDC)	12 to 30	12 ± 0.6
Interfaces	USB, RS-485, CAN, UART	USB, RS-485, CAN, UART
Accuracy (cm)	±5	±5
Data refresh rate (Hz)	Up to 100	Up to 100
Power consumption (W)	4	2

In Figure 1.7, the ibNav kit is depicted as mounted on the body, showcasing the entirety of its hardware components. Conversely, Figure 1.8 provides an in-depth visual examination of each individual component.

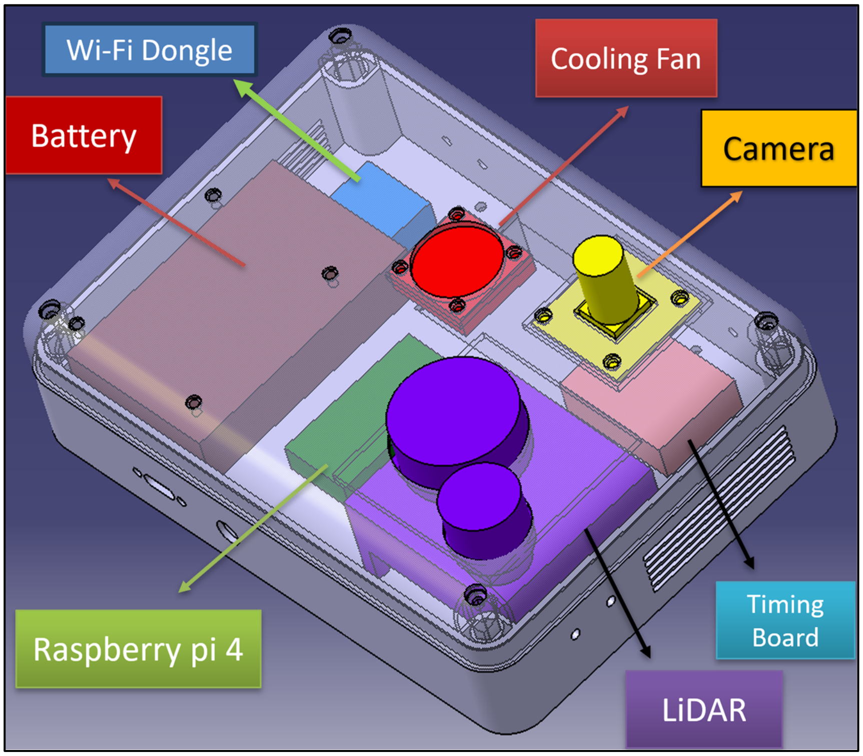


Figure 1.8 visual illustration of each ibNav kit component

Figure 1.9 LeddarTech M16 LSR

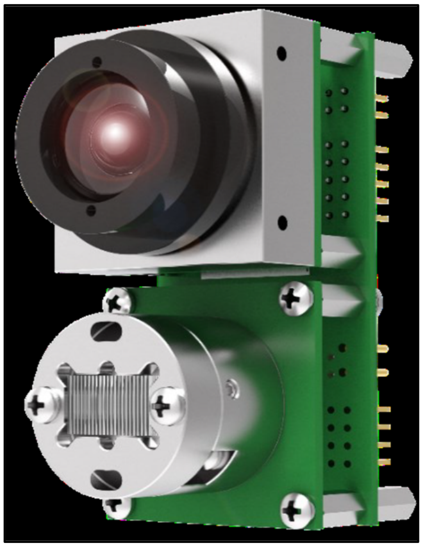


Figure 1.10 LeddarTech VU8

Extensive experiments utilizing the LiDAR for navigation and mapping applications have been undertaken. A comprehensive analysis of these experiments will be presented in the "Ground Control" section and Chapter 2. Figure 1.11 depicts a representative mapping derived from the unprocessed data measurements of the LiDAR in a 2D format.

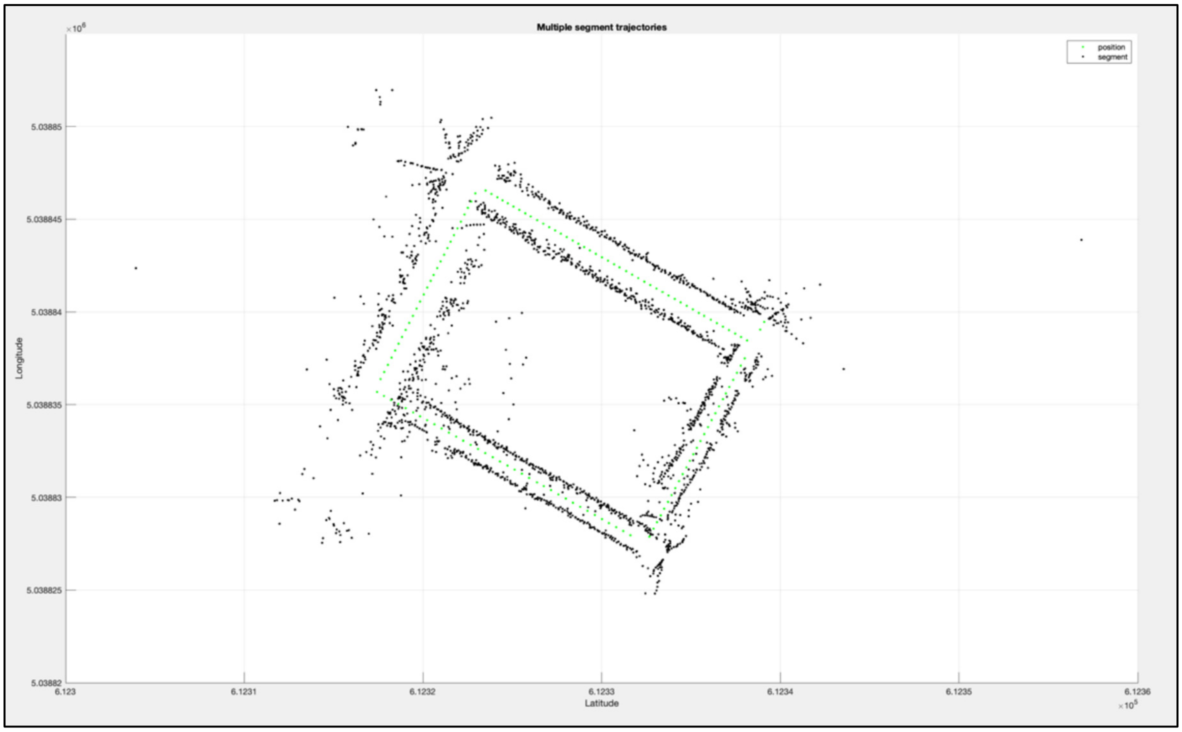


Figure 1.11 2D Mapping using raw LiDAR measurements

1.2.3 Optical Imaging and Camera

Optical imaging plays a pivotal role in diverse applications ranging from medical diagnosis to space exploration. For navigation and obstacle detection, particularly in intricate indoor environments, it is indispensable. The evolution of computer vision, in tandem with sophisticated algorithms, has enabled the accurate detection and localization of objects.

1.2.3.1 Types of Cameras Used

The Raspberry Pi camera, employed in this study, is a compact digital camera that offers capabilities optimal for the envisioned tasks. Its adaptability to diverse lighting conditions and its ability to render high-quality images make it an ideal choice for object detection and mapping applications. Figure 1.12 shows the specific Raspberry Pi camera used in this project.

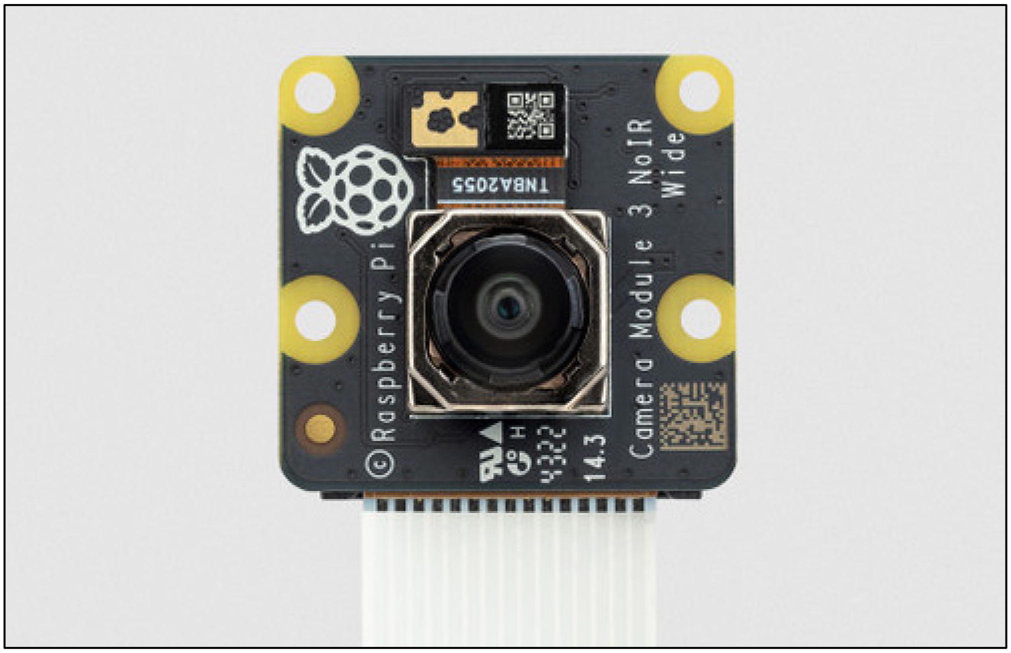


Figure 1.12 Raspberry Pi Camera

1.2.3.2 Image Processing and Analysis

Object detection is paramount in navigation, particularly in identifying and categorizing entities within an image frame. The overarching aim is to accurately demarcate the object's presence, often achieved through bounding boxes, followed by its correct classification. Contemporary approaches anchored in Deep-Learning (DL) have significantly transformed this domain. Notably, techniques such as the R-CNN family network, SSD, and YOLO have been at the forefront. Among these, the YOLO model, especially its third version - YOLO-V3, has been heralded for its real-time detection prowess. For this study, YOLO-V3 was tailored to detect indoor entities like doors, windows, and individuals. This customization necessitated the creation of a specific dataset and meticulous manual labeling. Additionally, the detection of structural elements, represented by horizontal and vertical lines, facilitates the discernment of walls and planning paths. The Probabilistic Hough Transform, an evolution of the traditional Hough transform, offers a robust solution for detecting these lines and ascertaining their orientations. Such information is quintessential in spatial understanding and subsequent navigation.

1.2.3.3 Role in Navigation and Obstacle Detection

For the practical application of these algorithms, translating the pixel-specific data into real-world coordinates is imperative. This necessitates camera calibration, achieved in this instance using a Raspberry-V4 camera. The resultant data, complemented by intrinsic and extrinsic camera parameters, can be translated to accurate real-world positions. The efficacy of the employed algorithms manifests in the ability to detect objects with a confidence level exceeding 70%. However, there's room for improvement, primarily by enriching the dataset with high-quality images. The culmination of these efforts is encapsulated in the ib-Nav project, which envisaged the creation of the Real-time Object Detection and Coordinate Estimation Framework. This Python-based cross-platform application boasts modules dedicated to object detection and 3D real-world coordinate estimation.

This framework can be adapted to detect a myriad of objects, both indoor and outdoor, including individuals, architectural elements, and even vehicles. A significant achievement of this platform is its capability to estimate 3D coordinates using a monocular system, which can discern the depth and real-world position of objects using just one optical camera. This modular framework is adaptable and can be augmented with additional modules as per the requirements, ensuring seamless integration and functionality. Figure 1.13 illustrates the estimating coordinates at long distance and object detection using raspberry pi camera mounted on the ibNav kit.

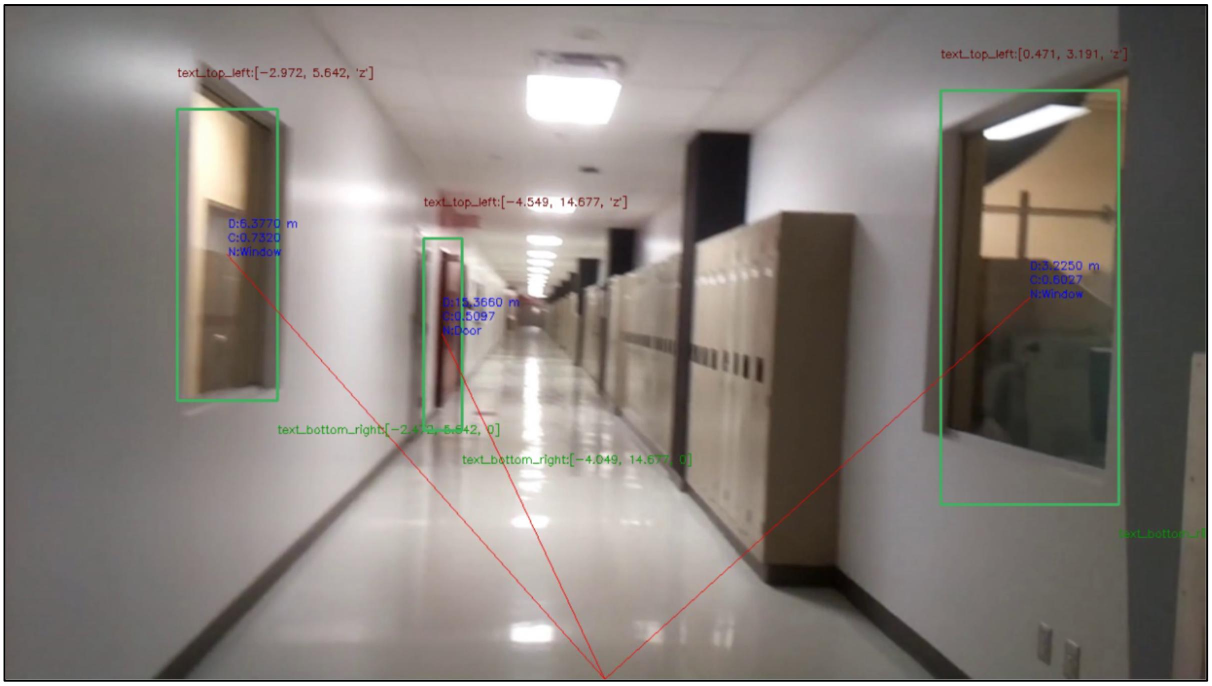

text_top_left:[-2.972, 5.642, 'z']
text_top_left:[-4.549, 14.677, 'z']
text_top_left:[0.471, 3.191, 'z']
D:6.3770 m
C:0.7320
N:Window
D:15.3660 m
C:0.5097
N:Door
D:3.2250 m
C:0.6027
N:Window
text_bottom_right:[-2.472, 5.642, 0]
text_bottom_right:[-4.049, 14.677, 0]
text_bottom_ri

Figure 1.13 Depth measurement and object detection using camera

1.2.4 Raspberry Pi

In the scope of this project, the Raspberry Pi serves a dual purpose: as a central processing unit and a conduit for data transmission. The Ibnav 6.1 sensor establishes a connection with the Raspberry Pi via a Wi-Fi MQTT protocol. To enhance the reliability of this connection, data from the sensor is routed to the Raspberry Pi 4, a pivotal gateway device, through a dedicated Wi-Fi (wlan0). Additionally, to facilitate efficient data flow, a secondary Wi-Fi dongle, interfacing via MQTT on wlan1, is integrated to forward the processed data to the server. This bifurcated Wi-Fi setup ensures segregation of incoming data streams from the sensor and outgoing streams to the server. On the software front, the Raspberry Pi runs applications developed in JavaScript and Python, responsible for the cleaning and intricate processing of data from the Ibnav 6.1 sensor. Moreover, wired connections anchor the Lidar and timing sensors to the Raspberry Pi. Upon local processing, the refined data is dispatched to a central server which then undertakes post-processing, visualization, and ultimately, dissemination to the Ground Control Station and associated mobile applications.

1.3 IbNav Software Framework

The RPNT-ibNav system has been meticulously developed to enhance soldiers' navigational capabilities in unfamiliar indoor and outdoor terrains while promoting efficient collaboration amongst team members. Through the integration of Unity3D, an advanced 3D engine recognized for driving over half of the contemporary mobile games, a graphical user interface (GUI) has been crafted to function seamlessly across multiple platforms. Given Unity3D's broad platform compatibility and its escalating significance in sectors such as automotive, engineering, and Artificial Intelligence (AI) due to its comprehensive toolset and cost-effectiveness, it emerges as an optimal selection for this project.

This interface is compatible with prevalent mobile platforms, namely iOS and Android, and has been expanded to encompass personal computers, specifically tailored for the Ground Control Station (GCS) of the ibNav System. The GCS interface is central to the system's functionality, playing a crucial role in displaying critical data emanating from User Equipment (UE) to the operator. Such data, procured from soldiers equipped with the RPNT-ibNav system and Unmanned Aerial Vehicles (UAVs), serves as a foundational element in facilitating real-time monitoring, analysis, and strategic mission-based decision-making.

At the core of this system's communication between ibNav sensors and the overarching RPNT framework is the MQTT protocol. It is highly valued for its attributes of low latency and reduced bandwidth requirements, especially in high-latency contexts. Its extensive application in the domain of the Internet of Things (IoT) and its accreditation as an open OASIS and ISO standard further accentuates its suitability. The Raspberry Pi, a revered open-source embedded system, functions as the cornerstone of this architecture, managing the connectivity and communication within the RPNT-ibNav infrastructure. For geospatial representation, Mapbox for Unity has been incorporated, a leading-edge mapping API that amalgamates data from diverse proprietary and open-source map sources, most prominently OpenStreetMap. This tool's refined 3D mapping capabilities solidify its position as an industry standard for mobile-based mapping within Unity.

Both mobile and PC interfaces adhere to principles of intuitive design, user-friendly interaction, and pragmatic utility. While the mobile GUI is characterized by a developer-oriented design, replete with a plethora of settings and analytical functionalities, the PC-adapted GCS interface underscores the swift accessibility to critical features. Such a design ethos ensures that operators can promptly extract necessary information, engage with the system's functionalities without undue distraction, and rely on the system to autonomously emphasize urgent communications or emergent situations.

1.3.1 Mobile Application Interface

The ibNav mobile application stands as a testament to cutting-edge navigational tools, originally developed in the Unity Engine 2019 and subsequently upgraded to the 2022 version. Central to its operation is its capability to directly interface with the primary server, allowing for a range of mission-centric functionalities.

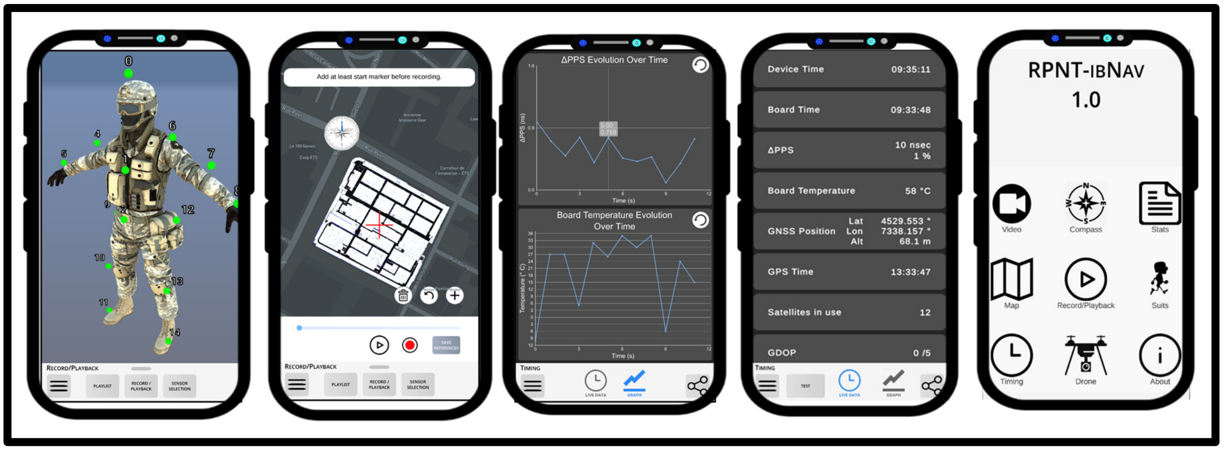
Upon establishing a connection with the server, the application furnishes users with the ability to initiate, terminate, and chronicle missions. Moreover, the application is replete with an array of features designed to enrich the user experience and bolster mission efficacy. These encompass:

- Video Recording: Empowers users to capture live visuals from the mission environment. Compass Functionality: Provides conventional orientation tools for streamlined navigation.
- Sensor Data Display: Relays real-time metrics related to the performance and status of integrated sensors.
- Environmental Mapping: Equips users with both 2D and 3D visual representations of the mission terrain.

- Mission Playback: Facilitates a retrospective examination of any mission, whether completed or ongoing.
- Mission Log Documentation: Avails the option to archive detailed mission data in a CSV format, aiding subsequent analyses.
- Suit Monitoring: Delivers a comprehensive overview of the connectivity and operational status of all associated suits.
- Drone Control Panel: Enables users to oversee timing metrics and exercise control over drone operations.
- Motion Capture: Integrates capabilities for recording precise human movements, transforming them into digital animations within the mission scenario. This is especially instrumental for tasks requiring detailed movement analysis or simulation in virtual environments.

Pertinently, the ibNav application's latest iteration, version 6, is laudably versatile in terms of platform compatibility. It operates seamlessly across Android and iOS operating systems. Additionally, this edition has undergone rigorous optimization to ensure efficient functioning on the most recent versions of both platforms.

For a detailed visualization of the application's user interface and its myriad functionalities spread across various tabs, Figure 1.14 provides an illustrative overview of different tabs if mobile application.



Add at least start marker before recording.

Device Time 09:35:11
Board Time 09:33:48
ΔPPS 10 nsec 1 %
Board Temperature 58 °C
GNSS Position Lat 4529.553 ° Lon 7338.157 ° Alt 68.1 m
GPS Time 13:33:47
Satellites in use 12
GDOP 0 /5

Figure 1.14 Different tabs of ibNav mobile application

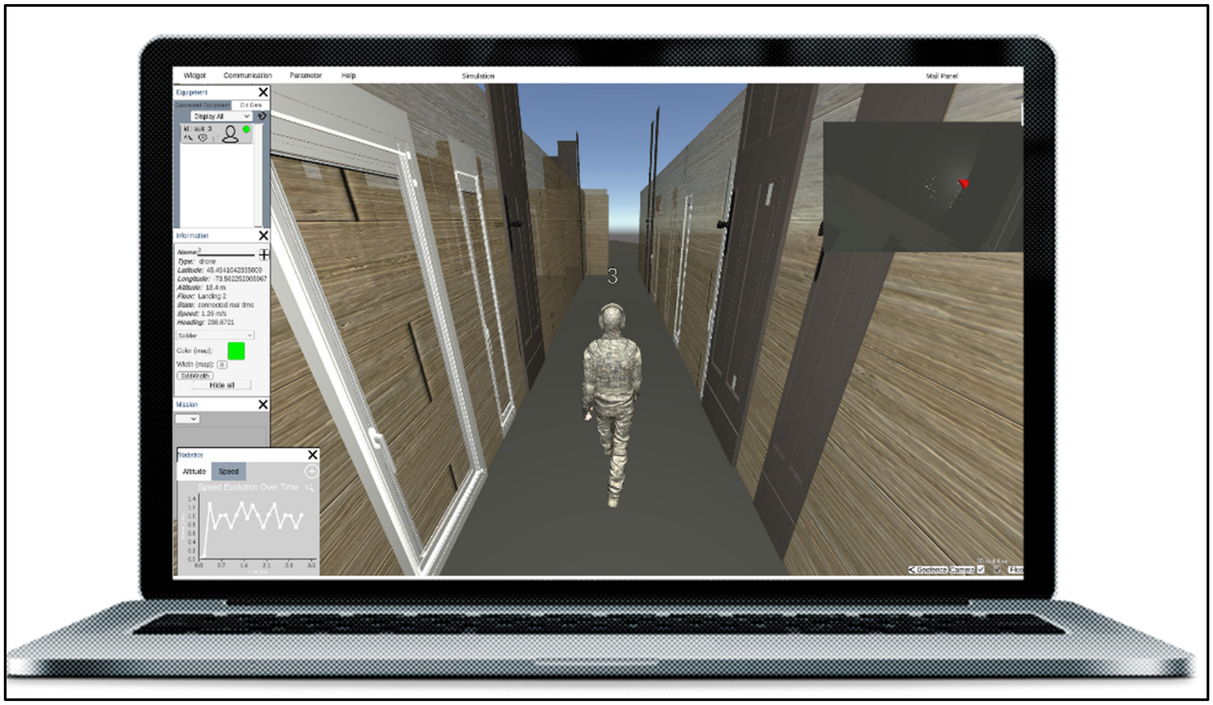
1.3.2 Ground Control System (GCS)

The Ground Control Station (GCS) serves as an indispensable interface for the relay and management of vital data to its users. The information it processes and displays is derived from a constellation of sensors incorporated within the ibNav system, which includes, but is not limited to, the IMU setup. The multifaceted nature of these sensor inputs ensures comprehensive coverage of navigational and operational metrics, bolstering the GCS's capacity to direct activities, notably the piloting of human-controlled UAVs.

To maximize user efficiency, the architectural design of the GCS application emphasizes both intuitiveness and practicality. There exists within the system a tiered arrangement of functionalities: primary or frequently utilized operations are presented prominently, ensuring immediate accessibility for operators. Conversely, functions of lesser regularity or secondary importance are subtler in their presentation, reducing potential visual and operational clutter, thereby maintaining optimal operator focus.

A noteworthy feature of the GCS is its automation capabilities. It is designed to autonomously activate specific panels upon receipt of emergency signals, ensuring operators are alerted instantaneously, without the need for manual oversight. The sophisticated intercommunication between the GCS client application and the dedicated server, mediated by the gRPC protocol, ensures a fluid exchange of data. Whether the need arises to retrieve specific information, or to read from a certain file, this protocol facilitates the structured data exchange. The adept integration of the gRPC protocol within the GCS ensures that every constituent element of the server's messages is accurately interpreted and acted upon by the application.

Operational compatibility is a core strength of the GCS, evidenced by its adaptability across various computing platforms, including Windows, Ubuntu, and Mac. This broad compatibility ensures that users across different environments can leverage the system's capabilities without technological constraints. A general overview of GCS application can be seen in Figure 1.15 where a 3D navigation and mapping under process.



Name: 3
Type: drone
Latitude: 45.4141042335009
Longitude: -73.562252008067
Altitude: 18.4 m
Floor: Landing 2
State: connected real time
Speed: 1.26 m/s
Heading: 298.6721
Soldier
Color (map): 🟩
Width (map): 8
Edit Width
Hide all

Figure 1.15 General overview of GCS

Moreover, the GCS, while primarily focused on navigation, has a spectrum of tabs and features dedicated to control and visualization. Among its varied functionalities, the 'widgets' tab stands out, encompassing equipment status, alerts, errors, and mission data. The 'parameters' tab offers connectivity options, allowing users to link to specified IP addresses. One of the standout features is the trajectory display, which visually represents a selected user's route within the environment. Especially crucial when navigating indoor terrains, this feature allows the GCS to showcase equipment trajectory based on a floor-by-floor selection, enabling granular movement tracking. The capacity for multi-floor event referencing in 2D ensures that trajectories are presented contextually, based on the equipment's movement across different floors.

Furthermore, the GCS demonstrates potential in the domain of environmental mapping. Through prospective integration of Simultaneous Localization and Mapping (SLAM) algorithms and the deployment of advanced sensors like LiDAR and camera systems, the platform holds promise for generating detailed environment maps. This feature, while currently under exploration, offers a tantalizing prospect for future research and innovation within the GCS framework. Accessing GCS require authentication using a username and password after which the main page of the GCS is shown in Figure 1.16.

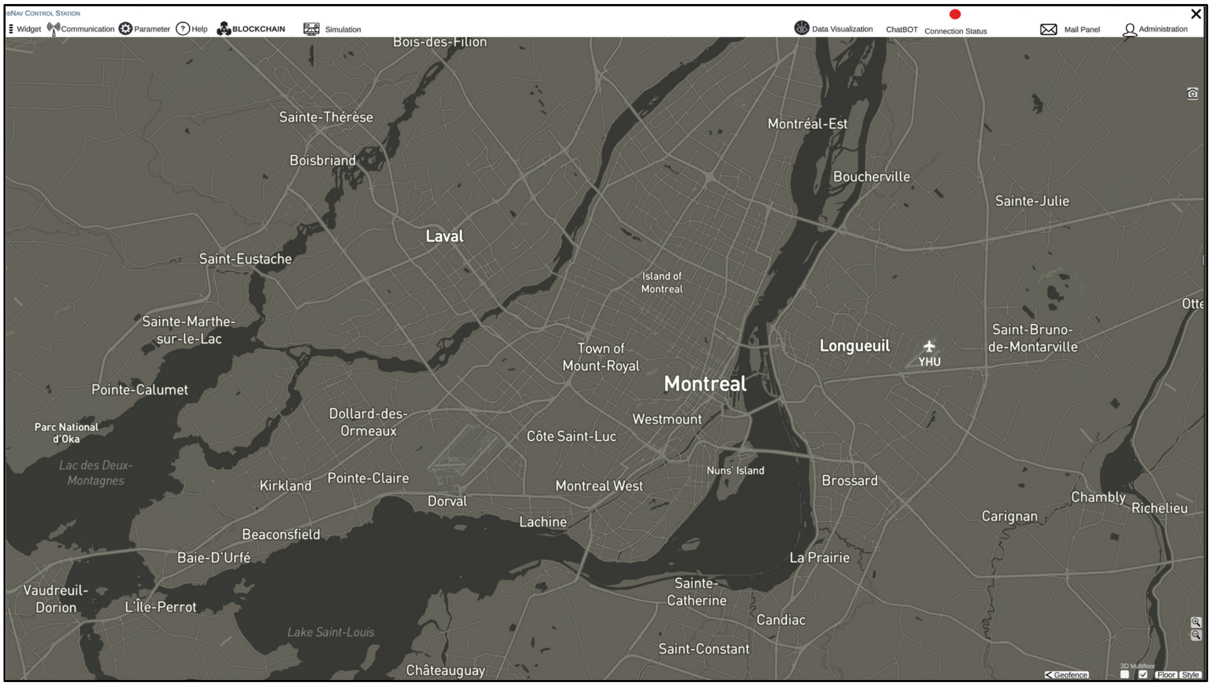


Figure 1.16 Main Page of GCS

1.3.2.1 Ground Control Station Functionalities

The Ground Control Station (GCS) has been rigorously developed to present a comprehensive suite of functionalities aimed at augmenting navigation, control, and monitoring tasks. The incorporation of these features within the GCS specifically caters to the multifaceted needs of the ibNav system:

- Data Display and Management: Proficiently processes and presents data sourced from an extensive ensemble of sensors integral to the ibNav system, guaranteeing real-time and accurate informational relay to the user.
- Unmanned Aerial Vehicle (UAV) Oversight: Empowered with tools to oversee and regulate the activities of human-controlled UAVs, establishing an unblemished environment for surveillance and control operations.
- Interface Hierarchization: The GCS design strategically positions functionalities, with primary operations given direct accessibility. Concurrently, secondary functionalities are integrated with discretion to minimize clutter and amplify the operator's concentration.
- Automated Emergency Response: The GCS possesses an intrinsic capability to autonomously unveil relevant panels during emergencies, guaranteeing instantaneous notifications to operators.
- Structured Data Communication: By leveraging the gRPC protocol, a streamlined data exchange between the application and server is actualized, ensuring seamless and accurate communication.
- 2D and 3D Navigation in Unity: The integration of Unity facilitates a sophisticated 2D and 3D navigation experience. This, paired with intricate mapping visualization, provides users with a multidimensional perspective of the navigational environment.
- Mapping Visualization using LiDAR: The GCS incorporates cutting-edge LiDAR technology to render precise mapping visualizations. This ensures an accurate representation of the environment, pivotal for navigational tasks.
- Trajectory Visualization: Enables graphical portrayal of user or equipment movement trajectories within a digital environment, enriching operational insights.

- Multi-level Event Referencing: The system incorporates a nuanced multi-floor event referencing in 2D, allowing operators to accurately trace equipment trajectories, vital for multi-storied infrastructures.
- Environmental Mapping Potential: While in the developmental purview, the GCS exhibits potential for comprehensive environmental mapping by synergizing SLAM algorithms and contemporary sensors, such as LiDAR and camera modules.
- Platform Versatility: Ensuring adaptability, the GCS is designed for compatibility across diverse computing platforms, including Windows, Ubuntu, and Mac.
- Real-time Navigational Analysis: With advanced communication protocols, the GCS can seamlessly initiate and conclude real-time navigation tests, supplemented with playback capabilities for in-depth analysis.
- Support for Collaborative Indoor Navigation: Through its functionalities and prospective mapping tools, the GCS emerges as a vital tool in the domain of collaborative indoor navigation.
- Equipment Movement Analytics: Tailored for intricate indoor landscapes, this feature facilitates granular insights into equipment movements, aiding strategic decision-making processes.

Incorporating this ensemble of functionalities, the GCS serves as an epitome of efficiency and precision, ready to address the evolving challenges in navigation and control systems.

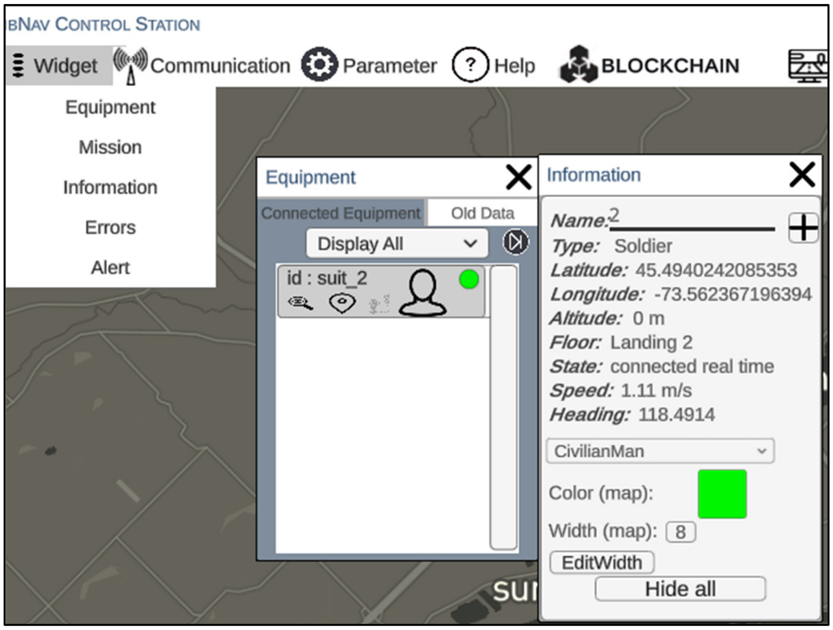
1.3.2.2 Ground Control Station Tabs

GCS uses different tabs to facilitate the performance and functionality of the platform for different purposes. Here is a list of the most prominent Tabs in GCS:

- **Widgets Tab**: The Widgets tab, shown in Figure 1.17, within the ibNav Ground Control Station (GCS) stands as an essential interface, meticulously structured to offer real-time insights about the mission, affiliated equipment, and potential discrepancies. Each embedded widget caters to a distinct facet of mission-related information, ensuring operators possess a holistic view of the ongoing operations. The "Equipment Widget" exemplifies this strategic design. Embedded within the Widgets tab, it furnishes users with intricate data about the various instruments integrated into the system. From showcasing the operational status of critical components like the Raspberry Pi, IMU, and drones to rendering a visual depiction of sensor outputs, this widget becomes indispensable.

By presenting this data in an intelligible manner, it empowers users to swiftly diagnose equipment-related anomalies and thereby, institute requisite corrective measures. Complementing the equipment monitoring capabilities is the "Mission Widget". This feature elucidates the nuanced aspects of the ongoing operation, encapsulating details ranging from initiation and culmination timestamps to geographical waypoints and task-specific objectives.

By consolidating this information into a singular interface, the Mission Widget promotes situational awareness, fostering an environment conducive to mission accomplishment. However, the functionalities are not solely limited to proactive mission tracking. The "Errors and Alerts Widget" serves as the GCS's troubleshooting beacon. This widget meticulously enumerates any operational deviations or potential pitfalls that could compromise mission integrity. Moreover, it doesn't merely highlight discrepancies but also proactively provides rectification suggestions. Such a feature ensures that operators are not only apprised of emergent issues but are also equipped with actionable insights to redress them.



Display All

id : suit_2

Name: 2
Type: Soldier
Latitude: 45.4940242085353
Longitude: -73.562367196394
Altitude: 0 m
Floor: Landing 2
State: connected real time
Speed: 1.11 m/s
Heading: 118.4914

CivilianMan

Color (map): 🟩

Width (map): 8

Figure 1.17 Widgets tab of GCS

- **Communication Tab**: Within the robust framework of the ibNav Ground Control Station platform, the Communication tab stands as an instrumental component dedicated to facilitating meticulous data transmission and validation. This tab, which is shown in Figure 1.18, is meticulously crafted to cater to diverse functionalities, encompassing both real-time testing and data playback. One of the primary utilities of the Communication tab is its provision for initiating real-time tests. This feature empowers users to validate the data transmission sequence, starting from its origin at the IMU, through its intermediate stop at the Raspberry Pi, and culminating at its destination at the RiCF server. Such validation is paramount to ensure data integrity and the smooth functioning of the entire navigation system.

  Beyond real-time validation, the tab also offers capabilities for playback from either previous data sets or locally stored files. This function is of significant utility, permitting users to retrospectively analyze how the system processed past data and assess its competency in handling particular data-intensive scenarios. Moreover, the tab's design is imbued with flexibility, granting users the ability to modify the server

IP address as required. Such adaptability is not merely a luxury but a necessity, considering the dynamic nature of data transmission networks, and the potential need to connect to a myriad of servers. Furthermore, the Communication tab affords a panoramic view of the system by enabling the monitoring of references, sensors, and drones. This holistic perspective ensures that users always remain apprised of the system’s multifaceted operations.

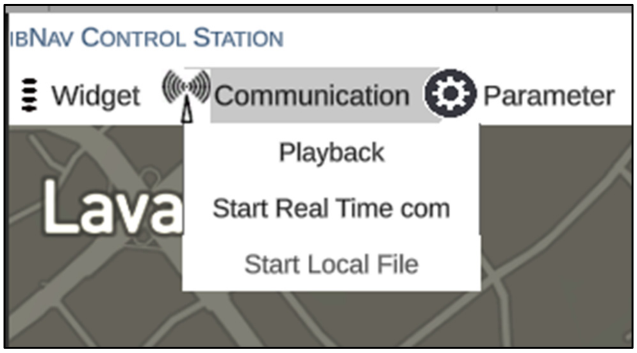


Figure 1.18 Communication tab of GCS

- **Parameters Tab**: Embedded within the architecture of the ibNav Ground Control Station platform is the Parameters tab, a pivotal interface that grants users the autonomy to customize and refine various operational parameters of the system. This facility accentuates the user-centric design of the platform, allowing for dynamic adaptability to diverse mission profiles and user specifications.

  Central to the functionalities of the Parameters tab is the ability to modify the server IP address. This feature, while seemingly straightforward, plays an instrumental role in ensuring seamless data communication. It caters to scenarios where users may need to transition to an alternate data-hosting server or instances where a server undergoes relocation. Beyond server settings, this tab expands its purview to encompass modifications related to references, sensors, and drones. The real-time monitoring capability embedded within the tab ensures that any modifications are not made in isolation but are continually assessed for their system-wide implications.

A noteworthy enhancement of the Parameters tab is its provision for simulative assessments. By offering a platform where drone and kit scenarios can be replicated, users are endowed with the foresight to predict system performance under specified conditions. Whether it's sketching a reference trajectory for a drone, delineating its velocity vector, or modulating settings for wall creation and object detection simulations, users are equipped to perform a spectrum of tests, thus enabling pre-emptive optimization. Figure 1.19 illustrates the parameters tab.

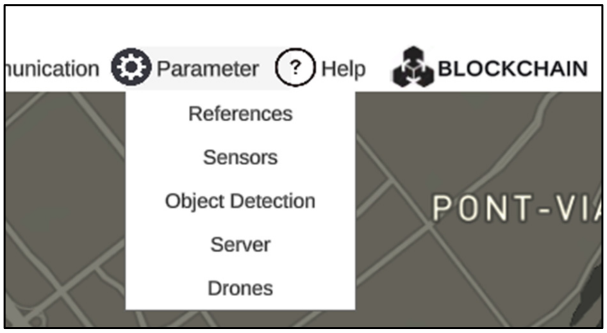


Figure 1.19 Parameters tab of GCS

- **Simulation Tab**: Within the framework of the ibNav Ground Control Station platform lies the Simulation tab, a well-designed interface tailored to facilitate robust simulation exercises. Its primary objective is to provide users with a controlled environment where different scenarios can be simulated, therein assessing the resilience and adaptability of the system. A salient feature of the Simulation tab is its capacity to mimic the behavior of drones and kits. Users have the latitude to initiate simulations for either, tweaking variables like velocity and trajectory direction to mirror real-world challenges. This dynamic simulation ability underpins the tab's significance, enabling users to evaluate the system's response to diverse operational landscapes, from intricate indoor settings to the unpredictabilities of outdoor terrains.

Beyond mere drone and kit simulations, the tab offers a facility to create reference trajectories. This feature, pivotal in its own right, allows users to extrapolate the system's performance metrics under predetermined conditions, thereby serving as a predictive tool. Another intriguing facet of the Simulation tab is its provision for object detection exercises. By granting users the capability to erect virtual barriers, it gauges the system's proficiency in recognizing such obstructions and promptly relaying alerts. This becomes particularly essential when circumnavigating environments laden with obstacles, ensuring user safety and system efficiency. Furthermore, the Simulation tab, which is shown in Figure 1.20, transcends its immediate simulation duties to proffer performance monitoring. Users can derive analytical insights from the simulation outcomes, iterating over the system parameters to hone its precision.

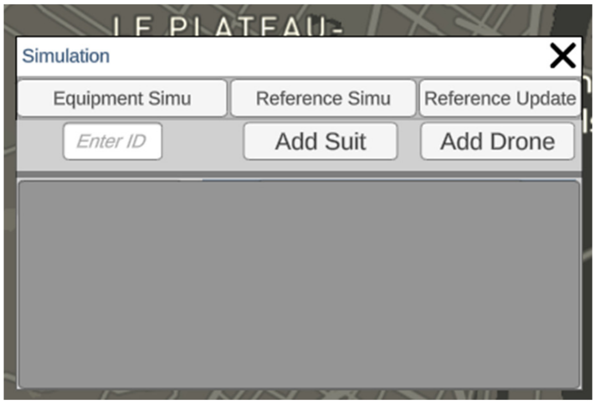


Figure 1.20 Simulation tab of GCS

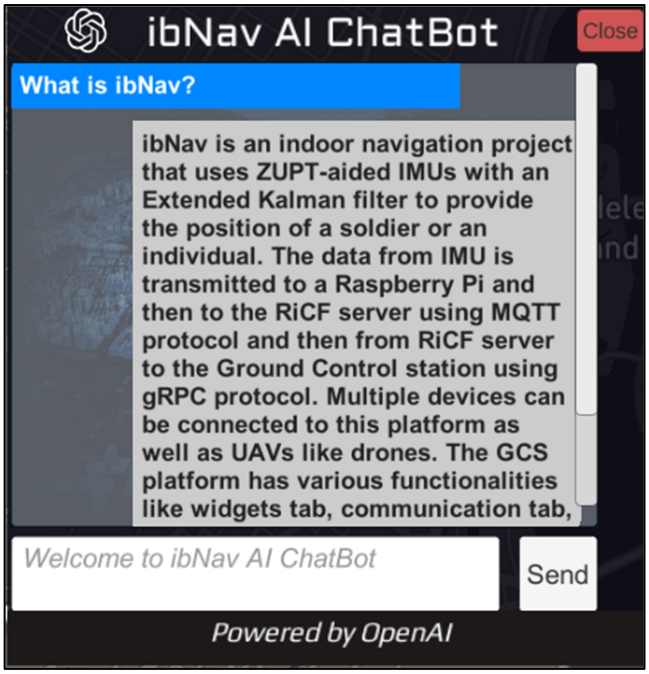
- **ChatBot Tab**: Within the constellation of functionalities offered by the ibNav Ground Control Station platform, the AI Chatbot Panel stands as a testament to the fusion of state-of-the-art AI and user-centric design. Designed meticulously to act as a bridge between users and the sophisticated mechanics of the platform, this panel ensures that individuals, irrespective of their technical prowess, can navigate and harness the system's capabilities with ease.

  The inherent purpose of the AI Chatbot Panel transcends the conventional paradigms of chat interfaces. Rather than merely providing pre-defined answers to user queries, it assists users in actual tasks. Whether initiating tests, recording data, or configuring IP addresses, the chatbot's utility in guiding users is unparalleled, obviating the need for extensive manuals or training sessions.

What sets this chatbot apart, however, is its integration with the OpenAI API. Recognized as a vanguard in the domain of natural language processing, OpenAI's technologies imbue the chatbot with an ability to understand and respond to user queries with human-like precision. This congruence of advanced AI with user interface design heralds a paradigm shift in how individuals interact with technology, ensuring a seamless, intuitive, and, importantly, efficient user experience.

Moreover, the specific tailoring of this chatbot for the Unity environment serves as an added feather in its cap. Unity developers, a cohort known for their pursuit of simplicity without compromising on functionality, find in this chatbot an ally. The incorporation of such a cutting-edge plugin not only streamlines their workflow but also elevates the overall ambiance of the development environment, rendering it both congenial and productive.

In summation, the AI Chatbot Panel of the ibNav Ground Control Station which is shown in Figure 1.21 is not merely a feature; it is a paradigmatic leap in human-machine interaction within the ambit of the Unity ecosystem. By simplifying complex operations and ensuring an intuitive interface, it undeniably positions itself as an indispensable tool for users and developers alike.



What is ibNav?

ibNav is an indoor navigation project that uses ZUPT-aided IMUs with an Extended Kalman filter to provide the position of a soldier or an individual. The data from IMU is transmitted to a Raspberry Pi and then to the RiCF server using MQTT protocol and then from RiCF server to the Ground Control station using gRPC protocol. Multiple devices can be connected to this platform as well as UAVs like drones. The GCS platform has various functionalities like widgets tab, communication tab,

Figure 1.21 AI ChatBot tab of GCS

- **Connection Status**: The Connection Status Tab within the ibnav Ground Control Station platform serves as a crucial diagnostic tool, providing users with real-time visual indications regarding the connectivity status with the RiCF server. As shown in Figure 1.22, employing a dichotomous color-coded scheme—green signifying a robust and stable real-time connection, while red denotes any disruptions or connectivity lapses—it aims to offer users immediate and unambiguous feedback. Beyond this direct visual cue, the tab also provides detailed error messages in the event of any disruptions. These messages are instrumental in pinpointing the underlying issues, thus facilitating more expedited and precise troubleshooting. This feature is pivotal, ensuring that the Ground Control Station maintains its operational efficiency, minimizes potential data losses, and supports prompt corrective actions.

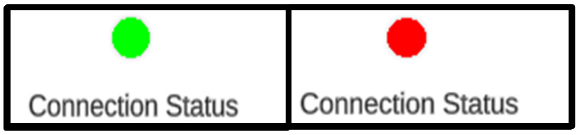


Figure 1.22 Connection status tab of GCS

1.4 RiCF Server Integration

1.4.1 Role and Function of the RiCF Server

The RPNT-ibNav Control Framework, commonly referred to as RiCF, stands as a monumental stride in the realms of cross-platform application development. Crafted meticulously using Python, this server acts as the linchpin in the ibNav communication architecture. It epitomizes the aspiration to coalesce the multifaceted communications scattered across the RPNT-ibNav system.

This alignment and assimilation become imperative, especially when there exists a pressing need to process and rechannel data across applications, which, under ordinary circumstances, would not synchronize due to disparate communication frameworks. The genius of RiCF lies in its ability to interconnect various Internet of Things (IoT) components associated with the ibNav project, ranging from the likes of ibNav sensors and Lidar to more advanced entities like Raspberry Pi and Drones, all of which predominantly communicate via the MQTT protocol. This connection extends to the Ground Control Station GUI, which communicates via the avant-garde open-source RPC framework, gRPC. It's imperative to note that such a bifurcation is not a mere fragment of design choice but a conscious strategic decision.

By ensuring that multiple graphical user interfaces (GUIs) can concurrently interact with the central server, the designers have overcome the typical limitations faced when embedding both within a singular application. Furthermore, the inherent flexibility of the RiCF server grants it the capability to operate within local network confines or transcend to operate in expansive cloud-based architectures, thus augmenting GUI's operability across diverse network structures.

1.4.2 Core Objectives

The very inception of the RiCF server was guided by a series of calculated objectives:

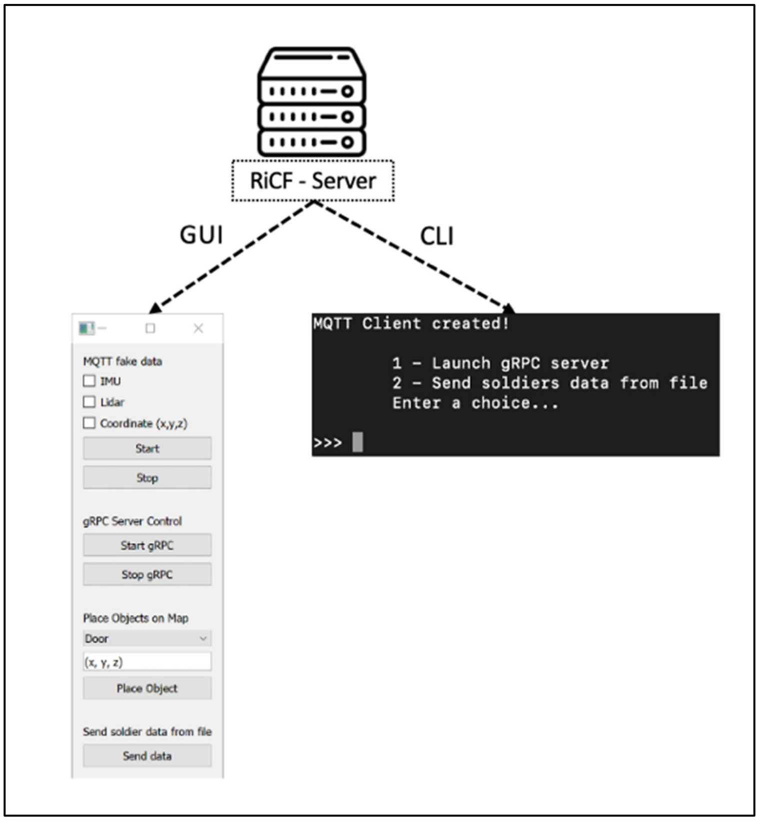
- Centralized Communication: One of the primary aspirations was to orchestrate a seamless channel of communication that interlinked the myriad entities of the ibNav project. This includes the varied players in the domain, such as drones, soldiers, and the integral ground control station GUI, colloquially referred to as RiCF-Unity.
- Data Processing: The server is adept at transmuting incoming MQTT data streams into gRPC data, tailor-made for the GUI. This essentially involves extracting pertinent information from MQTT frames sourced from the system's drones and soldiers.
- Data Logging: With an eye on post-operational analysis, the server logs all incoming data into CSV (Comma-separated values) files, making them primed for post-processing and subsequent playback.
- Real-Time Response: The server is tailored to respond in real-time to GUI requests. This can range from actions such as streaming to the playback of previously logged CSV files.
- Test Bed for Ground Control Station GUI: Besides its primary functionalities, the server doubles up as a testing platform for the ground control station GUI. It can concoct and dispatch simulated data, a feature invaluable for rigorous testing phases.
- Real-time Algorithmic Operations: A future-centric objective is the introduction of real-time algorithms that would process data inflow from various system components.
- Centralized MQTT Link: Owing to the fact that all Raspberry Pis in the project are architecturally designed to be attuned to the RiCF-Server’s MQTT broker, the server stands as the nerve center linking all equipment operating on MQTT.

1.4.3 Interfaces

The RiCF server has been envisioned to be adaptable, catering to both Command Line Interface (CLI) and a more immersive GUI. The CLI stands crucial for remote server applications. Should the application be operational on a remote server, users can effortlessly connect to the platform via SSH, thereby inputting commands directly through the terminal.

On the contrary, for a more localized experience, users can leverage the GUI, treating it as conventional software. However, it's noteworthy that even in such environments, the CLI interface remains accessible, especially in OS setups devoid of graphic capabilities, such as a GUI-less Linux installation. Expanding on visualization, the server integrates the prowess of Unity, a renowned cross-platform 3D engine crafted using C#. This is the very engine that powered the ibNav GUI development for portable devices like smartphones and tablets.

The communication between the primary python server and Unity is facilitated using the gRPC protocol, ensuring that even if the RiCF server operates remotely, the advanced visualization capabilities of Unity can still be rendered directly on the user's localized system. A detailed illustration of RiCF interfaces is shown in Figure 1.23.



```
MQTT Client created!

        1 - Launch gRPC server
        2 - Send soldiers data from file
        Enter a choice...

>>> 
```

Figure 1.23 RiCF Interfaces

1.4.4 Communication

RiCF's communication architecture is emblematic of simplicity and efficiency. It is envisioned to be the intermediary, harmonizing the diverse facets of the ibNav system. At its core, the system is bifurcated into components operating on MQTT and those functioning on gRPC. The server stands as the watchful sentinel, monitoring these communications. It receives and dispatches data via both MQTT and gRPC, establishing itself as the central nexus of the ibNav system's architecture. Furthermore, the RiCF server showcases adaptability; it can function on the same computer housing the Unity GUI application, migrate to a different computer on the identical network, or even operate from a cloud-hosted remote server in an entirely different network framework.

Certain nomenclatures within the system may seem deceptively similar, warranting differentiation:

- RiCF Server vs. RiCF GUI: While both entities are prefixed with "RiCF", their functionalities are divergent. The RiCF Server epitomizes the central communication hub of the entire RPNT-ibNav system, bridging MQTT and gRPC communications. In contrast, the RiCF GUI is the interface that interacts with the server to yield visual outputs such as 2D/3D maps.
- RiCF Server vs. General Servers: The term "server" is ubiquitous in IT lexicon. However, when referring to "RiCF Server", it is a specific component of the ibNav system. In contrast, the generic term "server" could be any hardware or software entity with server functionalities.
- RiCF Server vs. gRPC Server: The gRPC server operates within the RiCF Server's ecosystem, facilitating communication between the RiCF Server and other clients using the gRPC protocol.

The incorporation of gRPC showcases a strategic pivot from MQTT. While MQTT, with its lightweight architecture optimized for high-latency or unpredictable networks, is quintessential for IoT devices, it has limitations when viewed from a server-centric perspective. For instance, MQTT, being a messaging protocol, doesn't inherently support data communication. This implies that there's no standardized format for its payload, and any semblance of interoperability between applications hinges on a priori agreement on a specific format.

In stark contrast, gRPC, with its orientation towards application interoperability, emerges as the preferred choice. Its universal availability across myriad languages and its utilization of Google's Protocol Buffer to define structured inter-application data exchange sets it apart. The gRPC framework almost renders the server-client communication invisible, allowing for direct server method calls as though they were local objects.

With gRPC's service definition, methods accessible by clients are predefined, and their parameters and return types are articulated using Protocol Buffer. Moreover, gRPC extends multiple communication modes, ranging from unary and bidirectional data streaming to server-side and client-side streaming. Additionally, its foundation on the HTTP/2 protocol, which boasts of bidirectional data flow, persistent connections, and advanced authentication measures, enhances its efficacy.

CHAPTER 2

A SECURE ZUPT-AIDED INDOOR NAVIGATION SYSTEM USING BLOCKCHAIN IN GNSS-DENIED ENVIRONMENTS

Ali Shakerian [a], Ali Eghmazi [a], Justin Goasdoué [b] and René Jr Landry [a]

[a] Department of Electrical Engineering, École de Technologie Supérieure,
Montréal, QC H3C 1K3, Canada
[b] Department of Computer Sciences, Université de Technologie de Compiègne,
60200 Compiegne, France

Article published in *the journal MDPI-Sensors*[1], July 2023

2.1 Abstract

This paper proposes a novel Blockchain-based indoor navigation system that combines a foot-mounted dual-inertial measurement unit (IMU) setup and a zero-velocity update (ZUPT) algorithm for secure and accurate indoor navigation in GNSS-denied environments. The system estimates the user’s position and orientation by fusing the data from two IMUs using an extended Kalman filter (EKF). The ZUPT algorithm is employed to detect and correct the error introduced by sensor drift during zero-velocity intervals, thus enhancing the accuracy of the position estimate. The proposed Low SWaP-C blockchain-based decentralized architecture ensures the security and trustworthiness of the system by providing an immutable and distributed ledger to store and verify the sensor data and navigation solutions. The proposed system is suitable for various indoor navigation applications, including autonomous vehicles, robots, and human tracking. The experimental results provide clear and compelling evidence of the effectiveness of the proposed system in ensuring the integrity, privacy, and security of navigation data through the utilization of blockchain technology. The system exhibits an impressive ability to process more than 680 transactions per second within the Hyperledger-Fabric framework. Furthermore, it demonstrates exceptional accuracy and robustness, with a mean RMSE error of 1.2 m and a peak RMSE of 3.2 during a 20 min test. By eliminating the

[1] https://doi.org/10.3390/s23146393

reliance on external signals or infrastructure, the system offers an innovative, practical, and secure solution for indoor navigation in environments where GNSS signals are unavailable.

Keywords: indoor navigation, inertial measurement unit (IMU), zero-velocity update (ZUPT), positioning, blockchain, secure, GNSS-denied, extended Kalman filter (EKF)

2.2 Introduction

A secure and reliable indoor navigation system is essential for a wide range of applications, including emergency response, military operations, and location-based services. However, traditional indoor navigation systems are often vulnerable to privacy breaches and unauthorized access to sensitive location data. To address these concerns, researchers have explored the use of blockchain technology as a tamper-resistant and decentralized platform for data storage and verification. In recent years, there has been growing interest in integrating blockchain technology with zero-velocity updates (ZUPTs)-aided pedestrian inertial navigation systems (INS) for indoor navigation. This integration has the potential to provide a highly secure and trustworthy indoor navigation system that protects users' privacy and prevents unauthorized access to sensitive location data.

This paper proposes a novel secure ZUPT-aided indoor navigation system using blockchain technology, which leverages the strengths of ZUPT-aided pedestrian INS techniques and blockchain technology to provide a highly secure and reliable indoor navigation system.

Zero-velocity updates (ZUPTs) have been widely used in pedestrian inertial navigation systems (INS) for indoor positioning and navigation. A low-cost foot-placed ultra-wideband (UWB) and IMU fusion-based tracking system has been proposed for IoT applications (Naheem and Kim 2022). An enhanced foot-mounted PDR method with adaptive ZUPT and multisensor fusion has been developed for seamless pedestrian navigation (Tao et al. 2022). Error analysis has been conducted on ZUPT-aided pedestrian INS to identify sources of errors and improve system performance (Wang, Chernyshoff, and Shkel 2018). Improvements in, and evaluations of, zero-velocity detectors have been achieved for foot-mounted INS, including the use of double-threshold zero-velocity updates (ZUPTs) (Zemam and Benzerrouk 2010) and unscented Kalman filter for accuracy improvement. Several studies have investigated the performance of ZUPT-aided pedestrian INS and other related systems, including the characterization of foot-mounted ZUPT-aided INSs (Nilsson, Skog, and Handel 2010), pseudo-ZUPT re-detection with double-threshold ZUPT for better performance (Zhao and Ahamed 2021), re-evaluation of algorithmic basics for ZUPT-based pedestrian navigation (Wagner, Kohl, and Gyorfi 2022), scenario-dependent ZUPT-aided pedestrian INS with sensor fusion (Wang, Jao, and Shkel 2021), and smoothing techniques for ZUPT-aided INSs (Simon Colomar, Nilsson, and Handel 2012).

Other studies have focused on specific aspects of ZUPT-aided pedestrian INS, such as stance-phase detection (Wang et al. 2015), step detection using foot-mounted permanent magnets (Norrdine, Kasmi, and Blankenbach 2016), estimation errors due to IMU noises (Wang, Chernyshoff, and Shkel 2020), and the impact of IMU mounting position on accuracy (Wang, Askari, and Shkel 2019).

The rapid development of wireless communication technology and mobile devices has led to the widespread use of indoor localization and navigation using WiFi sensing. However, this advancement also raises concerns about the privacy and security of users' personal information and location data (Yang et al. 2022). To address this issue, a novel oblivious data sharing scheme utilizing an oblivious transfer protocol has been proposed, which effectively hides location coordinates and protects the privacy of users and servers (Yang et al. 2022). Additionally, a string length-based transformation algorithm based on private information retrieval has been integrated into the protocol to reduce communication overhead and ensure the privacy of required keywords (Yang et al. 2022).

Hyperledger Fabric, a permissioned blockchain platform, offers privacy features and endorsement policies, but there are still privacy risks and attacks associated with disclosing the endorser signature and endorsement policy (Dharani, Sundarakantham, and Singh 2022). To enhance the security of endorsements in Hyperledger Fabric, a framework with a secure endorsement system has been proposed, incorporating an efficient linkable threshold ring signature scheme and a ZK-SNARK set membership proof (Dharani et al. 2022). The endorsement policy is obfuscated using universal RSA accumulators, and individual membership proofs are batched for efficient validation (Dharani et al. 2022). While there are several survey papers on indoor positioning methods and the security/privacy of location-based services (LBS), there is a lack of comprehensive research on the security and privacy considerations specifically focused on indoor positioning methods (Sartayeva and Chan 2023). To bridge this gap, a systematic survey has been conducted, categorizing indoor positioning methods into non-collaborative, collaborative, and other methods, and discussing the security and privacy issues associated with each method (Sartayeva and Chan 2023).

Blockchain technology, particularly Hyperledger Fabric, has gained popularity in various fields, including supply chain and digital identity (Peregrina-Pérez, Lagares-Galán, and Boubeta-Puig 2023). Hyperledger Fabric is an open-source platform known for its modularity and versatility, allowing for the establishment of private networks to protect sensitive data. It supports smart contracts in different programming languages, including Solidity, through the Ethereum virtual machine. Its permissioned architecture makes it suitable for domains such as businesses, healthcare, and manufacturing (Peregrina-Pérez et al. 2023). The continuous evolution of blockchain technology has led to the emergence of new platforms, such as Ethereum and Hyperledger Fabric, which are widely used in industry at present (Zhao 2022). Different blockchain implementations utilize various consensus protocols, and a brief comparison between Hyperledger Fabric and Ethereum has been provided, considering aspects such as consensus and smart contracts (Zhao 2022).

To ensure the safety of positioning, navigation, and timing (PNT) in satellite navigation systems, a secure PNT system with hybrid physical principles has been proposed, focusing on the augmentations of the BeiDou satellite system (BDS) (Yang et al. 2023). This comprehensive PNT infrastructure encompasses deep space and deep ocean, combining BDS and various other technologies. Resilient PNT applications, intelligent PNT services, and key technologies are discussed as future research directions (Yang et al. 2023). With the widespread use of Internet of Things (IoT) technology, managing and protecting sensitive private data in data-sharing has become a critical issue (C. L. Chen et al. 2022).

A Hyperledger Fabric blockchain-based secure data-transfer scheme for enterprises in the Industrial Internet of Things (IIOT) has been proposed, utilizing the InterPlanetary File System (IPFS) network and the Elliptic Curve Digital Signature Algorithm (ECDSA) to ensure privacy protection and efficient data transmission (C. L. Chen et al. 2022). The scheme outperforms traditional networks and existing blockchain systems in terms of throughput, latency, and system overhead (C. L. Chen et al. 2022). The indoor navigation of unmanned aerial vehicles (UAVs) in GPS-denied areas requires accurate positioning techniques (Lin and Zhan 2023). To improve the accuracy of visual inertial odometry (VIO) in UAV indoor navigation, an integration of ultra-wideband (UWB) positioning and VIO has been proposed, which effectively addresses the limitations of each individual system (Lin and Zhan 2023). The proposed technique achieves centimeter-level accuracy and real-time performance for UAV indoor navigation applications (Lin and Zhan 2023).

Hyperledger Fabric Private Blockchain Network (HFPBN) architecture has gained attention due to its potential in various domains (Gaba et al. 2022). In the case of Vehicular Ad-hoc Networks (VANETs), an architecture for an HFPBN has been explored, considering components such as the ordering service, membership service, and endorsement policy (Gaba et al. 2022). The architecture provides secure and efficient data sharing and consensus mechanisms for VANETs, ensuring reliable and tamper-proof operations (Gaba et al. 2022).

In addition to the architecture, the block size is a crucial parameter affecting the performance of a Hyperledger Fabric Private Blockchain Network (HFPBN). The impact of block size on performance parameters such as transaction throughput, confirmation time, and resource utilization has been investigated. The findings provide insights into ways to optimizing the block size to achieve a better performance in HFPBNs (Gaba et al. 2022). Wearable-based collaborative indoor positioning systems (CIPSs) require secure and reliable authentication protocols (Casanova-Marqués et al. 2023). A decentralized attribute-based authentication (ABA) protocol has been proposed, which allows for users to access location-based services (LBS) securely and anonymously, leveraging wearable devices and blockchain technology. The protocol ensures the privacy, integrity, and non-repudiation of user attributes, enabling trusted and collaborative positioning in CIPSs (Casanova-Marqués et al. 2023).

Blockchain technology has demonstrated potential in improving supply chain management and traceability (Ravi et al. 2022). Focusing on the coffee supply chain industry, an analysis has been conducted on the application of blockchain, specifically Hyperledger Fabric, in enhancing transparency, accountability, and sustainability in the coffee supply chain. The proposed implementation utilizing smart contracts and immutable ledgers provide a tamper-proof and trustworthy system for tracking coffee products (Ravi et al. 2022).

Multi-robot collaboration in industrial applications, such as inventory management, requires efficient coordination and localization mechanisms (Salimi et al. 2022). To address this challenge, a framework has been proposed, integrating Hyperledger Fabric blockchain, UWB localization, and a multi-robot system for inventory management (Salimi et al. 2022). The framework ensures reliable data sharing, secure transactions, and accurate localization, enhancing the efficiency of inventory management in industrial settings (Salimi et al. 2022).

In recent years, there has been a growing need for secure and reliable indoor navigation systems that can protect users' privacy and prevent unauthorized access to location data. Several studies have explored the use of blockchain technology to provide a tamperresistant and decentralized platform for data storage and verification (Jao and Shkel 2021; Sangenis, Jao, and Shkel 2022; Yang et al. 2021). However, the integration of blockchain technology with ZUPT-aided pedestrian INS techniques has not been extensively studied. A recent study proposed a symmetrical-net method that uses adaptive zero-velocity detection for ZUPT-aided pedestrian navigation, which improves the accuracy of the system (Zhao and Ahamed 2021). Therefore, there is a need for further research on the development of secure and reliable indoor navigation systems that can meet the growing demand for privacy-preserving and trustworthy location-based services, while leveraging the strengths of blockchain technology and advanced ZUPT-aided pedestrian INS techniques.

To address this need, this study proposes a secure ZUPT-aided indoor navigation system using blockchain technology. The system aims to provide reliable and accurate indoor navigation while ensuring the security and privacy of users' location data. The proposed system leverages the strengths of ZUPT-aided pedestrian INS techniques and blockchain technology to develop a tamper-resistant and decentralized platform for data storage and verification. This system is expected to be particularly useful in scenarios where privacy and security are critical, such as healthcare facilities, financial institutions, and military facilities. The proposed system can also be extended to other applications that require accurate and secure indoor navigation, such as robotics, autonomous vehicles, and smart homes.

2.3 Materials and Methods

This section describes the methodology and algorithms used in this project in two sections. In the first sections, the focus will be on the mechanization of a foot-mounted IMU, which involves the integration of ZUPT, inertial measurement unit (IMU) measurements, and an extended Kalman filter (EKF) algorithm. The ZUPT equations and their mathematical background will be discussed, along with the mathematical principles of the IMU. Additionally, the EKF algorithm will be explained, and its role in improving the accuracy of the foot-mounted IMU will be highlighted.

These methods will be used to estimate the position and orientation of a pedestrian in indoor environments, with the goal of achieving higher accuracy than traditional methods. In the second section, the methodology regarding Blockchain will be explained. This will include a discussion on the fundamentals of blockchain technology, Hyperledger-Fabric, and its role in securing data in the context of the pedestrian positioning system described in the first section.

2.3.1 Indoor Navigation

This section aims to provide a brief overview of the methodology employed in the navigation section. The discussion will encompass a brief analysis of the ZUPT equations and their mathematical foundation, as well as the fundamental mathematical principles underpinning the IMU. Additionally, a brief explanation of the EKF algorithm will be presented. Furthermore, the developed server and Ground Control Station (GCS) will be elucidated, with a particular emphasis on their functionality in the context of this research. This will facilitate a deeper understanding of the various components involved in the navigation system and their roles in enabling accurate and reliable navigation.

2.3.1.1 Inertial Measurement Unit (IMU)

To discuss the selection of an appropriate IMU for the pedestrian motion tracking system, various factors such as size, weight, power, and cost (SWaPC) were taken into consideration. In this regard, we have chosen to use two MPU-20948 from InvenSense. This is a MEMS-based IMU that combines the 3-axis accelerometer, 3-axis gyro and the 3-axis magnetometer, making it a low-cost 9-DOF IMU. One of the advantages of this IMU is its small size and low power consumption. Table 2.1 illustrates the specifications of this IMU, which provides inertial data at a sampling rate of 100 Hz.

Table 2.1 Specifications of the ICM-20948

Specifications	**Accelerometer**	**Gyroscope**	**Magnetometer**	**Barometer**
Measurement Range	±2 g, ±4 g, ±8 g, ±16 g	±250 dps, ±500 dps, ±1000 dps, ±2000 dps	±4900 µT	260–1260 hPa
Sensitivity	16,384 LSB/g	131 LSB/dps	0.15 µT/LSB	0.01 hPa/LSB
Noise Density	200 µg/√Hz	20 mdps/√Hz	0.15 µT/√Hz	0.01 hPa/√Hz
Output Data Rate (ODR)	4 Hz–1.12 kHz	4 Hz–8 kHz	4–100 Hz	1–200 Hz
Current Consumption (Operating)	450 µA	1.2 mA	90 µA	0.9 mA

The IMU was placed on the forefoot in this study, which has been proven to be a more favorable mounting position than the heel in ZUPT-aided pedestrian inertial navigation to capture the sequential nature of pedestrian motion and utilize error-reduction techniques such as zero-velocity update (ZUPT) during foot stances/swings. This is attributed to several advantages, including a longer stance phase, reduced shock levels, and decreased velocity uncertainty during the stance phase (Wang et al. 2021). We assume that the user is walking along the global x-axis , and the sensor's local (body) coordinates and global coordinates are shown in Figure 2.1. Our algorithm is developed irrespective of the exact position and orientation of the INS on the user's shoe.

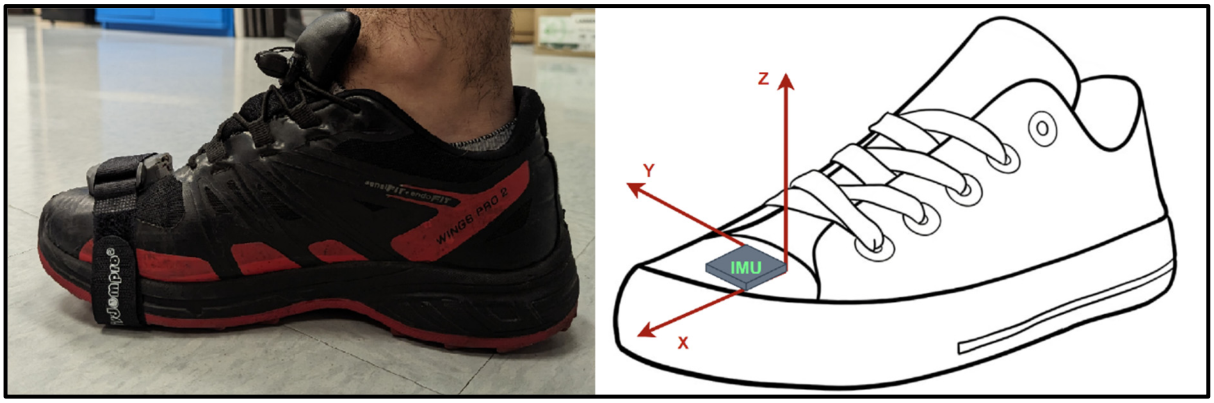


Figure 2.1 Overview of the placement and alignment of the ibNav IMU on the user's foot

Aggarwal, in (Aggarwal 2010) , provides the simplified mechanization equations for IMU. In these equations, n and b represent the navigation frame and sensor body frame, respectively. The specific force vector of the accelerometer is represented by f^b , and the angular rate vector of the gyroscope is represented by w_{ib}^b. The position vector and velocity vector in the earth-centered earth-fixed frame are represented by p^n and v^n. The transformation matrix from the body frame to the navigation frame, as a function of attitude components, is represented by C_b^n. The skew-symmetric matrices of the gyrosco pe-based angular rate vector are represented by $w_{ib}^b X$. Finally, the gravity vector in the navigation frame is represented by g^n.

$$\begin{bmatrix} p^n \\ v^n \\ C_b^n \end{bmatrix} = \int \begin{bmatrix} v^n \\ C_b^n f^b + g^n \\ C_n^b\left(w_{ib}^b X\right) \end{bmatrix} \tag{2.1}$$

In this study, we make the assumption that the initial velocity is zero, and it is further assumed that the initial position is already known. The starting angle is then calculated using the IMU, which operates based on the following principles:

$$\left\{ \begin{matrix} \phi = \arctan\dfrac{g_y^b}{q_z^b} \\ \psi = \arctan\dfrac{g_x^b}{\sqrt{\left(g_y^b\right)^2 + (g_z^b)^2}} \end{matrix} \right\} \tag{2.2}$$

The precise determination of attitude angles is crucial. In this regard, the gravity components g_x^b, g_y^b, and g_z^b in the body frame can be utilized to determine the roll angle ϕ and pitch angle ψ of the attitude. It is noteworthy that the accelerometer readings predominantly reflect the gravity components when the velocity is zero, indicating that they remain independent of linear acceleration. As a result, the horizontal attitude angles can be reliably estimated from the accelerometer readings. Following the determination of the horizontal angles, the magnetometer can be aligned with the horizontal position to facilitate the precise calculation of the heading.

$$\left\{ \begin{matrix} m_x = m_x^b \cdot \cos\psi + m_y^b \cdot \sin\phi \cdot \sin\psi + \sin\psi \cdot \cos\phi \cdot m_z^b \\ m_y = m_y^b \cdot \cos\phi - \sin\phi \cdot m_z^b \end{matrix} \right\} \tag{2.3}$$

where m_x and m_y stand for the components of magnetic measurements in the horizontal direction, and m_b^x, m_y^b, and m_z^b are the magnetometer measurements in the sensor frame. The heading angle can be calculated as follows:

$$\alpha = \arctan\frac{m_x}{m_y} \tag{2.4}$$

Figure 2.2 provides a high-level overview of the architecture in the system used for dual-IMU ZUPT-aided navigation.

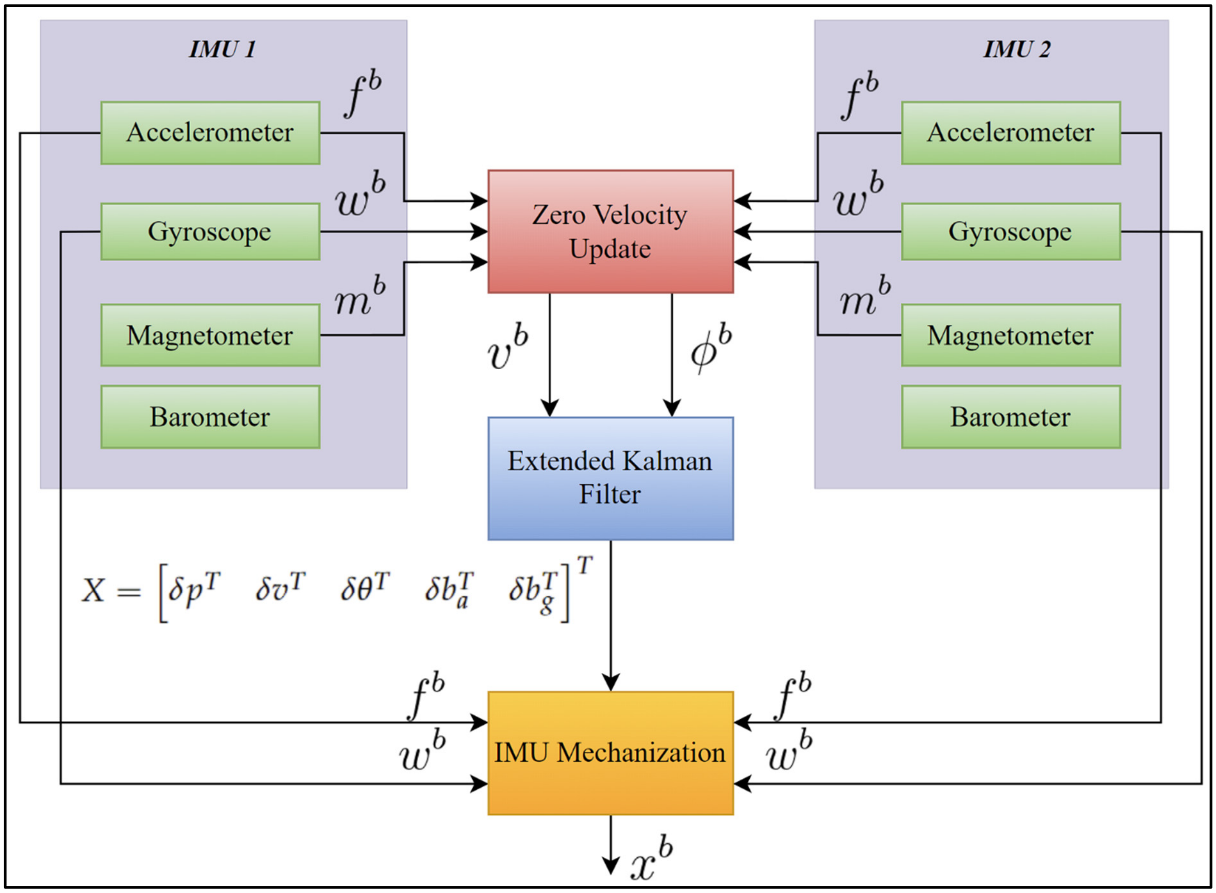


Figure 2.2 Overview of the Inertial Navigation Architecture with ZUPT and EKF for Dual-IMU

2.3.1.2 Zero Velocity Update (ZUPT)

The zero-velocity update (ZUPT) is a technique used in navigation systems to improve the accuracy of position estimates by detecting when a pedestrian or vehicle is stationary and correcting any drift errors in the sensor measurements during that stationary period The zero-velocity update (ZUPT) method, which uses an inertial measurement unit (IMU), can greatly increase the accuracy of location estimates during stationary periods, where the pedestrian's or vehicle's velocity and angular velocity are close to zero. The ZUPT technique compares the magnitude of acceleration and angular velocity measurements to a threshold value in order to solve a binary hypothesis testing issue. The position estimate is updated with zero-velocity information if the magnitude is less than the threshold number, at which point the pedestrian or vehicle is assumed to be stationary. This is crucial since sensor drift can cause substantial positioning errors, necessitating the use of zero-velocity data. While ZUPT is effective in most walking scenarios, it may not be suitable for activities such as running, jumping, and climbing due to its threshold-based approach (Zemam and Benzerrouk 2010). Using the collected data, the ZUPT algorithm performs initial alignment, monitors acceleration to detect stationary periods, estimates gyroscope bias, corrects for bias in angular velocity measurements, integrates position estimates to correct for drift, and repeats the process for subsequent motion intervals. By minimizing drift, utilizing stationary periods, suppressing noise, and offering real-time updates for accurate position estimate, this approach increases accuracy.

2.3.1.3 Extended Kalman Filter (EKF)

To address the issue of drift in navigation outcomes caused by the nonlinear system model used in the mechanization algorithm, we employ the extended Kalman filter (EKF). Unlike the standard Kalman filter, the EKF is capable of dealing with nonlinear models by approximating them through linearization around the current state estimate. The EKF can handle nonlinear models, making it well-suited to fusing additional update information and reducing drift in the mechanization algorithm's navigation results. The mechanization algorithm's navigation outcomes can drift apart quickly over time because of inertial sensor noise and system model inaccuracies. To alleviate this divergence, the extended Kalman filter (EKF) is typically employed to blend additional update data (Tao et al. 2022). To implement the ZUPT in the EKF, it is crucial to accurately identify the stationary state of the user's foot during the stance phase. If this is not done correctly, the velocity estimate error would drastically increase, leading to a quadratic growth in the position estimate error and state error vector over time (Norrdine et al. 2016). The state error vector is stated as follows:

$$X = \begin{bmatrix} \delta p^T & \delta v^T & \delta \theta^T & \delta b_a^T & \delta b_g^T \end{bmatrix}^T \tag{2.5}$$

where δp , δv, and $\delta \theta$ are the errors of the location, velocity, and attitude errors in equation (2.1. The bias errors of the accelerometer and gyroscope are represented by δb_a and δb_g, respectively. The system model in discrete form can be shown as follows:

$$\begin{Bmatrix} X_k^- = \Phi_k X_{k-1} + G_k W_{k-1} \\ P_k^- = \Phi_k P_{k-1} \Phi_k^T + G_k Q_{k-1} G_k^T \end{Bmatrix} \tag{2.6}$$

The terms X_k^- and P_k^- denote the anticipated state and its corresponding covariance matrix. The matrixΦ_k represents the transition of the state from the previous epoch $(k-1)$ to the current epoch (k). G_k is the matrix that determines the effect of state noise. W_{k-1} and Q_{k-1} are the vector and matrix, respectively, which describe the system noise. The matrix F_k can be calculated using the state differential formula in Equation (2.1 and the bias error model. Finally, the discretized state transition matrix can be obtained through the following process(Tao et al. 2022) :

$$\Phi_k = I + F_k \cdot T_s = \begin{bmatrix} I_3 & I_3 \cdot T_s & 0_3 & 0_3 & 0_3 \\ 0_3 & I_3 & (f^b \times) \cdot T_s & C_b^n \cdot T_s & 0_3 \\ 0_3 & 0_3 & I_3 & 0_3 & -C_b^n \cdot T_s \\ 0_3 & 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \tag{2.7}$$

The discretization time interval of the state equation is denoted by T_s. The matrix $f^b \times$ is a skew-symmetric matrix formed from a specific fore vector. The state noise gain matrix G_k and system noise covariance matrix Q_{k-1} are defined with explicit detail as (Tao et al. 2022):

$$G = \begin{bmatrix} 0_3 & 0_3 & 0_3 & 0_3 \\ C_b^n \cdot T_s & 0_3 & 0_3 & 0_3 \\ 0_3 & -C_b^n \cdot T_s & 0_3 & 0_3 \\ 0_3 & 0_3 & I_3 \cdot T_s & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \cdot T_s \end{bmatrix} \tag{2.8}$$

$$Q = \begin{bmatrix} I_3 \cdot \omega_f^b & 0_3 & 0_3 & 0_3 \\ 0_3 & I_3 \cdot w_\omega^b & 0_3 & 0_3 \\ 0_3 & 0_3 & I_3 \cdot \widetilde{\omega}_f^b & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \cdot \widetilde{\omega}_\omega^b \end{bmatrix} \tag{2.9}$$

The terms I_3 and 0_3 represent a 3 × 3 diagonal matrix with ones on the main diagonal and a matrix filled with zeros, respectively. ω_f^b and ω_ω^b denote the noise present in the measurements taken from accelerometers and gyroscopes. Similarly, $\widetilde{\omega}_f^b$ and $\widetilde{\omega}_\omega^b$ refer to the noise that influences the accelerometers and gyroscopes (Aggarwal 2010).

2.3.1.4 Hardware and Connection Setup

The ibNav 6.1 sensor is specifically designed for indoor navigation and is equipped with a dual-IMU setup, providing accurate and reliable navigation data, as discussed in the previous sections. To collect and transmit the navigation data, we utilized the Message Queuing Telemetry Transport (MQTT) protocol, which is a lightweight and efficient communication protocol commonly used in IoT applications.

The collected data were transmitted to a Raspberry Pi 4, which served as a gateway device, via the Wi-Fi wlan0 connection. To ensure the reliability and stability of the data transmission, we used a separate Wi-Fi dongle and established a connection via MQTT wlan1 to transmit the processed data to our server, named RiCF, and Blockchain for storage and analysis purposes. The use of blockchain technology in this context provides an added layer of security and transparency, ensuring the integrity of the data using Proof Of Location (POL). The POL mechanism involves cross-referencing the data from multiple sources to verify the device’s location within the indoor environment. This added layer of validation helped to eliminate any potential errors or inaccuracies caused by data being transmitted from a different location.

Ground Control Station (GCS) plays a critical role in the operation of the system, providing a centralized platform for data visualization and the monitoring of system performance. With the GCS, the platform can receive and display data from RiCF and Blockchain in real-time, allowing the system to quickly identify and analyze any issues or anomalies in the data. The GCS provided a user-friendly graphical interface that enabled us to easily visualize the data and gain insights into the performance of the system. Additionally, the GCS allows users to remotely control the system and adjust various parameters, such as data transmission rates and processing algorithms, to optimize the performance of the system. Overall, the GCS is an essential component of the system, providing a powerful tool for real-time data analysis and system management.

Figure 2.3 illustrates the system architecture, providing a clear and concise overview of the components and their interactions.

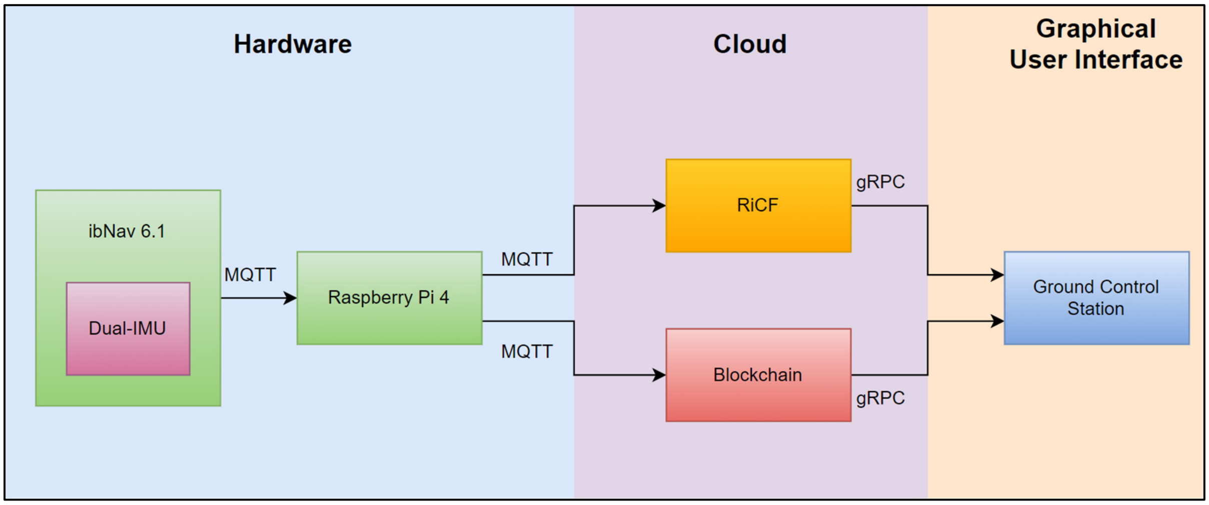


Figure 2.3 A comprehensive data flow architecture setup

2.3.1.5 RiCF Server

RiCF (RPNT-ibNav Control framework) is a cross-platform Python server that is intended to consolidate and simplify communication between all Resilient Positioning and Timing (RPNT)-ibNav components including ibNav 6.1 sensors and Raspberry Pi and, on the other interface, transmit it to the Ground Control Station (GCS). The RiCF server centralizes communication between such different components with diverse applications for compatibility. The MQTT Protocol is the communication link between RiCF and the components while RiCF and Ground Control Station communicate by Google Remote Procedure Call (gRPC). In contrast to MQTT, gRPC is developed to support several languages, improve application compatibility, and use Google's Protocol Buffer to specify structured data for communication. When using gRPC, server, and client communication is seamless, enabling clients to call server functions directly as if they were on a local object.

On the contrary, the MQTT protocol is lightweight and ideal for the ibNav system used by users with Raspberry Pi, Lidar, and inertial sensors. However, it is not suitable for a server-based system since it is a messaging protocol without a payload format specification, and interoperability between applications depends on prior agreement on a format. Additionally, MQTT is a publish/subscribe protocol, and the broker sends messages to all subscribers of a particular topic without differentiating clients. To provide a more detailed explanation of data processing in RiCF, Figure 2.4 is presented for reference. The users' data, collected from inertial sensors, are transmitted via MQTT through the local network using the WiFi bridge, received by the RiCF server, and logged into a corresponding CSV file for post-processing purposes. Simultaneously, the messages are processed and formatted to be utilized as gRPC messages. Real-time data streaming is then transmitted to GCS for visualization and demonstration.

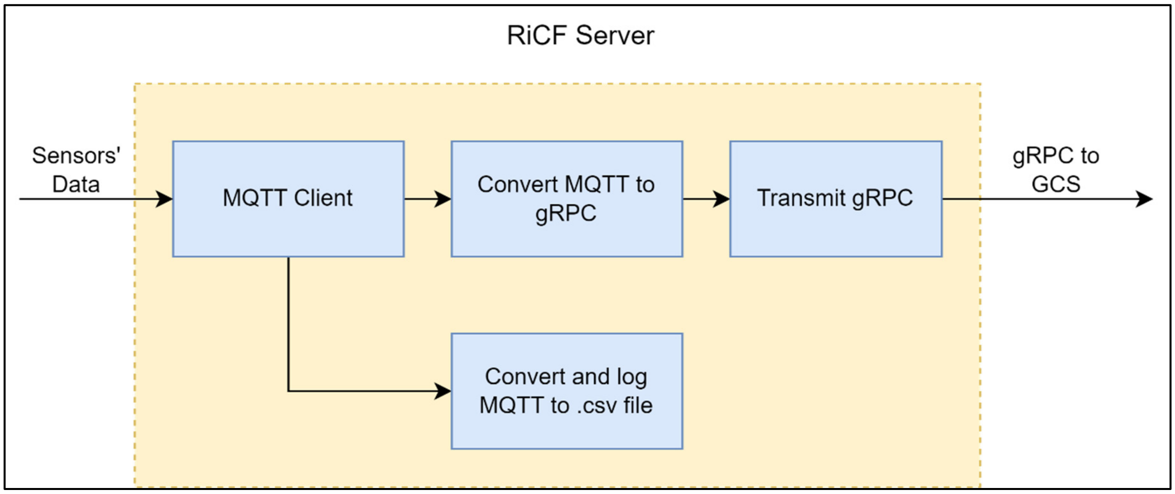


Figure 2.4 A general architecture of RiCF

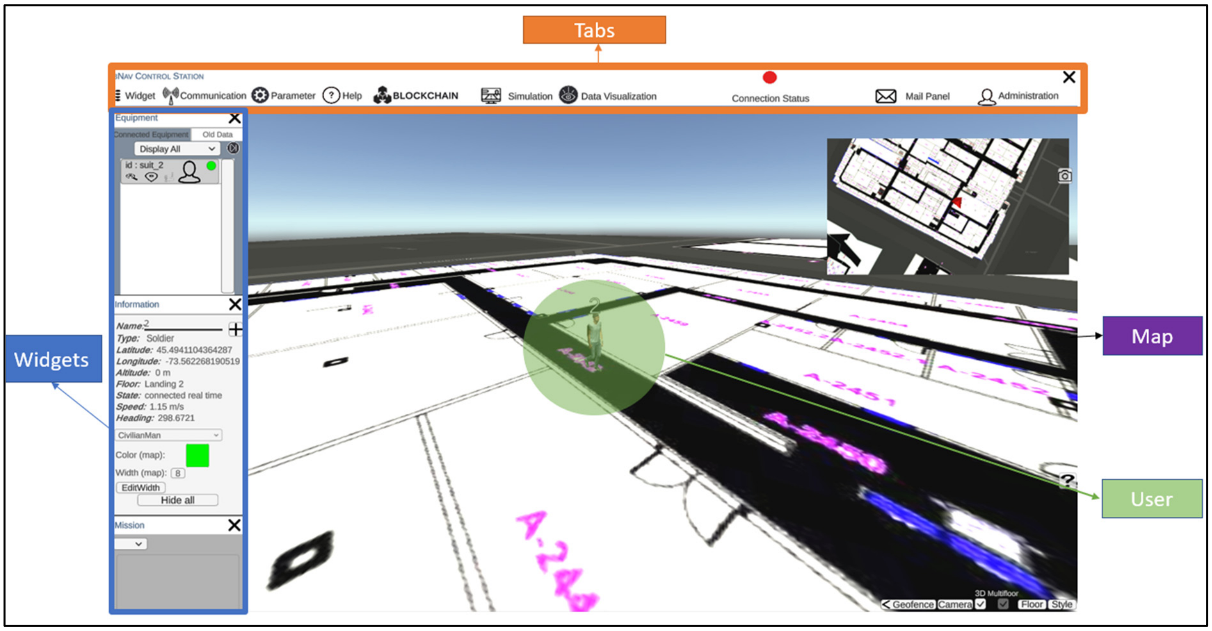
2.3.1.6 Ground Control Station (GCS)

The Ground Control Station displays information to users to identify and direct activities, including human-controlled UAVs. It receives and displays data from the IMU setup worn by users. GCS has an easy and practical application design. In order to optimize the design of the GCS, it is important to prioritize the main functionalities to be easily accessible to the operator. Less frequently used functionalities should also be included but hidden from view to minimize distractions and maintain the operator's focus. This approach can enhance the overall usability and effectiveness of the GCS.

Additionally, it is essential for a GCS to be equipped with the ability to automatically open relevant panels upon receiving emergency messages, to ensure prompt notification to the operator without requiring any manual action. The communication between the client application and server, facilitated by the gRPC protocol, allows the application to make requests to the server, such as retrieving information or reading a file. This communication also enables the application to access the various features available on the server. The server sends data to the application following a specific structure, where a message can consist of multiple elements.

The gRPC protocol implemented in the application allows for it to receive and identify each element of the message sent by the server. The global view of the GCS is presented in Figure 2.5. GCS offers a wide range of beneficial applications, including its capacity for facilitating collaborative indoor navigation.

With the integration of diverse tools for navigating indoors and drones, users can explore uncharted territories within this field using the GCS platform. Despite its limited development in collaborative indoor navigation methods, GCS remains prepared to provide full support for their implementation while offering exciting possibilities for advancing knowledge.



id : suit_2

Name: 2
Type: Soldier
Latitude: 45.4941104364287
Longitude: -73.562268190519
Altitude: 0 m
Floor: Landing 2
State: connected real time
Speed: 1.15 m/s
Heading: 298.6721

Color (map): 🟩
Width (map): 8

Figure 2.5 A global view of the Ground Control Station

GCS has several tabs and functionalities to control and visualize the navigation for the user. The widgets tab includes the equipment, alerts, errors, information and mission windows. Communication functionality is to start or end a real-time navigation test, although playback tests are available for developers. The parameters tab allows the user to connect to the desired IP address.

The blockchain tab sends requests to the IPFS to request stored data in the Blockchain. A notable feature in GCS is the trajectory feature that shows the selected user's route on the map. GCS can display the trajectory of equipment based on the floor selected by the operator on the map, especially when the equipment is indoors. Moreover, the application provides a multi-floor reference event in 2D, where only the trajectories on the appropriate floor are displayed. Multiple trajectories may be displayed if the equipment returns to the same floor multiple times or only a single trajectory if it travels to one floor and does not return.

The GCS platform boasts a potential ability to generate an environment map through the utilization of Simultaneous Localization and Mapping (SLAM) algorithms alongside appropriate sensors, including LiDAR and camera. However, this particular feature has yet to be implemented, and thus may serve as a promising avenue for future research and development.

2.3.2 An Overview of Blockchain Technology

The concept of blockchain was first introduced in 2008 by a person using the pseudonym “Satoshi Namakato”. The innovation was applied to create the world’s first cryptocurrency, Bitcoin, in 2009 (Arora and Yadav 2019). Since then, blockchain has become a buzzword in various industries, and its use cases have expanded beyond cryptocurrency. Blockchain is a decentralized ledger built on consensus principles to have a verifiable, append-only chained data system of transactions. Because of its decentralized architecture, blockchain makes it easy to store and upgrade the data, rendering blockchain a perfect architecture for ensuring distributed transactions, such as avionics networks, among all participants in a trustless setting. Appearing from the intelligence assets, a smart contract enables a team to accomplish agreements among parties via a blockchain network (Blasch et al. 2019). Based on how much confidence is needed in the network, blockchain confidentiality will be chosen for the network. Due to its short transaction latency and high throughput, the permissioned blockchain Fabric is better suited for handling IoT interactions (Xu et al. 2021).

2.3.2.1 Developing Blockchain Solutions by Hyperledger-Fabric

Hyperledger Fabric, launched in 2015 as part of the Hyperledger project by the Linux Foundation, is a highly modular and adaptable blockchain platform. It provides a permissioned network in which individuals must be identified and networks must be permitted. Hyperledger Fabric prioritizes the privacy and secrecy of business transactions and related data while focusing on a high transaction throughput speed and low transaction confirmation latency. The possibility of malicious code being inserted through a smart contract is reduced in this permissioned setting. Hyperledger Fabric, like other blockchain technologies, employs a ledger and smart contracts to provide a way for participants to manage their transactions (Anon n.d.).

Hyperledger Fabric is an open-source blockchain-based distributed ledger targeting enterprises, with smart contracts providing a high degree of trust. It serves as a foundation for developing distributed ledger systems, with a modular structure that is adaptable and scalable, guaranteeing strong privacy and resilience (Pešić et al. 2019).

Hyperledger Fabric's shared ledger consists of the global state and transaction log, with each participant having a copy. Smart contracts, called chaincode, interact with the ledger's world state and can be written in various programming languages. Hyperledger Fabric caters to different privacy requirements, supporting both highly private networks using channels and more open networks. Transactions in Hyperledger Fabric are recorded in the order they occur, and various consensus methods, such as PBFT and mining, can be chosen based on network relationships and structures (Anon n.d.).

The Hyperledger-Fabric blockchain comprises critical components such as domain, organizations, peers, orderers, and certificate authorities. Physical separation of the blockchain network is achieved using organization. Nodes validate and store transactions under an execute–order–validate transaction flow, and they are classified as a client, peer, and ordering nodes depending on their processing and storage capacity. Peers maintain the blockchain ledger, execute the chain code (or smart-contracts), and commit transactions to the global state. Orderer nodes provide communication channels and generate new blocks based on the consensus mechanism, while the certificate authority confirms ownership (Anon n.d.; Pešić et al. 2019). Figure 2.6 illustrates the Transaction flow in Hyperledger-Fabric.

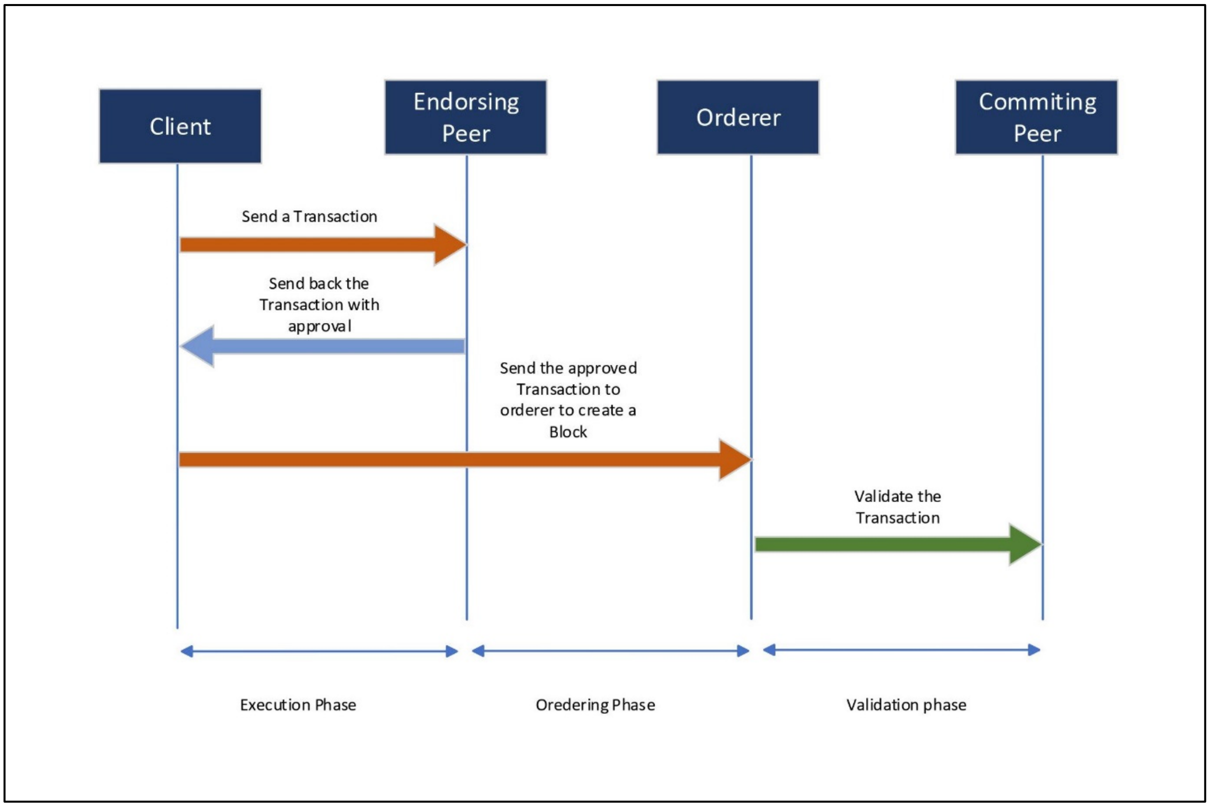


Figure 2.6 Transaction Flow in Hyperledger-Fabric

Hyperledger Fabric implements various security measures including ledger immutability, consensus mechanisms, membership services provider (MSP), secure communication, code isolation, and auditing to address vulnerabilities and enhance system resilience against attacks. These measures ensure tamper-proof transaction records, prevent unauthorized data modifications, manage identities and access control, encrypt communication, isolate code, and enable monitoring and auditing for enhanced security. Regular updates and adherence to best practices are essential for maintaining a secure Hyperledger Fabric network.

2.3.2.2 Hyperledger-Fabric Deployment

In this deployment, the Hyperledger Fabric Blockchain was leveraged as a secure and reliable database to store data generated by Raspberry Pi devices. To ensure the authenticity and integrity of the devices, each was assigned a unique public–private key pair, employing RSA

2048-bit encryption. The fabric Software Development Kit (SDK) played a vital role in this process by providing the necessary resources and libraries to establish communication between the devices and the Hyperledger Fabric Blockchain. Through the utilization of JSON Web Tokens (JWT) protected with RSA-based encryption, the SDK verified the authentication of the devices, guaranteeing secure and trusted communication.

Once the authentication mechanisms were in place, the Raspberry Pi devices could securely interact with the Hyperledger Fabric Blockchain. They were capable of submitting data transactions to the blockchain network, which were subsequently stored in the distributed ledger. Figure 2.7 provides a concise overview of the deployment process for Hyperledger-Fabric. The fabric SDK facilitated this interaction, ensuring that transactions were properly verified and certified before being included in the ledger, thereby maintaining the integrity of the data.

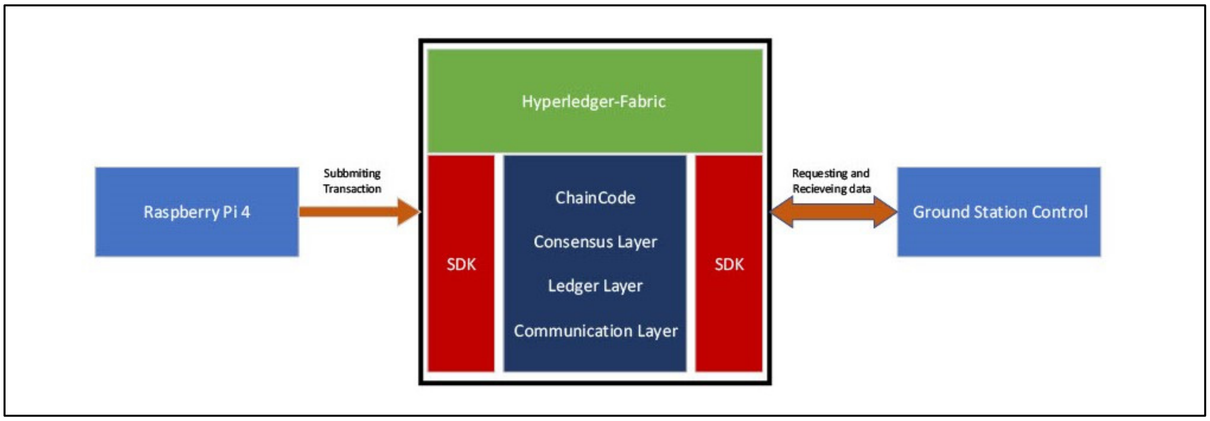


Figure 2.7 Overview of Hyperledger-Fabric Deployment

2.4 Results and Discussion

In accordance with the methodology used in this study, the following part will be divided into two separate subsections, each concentrating on a different element of the analysis. The first subsection will examine the system’s navigation performance. The emphasis of the second subsection will be on the interpretation and efficacy of the Blockchain used in the system. Building on the previous subsection's analysis, this part will look at how Blockchain can improve the system's effectiveness, security, and dependability. Furthermore, it will assess the effectiveness of the Blockchain implementation in meeting the system's particular requirements and address any possible limitations or challenges that may have emerged during the analysis.

2.4.1 Indoor Navigation Performance Analysis

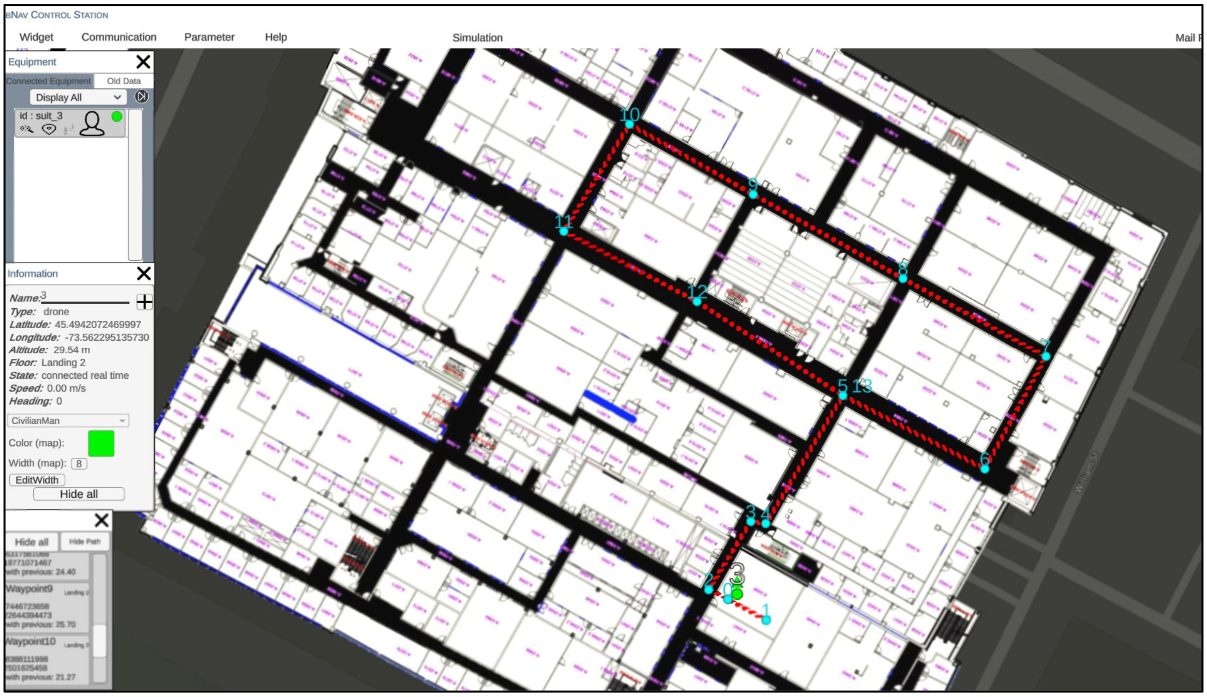
In order to evaluate the performance of our system in real-time, we conducted a series of indoor tests in École de technologie supérieure (ÉTS). The tests varied in duration from 10 to 30 min and involved providing the Ground Control Station (GCS) with a reference trajectory and an initial position. The GCS then displayed the real-time position of the user throughout the test, while an ibNav 6.1 sensor was mounted on the user's foot to collect the necessary data. Table 2.2 illustrates the selected test configuration chosen for our tests. It should be noted that data collected from the IMUs transmit to the Raspberry Pi through MQTT using Mosquitto with the frequency of 100 Hz on port 1883.

Table 2.2 Selected Test Configurations for the ICM-20948

Specifications	**Accelerometer**	**Gyroscope**	**Magnetometer**	**Barometer**
Measurment Range	8 g	1000 dps	4900 µT	1000 hPa
Low Pass Filter	Enabled	Enabled	Disabled	Disabled
Output Data Rate (ODR)	1.125 kHz	1.125 kHz	100 Hz	100 Hz

The use of an ibNav 6.1 sensor allowed us to collect navigation data, which we subsequently used to calculate the final trajectory and Root Mean Square Error (RMSE) for each test. The RMSE metric provided us with a measure of the deviation between the reference trajectory and the actual trajectory, allowing us to evaluate the system's accuracy.

In order to analyze the navigation performance of the system, a reference trajectory is given to GCS, as seen in Figure 2.8. While the reference trajectory is not mandatory when performing an indoor test, it is necessary to have a reference in order to analyze the error and performance of the proposed platform.

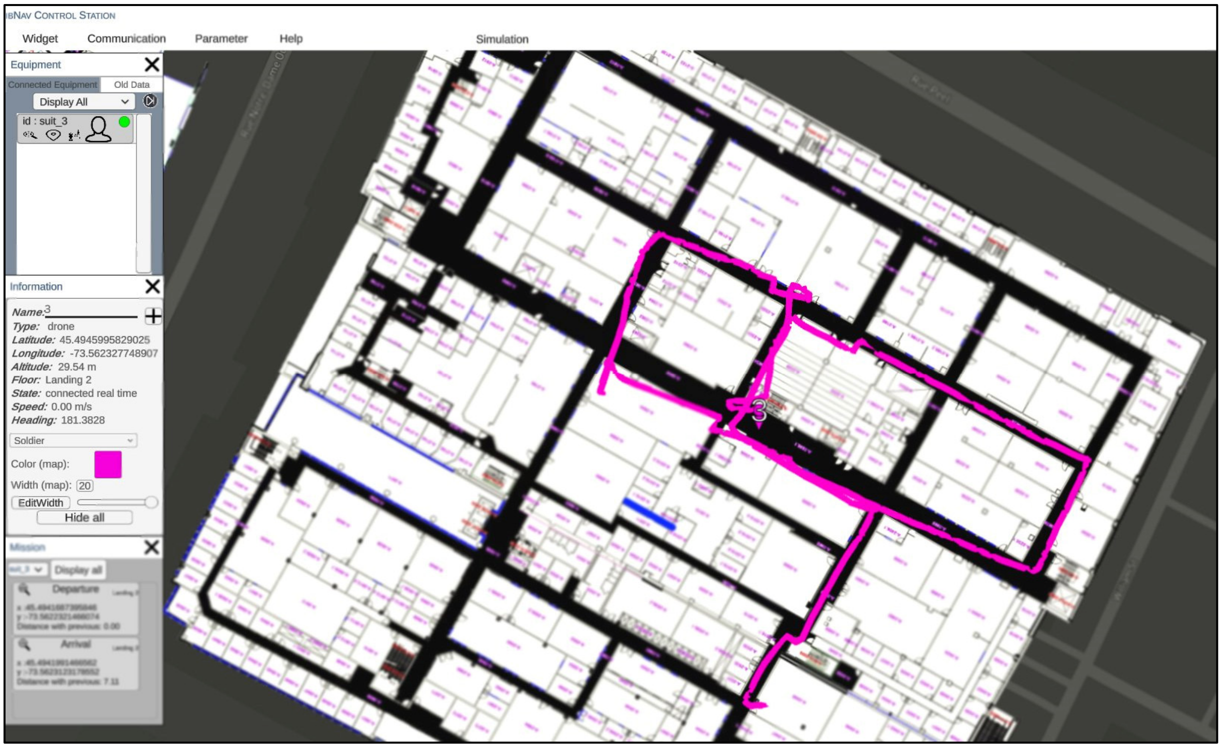


Name: 3
Type: drone
Latitude: 45.4942072469997
Longitude: -73.562295135730
Altitude: 29.54 m
Floor: Landing 2
State: connected real time
Speed: 0.00 m/s
Heading: 0
CivilianMan
Color (map): 🟩
Width (map): 8
EditWidth

6317561068
19771071467
with previous: 24.40
Waypoint9 Landing 2
7446723658
22644394473
with previous: 25.70
Waypoint10 Landing 2
8388111998
2501625458
with previous: 21.27

Figure 2.8 Visualizing a Sample Test Trajectory on GCS Platform: Map of ÉTS with Reference Path

After saving the reference points to the GCS, we started the test by launching the realtime connection. The test took 30 min and 12 s. The results of the tests are presented in Figure 2.9, and demonstrate the system's high level of accuracy and reliability. This result is shown in the GCS platform as a user interface. However to precisely calculate the RMSE error of the test, data collected from the ibNav 6.1 are saved in a CSV file with a corresponding CSV file of the reference trajectory. Then, MATLAB software was used to calculate the RMSE of the whole test. Figure 2.10 illustrates the RMSE over the duration of the test. As shown in this figure, the mean RMSE is 1.2 m and the peak RMSE is 3.28 m.



id : suit_3

Name: 3
Type: drone
Latitude: 45.4945995829025
Longitude: -73.562327748907
Altitude: 29.54 m
Floor: Landing 2
State: connected real time
Speed: 0.00 m/s
Heading: 181.3828

Soldier

Color (map): 🟪

Width (map): 20

suit_3

Departure Landing 2
x: 45.4941687395846
y: -73.5622321466074
Distance with previous: 0.00

Arrival Landing 2
x: 45.4941981466562
y: -73.5623123178552
Distance with previous: 7.11

Figure 2.9 Visualizing a sample test trajectory on GCS platform: map of ÉTS with reference path

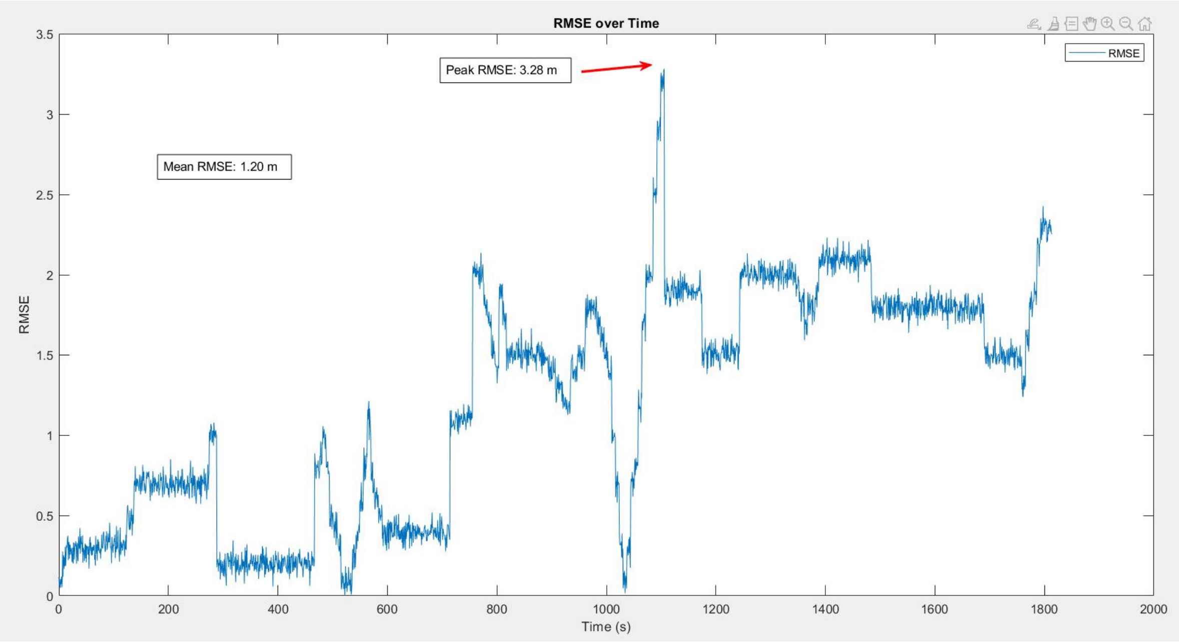


Figure 2.10 RMSE analysis over a sample test of 30 min

A number of significant insights were gleaned from an in-depth examination of the collected data, shedding light on the system's performance under real-time situations. The results unambiguously indicate the system's astonishing capacity to track the user's position in real-time and navigate the environment with exceptional precision.

It is commonly acknowledged that heuristic drift and numerous environmental conditions can provide substantial obstacles, hindering the performance of navigation systems that rely on Inertial Measurement Units (IMUs). Nonetheless, the technology under examination overcomes these challenges and achieves an outstanding performance that exceeds current restrictions.

However, it is important to recognize the experiment's scope, which is mostly focused on easy walking settings, ignoring the incorporation of more difficult movements such as running and jumping. As a result, the results of the experiment should be evaluated in this context. The lack of a comprehensive analysis of these complicated scenarios requires additional research to determine the system's abilities and potential limitations in a variety of dynamic settings. Delving deeper into the examination of sophisticated scenarios involving complex actions, such as running and jumping, undoubtedly has tremendous promise for future research endeavors. This investigation would not only improve our understanding of the system's behaviour under stress, but would also open the door to new possibilities and enhancements, assuring the system's continual evolution and optimisation in realworld applications.

2.4.2 Blockchain Performance Analysis in Hyperledger Fabric

Following the successful deployment, the stored data within the Hyperledger Fabric Blockchain became accessible for analysis and utilization. The Ground Control Station (GCS) played a pivotal role in this stage as it served as the central interface for accessing and retrieving data from the Blockchain. Functioning as a centralized administration and tracking system for the Raspberry Pi devices, the GCS provided users with the necessary tools to manage and handle their activities effectively. To enhance the user experience and enable efficient data analysis, a Graphical User Interface (GUI) was developed. The GUI offered a visually appealing representation of the acquired data, incorporating various graphical elements and real-time updates. This user-friendly interface facilitated easy interpretation and evaluation of the data collected from the Raspberry Pi devices. Users could gain valuable insights and make informed decisions based on the knowledge derived from the Hyperledger Fabric Blockchain.

In summary, this deployment showcased the successful integration of the Hyperledger Fabric Blockchain's robustness and reliability with the data collection abilities of Raspberry Pi devices. The utilization of the fabric SDK and the implementation of robust authentication procedures ensured secure and seamless communication between the devices and the blockchain network. The Ground Control Station (GCS) provided a comprehensive interface for data access and management, while the Graphical User Interface (GUI) enhanced data visualization and analysis, empowering users to leverage the insights provided by the Hyperledger Fabric Blockchain for informed decision-making.

In order to measure the performance of the platform, we used Hyperledger Caliper to measure the performance of Blockchain. Hyperledger Caliper is a free and open-source benchmarking tool for blockchain networks. It enables users to evaluate and quantify the performance of various blockchain implementations based on a predefined set of use cases. The program is extremely flexible, allowing customers to customize their testing scenarios to their exact requirements. Hyperledger Caliper is a vital tool for blockchain developers and system administrators looking to maximize the efficiency of their blockchain networks, thanks to its user-friendly interface and extensive functionality. The benchmark test was run on a hyperledger fabric blockchain network with three organizations, and each organization has one peer and three orderers.

The test was run on a Ubuntu machine with 16 GB RAM and an Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz. In order to eliminate any biases from our performance test, we performed three rigorous tests to ensure the stability and reliability of the results. Table 2.3 depicts the performance of a blockchain network as measured by the Hyperledger Calliper tool for writing and reading transactions in the tests. The analytical findings reveal the lowest and highest Transactions Per Second (TPS) attained during the trial. The investigation specifically examined the TPS for both the creation of a new transaction and for reading a transaction from the blockchain network.

Table 2.3 Comparison of three sets of tests using hyperledger caliper

Test Number	Min TPS	Max TPS	Avg TPS
Test 1- Read	638	673	660.846
Test 1- Create	364	602.3	411.746
Test 2- Read	630	680	656.692
Test 2- Create	360	610.3	428.623
Test 3- Read	636	677	661.077
Test 3- Create	360	587.3	433.777

The lowest recorded TPS for creating a new transaction is 360 and the highest recorded TPS is 610.3. As the number of transactions grows, so does the TPS. However, as the quantity of transactions rises, the success percentage of accepting transactions falls. Figure 2.11, Figure 2.12 and Figure 2.13 illustrate the outcomes of reading transactions from the blockchain network. Interestingly, the TPS figures for reading transactions are relatively constant, ranging from 630 to 680. This means that the blockchain network is more optimized for reading transactions than it is for producing new ones. Furthermore, when the number of transactions rises, the success percentage of accepting transactions for reading remains roughly consistent. This indicates that the network is capable of processing a high amount of read requests without affecting the accuracy of the data being retrieved.

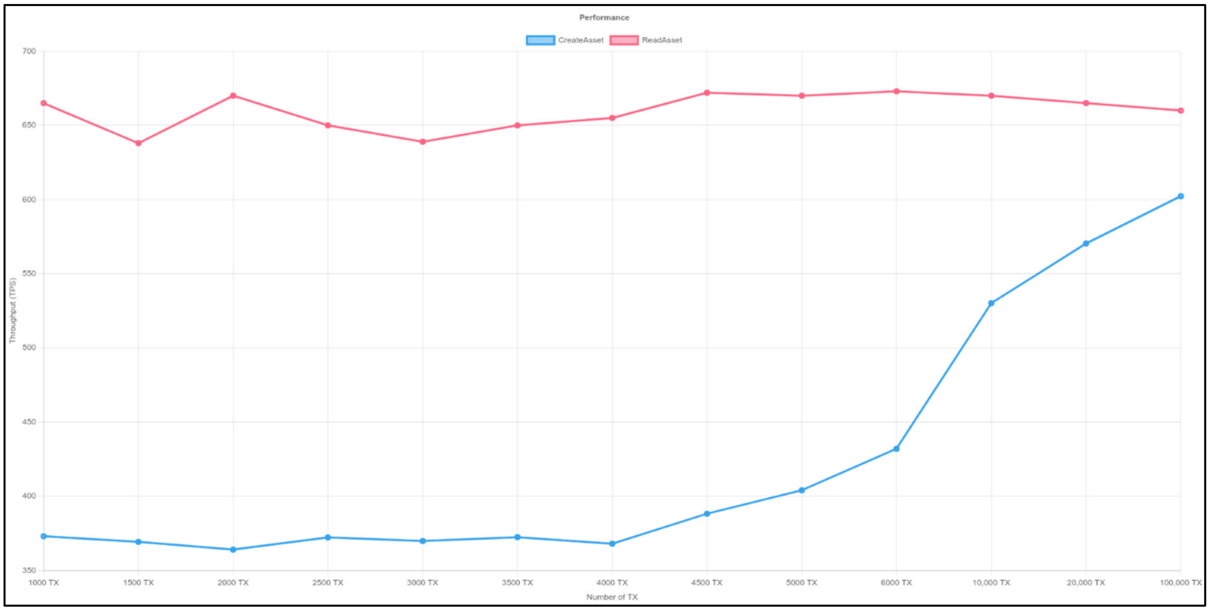


Figure 2.11 Performance of Hyperledger-Fabric- Test 1. Y-axis represents throughput (TPS) and X-axis

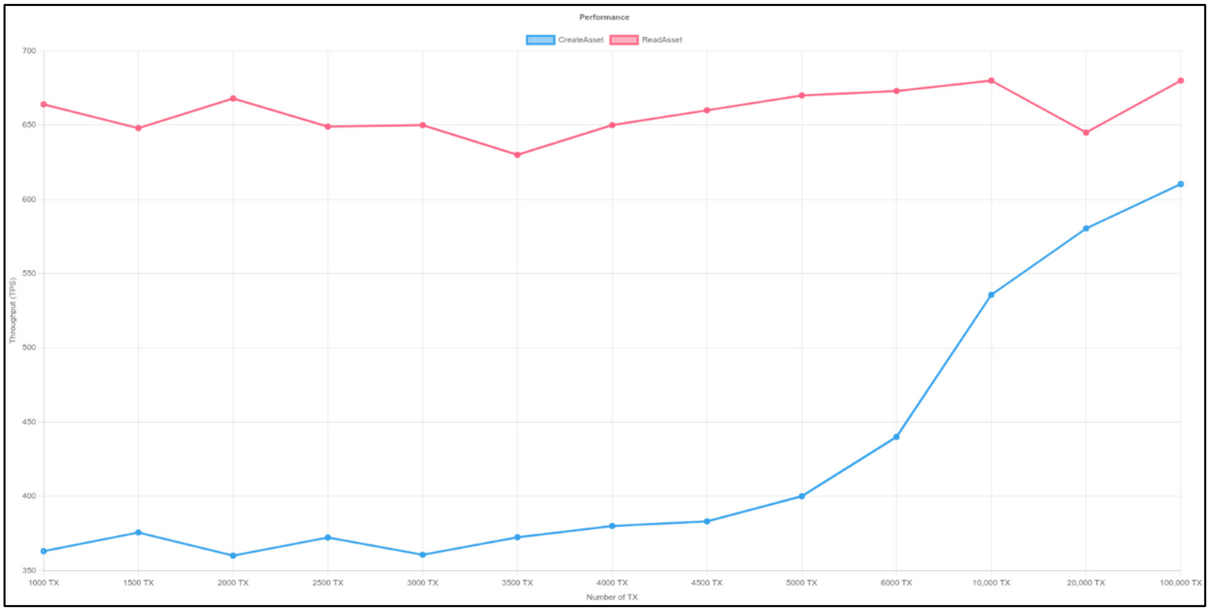


Figure 2.12 Performance of Hyperledger-Fabric- Test 2. Y-axis represents throughput (TPS) and X-axis

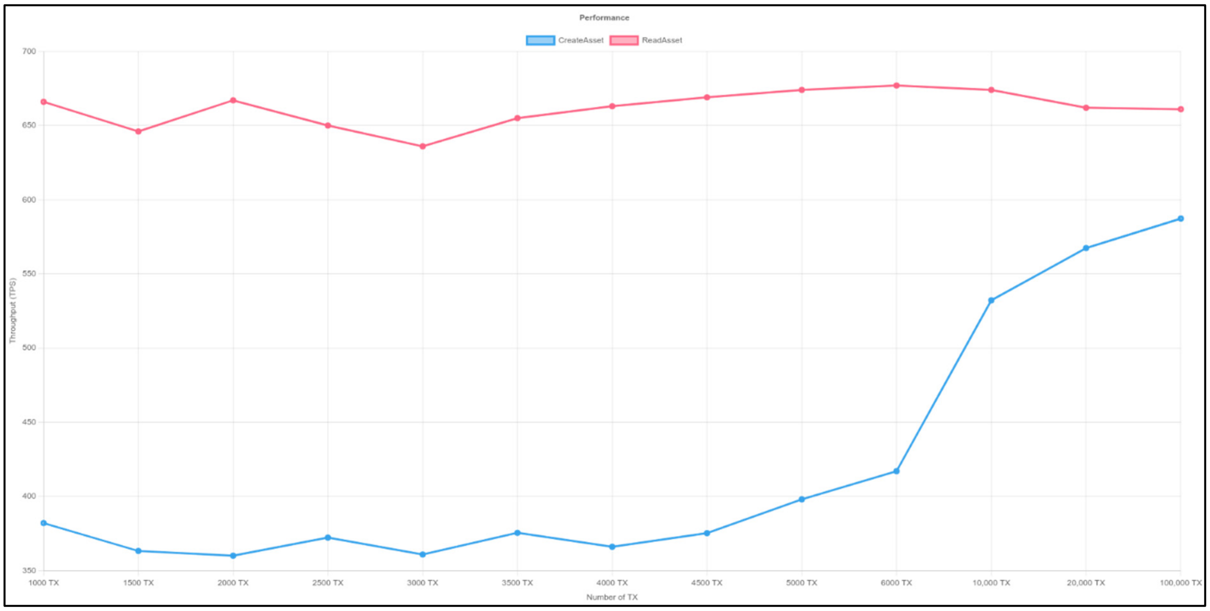


Figure 2.13 Performance of Hyperledger-Fabric- Test 3. Y-axis represents throughput (TPS) and X-axis

2.5 Conclusions

To address the challenges of providing accurate and secure indoor navigation in GNSSdenied environments, this paper proposes a novel blockchain-based indoor navigation system that integrates a foot-mounted dual-inertial measurement unit (IMU) setup and a zero-velocity update (ZUPT) algorithm. The suggested system uses an extended Kalman filter (EKF) to combine data from two IMUs and the ZUPT method to rectify the error caused by sensor drift during zero-velocity periods. By offering an immutable and distributed database to store and validate sensor data and navigation solutions, the blockchain-based design guarantees the system's security and reliability. Based on tests conducted in École de technologie supérieure (ÉTS) university department of electrical engineering, the experimental results show that the proposed system achieves high accuracy and robustness in indoor environments, with a mean RMSE error of 1.2 m and a peak RMSE of 3.28. Without depending on exterior signals or infrastructure, the suggested system provides a feasible and safe solution for interior guidance in a variety of uses such as driverless cars, robots, and human monitoring. Because of the system's ability to provide precise and secure interior navigation in GNSS-denied environments, it is a hopeful option for future indoor navigation systems that require reliable and secure navigation. When considering the scalability of the proposed platform, it is important to note that the system can be readily implemented in any environment with a WiFi connection. This is facilitated by the utilization of MQTT and gRPC protocols, which are compatible with WiFi technology. The flexibility of WiFi connectivity enables the system to be deployed in various settings, providing convenience and accessibility for users. In terms of sensor durability, the ibnav sensor, as mentioned in the paper, can sustain approximately 90 min of continuous usage. This lifespan provides a reasonable timeframe for typical applications and usage scenarios. However, it is worth noting that the duration may vary depending on specific use cases and sensor settings. Conducting further research or experimentation to evaluate the sensor's performance under different conditions would enhance our understanding of its operational limits and potential battery life improvements.

The integration of LiDAR in the future has enormous potential to improve system performance. Not only may introducing LiDAR into the system boost its abilities, but it can also lead to more accurate and dependable outcomes. Furthermore, integrating cameras with Visual SLAM can improve indoor navigation by providing a thorough and realtime understanding of the surroundings. To ensure the system's robustness, it should be rigorously tested in a variety of weather situations. The system's performance in different temperatures may be fine-tuned to work effectively and reliably in a variety of conditions, maximizing its utility and applicability. Furthermore, including augmented reality (AR) can significantly improve the system's performance. AR can overlay useful information over the user's real-world perspective, such as instructions, points of interest, or safety alerts. This integration improves not only navigation but also the overall user experience, making it more intuitive and engaging.

CHAPTER 3

CALIBRATION OF 3-AXIS LOW-COST MAGNETOMETER USING THE LEAST SQUARE ELLIPSOID FITTING ALGORITHM

Ali Shakerian [a], Saoussen Bilel [a] and René Jr Landry [a]

[a] Department of Electrical Engineering, École de Technologie Supérieure,
Montréal, QC H3C 1K3, Canada

Article published in *the International Journal of Sensors and Sensor Networks*[1],
August 2023

3.1 Abstract

This paper presents a calibration method for low-cost 3-axis magnetometers using the least square ellipsoid fitting algorithm. The aim of the calibration process is to reduce noise and mitigate the effects of magnetic interferences and instrumentation errors, thereby enhancing the accuracy and reliability of magnetometer measurements. By collecting data while moving the sensor in arbitrary directions, the calibration parameters are estimated, including magnetic disturbances (soft iron and hard iron effects) and instrumental errors (scale factor, nonorthogonality, and bias). The measured data are modeled as a combination of these errors, and the calibration parameters are obtained by solving a quadratic form equation using the least square ellipsoid fitting algorithm. The results demonstrate that the proposed calibration method using the least square ellipsoid fitting algorithm provides a valuable contribution to the field of magnetometer calibration, with the calibrated data exhibiting a better fit to the surface of an ellipsoid compared to the original magnetometer data, indicating its effectiveness, achieving 90% accuracy in magnetometer calibration of module MPU-9250. The proposed calibration method offers several advantages, including its simplicity and cost-effectiveness. Furthermore, the real-time capability of the algorithm makes it suitable for applications that require continuous calibration, ensuring accurate and reliable measurements over time. The integration of the calibration method into the intelligent IMU Sensor (IIS) further enhances its practicality and applicability in real-world scenarios.

[1] https://doi.org/10.11648/j.ijssn.20231101.13

Keywords: Magnetometer Calibration, Magnetic Interferences, Instrumentation Errors, Ellipsoid Fitting

3.2 Introduction

Magnetometers play a vital role in a wide range of applications, including navigation systems, robotics, geophysics, and motion capture. However, to ensure accurate and reliable measurements, magnetometers require calibration to correct various errors such as bias, scale factor, misalignment, and non-orthogonality. In recent years, significant research efforts have been directed toward developing calibration methods for magnetometers, particularly those with a 3-axis configuration.

Andel et al. (Andel et al. 2022) propose a calibration method for low-cost magnetometers that utilizes Global Navigation Satellite System (GNSS) data. By incorporating GNSS-derived position and velocity errors, the authors effectively correct the magnetometer measurements, thereby improving their accuracy. This approach is particularly useful in applications where accurate positioning is crucial, such as navigation systems and robotics. Cao et al. (Cao, Xu, and Xu 2020) present a real-time calibration method specifically designed for magnetometers embedded in inertial navigation systems. Their approach employs the Recursive Least Squares/Maximum Likelihood (RLS/ML) algorithm, which enables accurate and efficient calibration. By continuously updating the calibration parameters in real-time, the overall performance of the navigation system can be significantly enhanced. Hu et al. (Hu et al. 2022) introduce an automatic method for calculating the zero offset of magnetometers using the interplanetary magnetic field. By analyzing variations in the interplanetary magnetic field obtained from spacecraft, the authors estimate the magnetometer's zero offsets. This approach eliminates the need for external references and provides a self-contained calibration technique, making it suitable for space missions and remote sensing applications. Li et al. (Li et al. 2023) propose a calibration method specifically designed for strapdown magnetic vector measurement systems.

By fitting magnetic field measurements to a plane model in different orientations, they estimate errors and improve the reliability of magnetometer measurements. This approach is particularly useful in applications where precise attitude estimation is required, such as autonomous vehicles and unmanned aerial systems. Ru et al. (Ru et al. 2022) provide an overview of the current status and future trends in MEMS inertial sensor calibration technology. While not directly focused on magnetometers, their insights into calibration techniques and challenges for achieving accurate and reliable sensor measurements are valuable. The authors discuss the calibration of accelerometers and gyroscopes, providing a broader context for magnetometer calibration studies. Renaudin et al. (Renaudin, Afzal, and Lachapelle 2010) present a comprehensive calibration method for triaxial magnetometers. Their approach addresses various errors, including scale factor, misalignment, and non-orthogonality errors.

By combining calibration models and optimization techniques, they achieve accurate calibration and precise magnetic field measurements. This method is applicable to a wide range of triaxial magnetometers used in diverse applications such as geomagnetic mapping and attitude determination. Styp-Rekowski et al. (Styp-Rekowski et al. 2022) take a machine learning-based approach to calibrating magnetometers on the GOCE satellite platform. By utilizing support vector regression and random forests, they model and correct errors, resulting in improved calibration accuracy. This approach demonstrates the potential of machine learning techniques in magnetometer calibration and provides insights into the integration of artificial intelligence methods with sensor calibration. Abosekeen et al. (Abosekeen et al. 2017) conduct a comparative analysis of MEMS-based magnetometers in magnetic-based Relative Incremental Slip Sensor (RISS) systems.

By examining accuracy, precision, and reliability, their findings aid in selecting and calibrating magnetometers for RISS applications. This study contributes to the understanding of the performance characteristics of MEMS-based magnetometers in specific application contexts, helping researchers and engineers make informed decisions regarding sensor selection and calibration. Brigante et al. (Brigante et al. 2011) discuss the miniaturization of MEMS-based wearable motion capture systems. Although not directly related to calibration, their insights into design, calibration, and performance considerations are valuable for miniaturized MEMS sensor applications.

This study highlights the importance of considering calibration requirements during the design phase of wearable systems to ensure accurate and reliable motion capture data. Coulin et al. (Coulin et al. 2022) propose a method for online magnetometer calibration in indoor environments for magnetic field-based Simultaneous Localization and Mapping (SLAM). By estimating magnetometer errors in real-time during SLAM operations, their approach enhances the accuracy of magnetic field-based SLAM systems. This study addresses the practical challenges of online calibration in dynamic environments and contributes to the development of robust SLAM algorithms. Chen et al. (X. Chen et al. 2022) introduce a novel calibration method for triaxial magnetometers. Their expanded error model and two-step total least square algorithm effectively estimate and correct bias, scale factor, and non-orthogonality errors, leading to improved calibration accuracy.

This study provides an advanced calibration technique that can be applied to triaxial magnetometers used in various applications, including robotics, geophysics, and navigation systems. Ouni and Landry (Ouni and Landry 2016) focus on calibrating low-cost magnetometers using the Particle Swarm Optimization (PSO) algorithm. Their PSO-based calibration method successfully estimates bias and scale factor errors, enhancing the calibration accuracy of these devices. This approach offers a computationally efficient solution for calibrating low-cost magnetometers, making it suitable for resource-constrained applications. Pylvänäinen (Pylvänäinen 2008) discusses the automatic and adaptive calibration of 3D field sensors, including magnetometers. Through an iterative model-based approach, their method adaptively updates calibration parameters, ultimately improving the accuracy and reliability of 3D field sensor measurements. This study highlights the importance of continuous calibration in maintaining the performance of field sensors over time, especially in dynamic environments where the calibration parameters may change. Yan et al. (Yan, Zhang, and Chen

2019) propose a calibration approach for MEMS-based magnetometers in Autonomous Underwater Vehicle (AUV) navigation systems.

By utilizing the Earth's magnetic field as a reference, their method estimates magnetometer errors using field measurements collected during AUV operations. This study addresses the unique challenges of underwater navigation and provides a calibration solution specifically tailored to AUV applications. Wahdan et al. (Wahdan et al. 2014) present a fast and autonomous magnetometer calibration technique for portable navigation devices in vehicles. Leveraging the Earth's magnetic field during vehicle movements, their approach achieves accurate calibration, enhancing the reliability of magnetometer measurements. This study contributes to the development of calibration techniques suitable for real-time applications, such as navigation systems in vehicles. Micro Electromechanical Systems (MEMS) are widely used in navigation and motion capture systems because of their affordable price and their high sampling frequency.

Nonetheless, the outputs of the MEMS-IMUs, especially the magnetometers, are affected by the presence of sensor noise and by external magnetic interferences. Hence, the parameters of the magnetic distortions should be precisely estimated to reduce the drift of the measured data. In this paper, a real-time magnetometer calibration method is performed based on the estimation of two calibration parameters: the magnetic disturbances and the instrumentation errors. The errors need to be removed from the measured data to obtain an estimation of the calibrated data. This method does not require any additional equipment for measuring the data, in fact, the data is being collected while moving the sensor by the hand in arbitrary directions.

This method was simulated and developed on MATLAB and integrated into the intelligent IMU Sensor (IIS). The algorithm consists of the magnetometer’s field measurements fitting to an ellipsoid surface. Thus, the Least Square Fitting algorithm was used for this purpose. The obtained results proved that the calibrated data fits the surface of an ellipsoid unlike the original magnetometer’s data. (Renaudin et al. 2010)

3.3 Magnetometer Error Models

In this part, the model of error for a real magnetometer is described. The model consists of two main parts, the first part is called magnetic interferences, and the second one is defined as instrumentation error. Each part has its own error parameters. Table 3.1 demonstrates the parameters of error for defining the main model of the magnetometer. (Renaudin et al. 2010)

Table 3.1 Definition and type of magnetometer error

	Definition of error	**Type**
1	Soft iron effect	Magnetic interferences
2	Hard iron effect	
3	Scale factor error	Instrumental errors
4	Nonorthogonality	
5	Bias	

3.3.1 Magnetic Interferences

The local magnetic field measured by the magnetometers is disturbed by the presence of ferromagnetic materials and electromagnetic systems. These errors are presented as the hard and soft iron errors. The soft iron errors are created from the presence of the ferromagnetic materials that generate complex magnetic fields in the surrounding of the magnetometer. The soft iron error is modeled by the following matrix: (Renaudin et al. 2010)

$$A_{si} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{3.1}$$

The hard iron errors are generated by permanent magnets and magnetic hysteresis. This error is equivalent to a bias in the measured data and it is represented by the following vector: (Renaudin et al. 2010)

$$b_{hi} = \left[b_{hi_x}\, b_{hi_y}\, b_{hi_z}\right]^T \tag{3.2}$$

3.3.2 Instrumentation Errors

The instrumentation errors vary from one magnetometer to another, and they are not variable. The first instrumentation error is defined as the scale factor error. It represents a constant of proportionality which relies on the input to the output. The scale factor error is modeled by the following matrix: (Renaudin et al. 2010)

$$S = diag(S_x\, S_y\, S_z) \tag{3.3}$$

The matrix represented the misalignment errors are represented as follows: (Renaudin et al. 2010)

$$M = N^{-1} = \left[\varepsilon_x\, \varepsilon_y\, \varepsilon_z\right]^{-1} \tag{3.4}$$

The vectors of the matrix N represent the direction of the magnetometer axis according to the sensor frame. A bias is introduced by the sensor offset in the output data. This bias is modeled by the following vector:

$$b_{so} = \left[b_{so_x}\, b_{so_y}\, b_{so_z}\right]^T \tag{3.5}$$

3.3.3 Modeling of the Measured Data

The outputs of the magnetometers are affected by the combination of the magnetic interferes and the instrumentation errors. The expression of the measured data is given as follow: (Renaudin et al. 2010)

$$h_m = SM(A_{si}h + \; b_{hi}) + b_{so} + \varepsilon \tag{3.6}$$

Where h represents the ideal data that is free from the magnetic disturbances and the instrumentation errors. ε represents the Gaussian white noise. The equation can be rewritten into a more simplified equation which is expressed as follow:

$$h_m = Ah + b + \varepsilon \tag{3.7}$$

Where,

$$A = SMA_{si} \tag{3.8}$$

$$b = SMb_{hi} + b_{so} \tag{3.9}$$

A is a 3×3 matrix that represents a combination of the scale factor, misalignment and soft iron interference. On the other hand, b is a vector that represents a combination of the bias. (Renaudin et al. 2010)

3.4 Calibration Algorithm

From the equation (3.8 it can be seen that the procedure of magnetometer calibration is reduced for finding an estimation of the calibration parameters A and b. In case we neglect the Gaussian noise, the calibration equation will become: (Renaudin et al. 2010; Teslabs n.d.)

$$h = A^{-1}(h_m - b) \tag{3.10}$$

If we suppose that the measured data are recorded by an ideal tri-axis magnetometer in an environment which is free from any magnetic distortion, their norm should be equal to the norm of the earth magnetic field: (Renaudin et al. 2010)

$${H_m}^2 - \|h\|^2 = {H_m}^2 - h^T h = 0 \tag{3.11}$$

Replacing the equation (3.11 in the equation (3.10, gives us the following equation:

$$(h_m - b)^T A^{-1^T} A^{-1}(h_m - b)\ H_m{}^2 = 0 \tag{3.12}$$

The equation (3.12 can be rewritten into a quadratic form as follow: (Renaudin et al. 2010)

$$h_m{}^T Q h_m + u^T h_m + k = 0 \tag{3.13}$$

Where :

$$Q = A^{-1^T} A^{-1} \tag{3.14}$$

$$u = -2Q^T b \tag{3.15}$$

$$k = b^T Q b - H_m^2 \tag{3.16}$$

To find the values of Q, u, and k we use the least square ellipsoid fitting algorithm which fits a set of points into a surface of an ellipsoid. In this paper, the points represent the measured data of the magnetometer. This algorithm gives us as an output of the algebraic ellipsoid description (the nine elements of the equation of the ellipsoid) and the geometric characteristic of the ellipsoid (center, radii, principal axis, Volume). The following Figure 3.1 shows the block scheme of the ellipsoid fitting algorithm. (Renaudin et al. 2010)

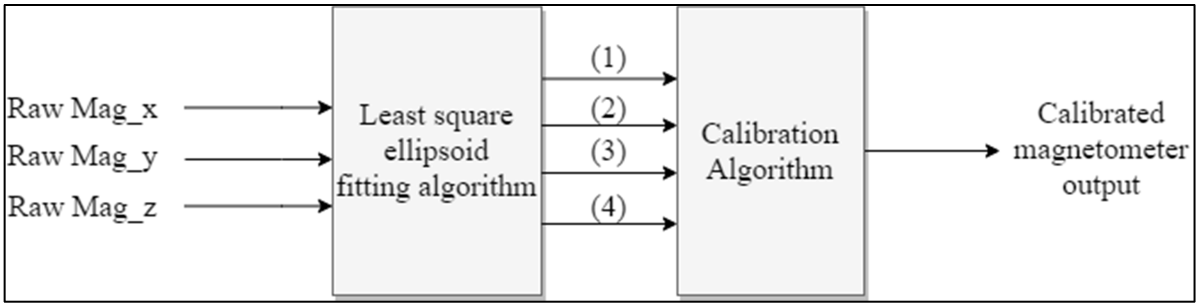

Raw Mag_x
Raw Mag_y
Raw Mag_z
Least square ellipsoid fitting algorithm
(1)
(2)
(3)
(4)
Calibration Algorithm
Calibrated magnetometer output

Figure 3.1 Block scheme of the ellipsoid fitting algorithm
1) center
2) Radii
3) Principal axis
4)V

The vector v contains the nine coefficients of the quadratic equation S of the ellipsoid. The quadratic equation is given by: (Renaudin et al. 2010)

$$S: ax^2 + by^2 + cz^2 + 2Dxy + 2Exz + 2Fyz + 2Gx + 2Hy + 2Iz = 1 \tag{3.17}$$

The nine coefficients are used to calculate the matrix Q and the vector u. The expression of the matrix Q and the vector u is given as follow: (Teslabs n.d.)

$$Q = \begin{bmatrix} a & D & E \\ D & b & F \\ E & F & c \end{bmatrix} \tag{3.18}$$

$$u = \begin{bmatrix} 2G \\ 2H \\ 2I \end{bmatrix} \tag{3.19}$$

From the estimated values of Q and u, it can be determined that the values of A and b can be calculated from the following equations:

$$b = -\frac{1}{2}Q^{-1}u \tag{3.20}$$

$$A^{-1} = \frac{F}{\sqrt{u^T Q^{-1} u} - k} M^{\frac{1}{2}} \tag{3.21}$$

After obtaining the values of A and b, we replace in the equation (3.10 to obtain the calibrated data. (Teslabs n.d.)

3.5 Online Calibration Method

The module which is utilized for the project needs to perform the calibration algorithm in real-time. As a result of that, the system needs the real-time method without any interfere or delay in performance of module. In this part, the method for performing the calibration method in real-time is designed and the results of real-time experiments are shown in the next part of this paper.

The system consists of timing block, sample recorder block and threshold parameters. The threshold parameter is defined as a sampling amount in which the sampling block will stop the storing data. It can be defined at each amount more than 1000 in according to the fact that, for accurate calibration at least 1000 sample for 10 seconds with sampling frequency of 100 Hz is needed. All the samples will be recorded in vectors in sampling recorder data and after the threshold time, the output of online calibration block will demonstrate the calibrated data. Block diagram of online calibration method is illustrated in the Figure 3.2.

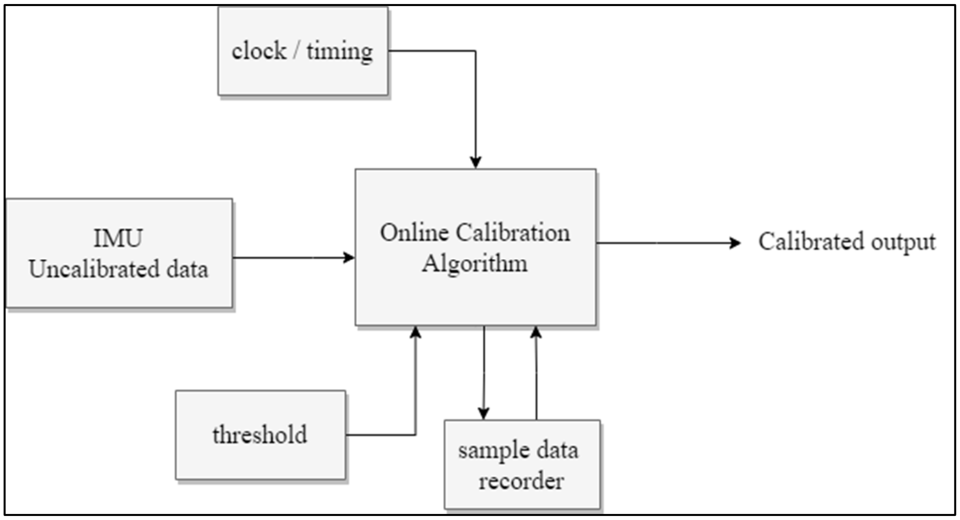


Figure 3.2 Block scheme of the ellipsoid fitting algorithm

3.6 Experimental Results

The original magnetometer data are distributed into an arbitrary shape due to the presence of the instrumentation errors and the magnetic disturbance. As an evaluation of the project, a MATLAB based algorithm is designed to test the calibration of real magnetometer. For that, the low-cost module MPU-9250 set to send its magnetometer's output in serial to record in the system. Table 3.2 illustrates the specification the MPU-9250. In this study the test was conducted in the room temperature with the magnetometer range set to ±500°/s.

Table 3.2 Specification of MPU-9250

Specification	**Value**
Accelerometer Range	±2g, ±4g, ±8g, ±16g
Gyroscope Range	±250°/s, ±500°/s, ±1000°/s, ±2000°/s
Magnetometer Range	±4800µT
Operating Temperature Range	-40°C to +85°C

After sending the data with sampling frequency of 100 Hz, the data is recorded with serial reader software.

Table 3.3 Experimental results of calibration parameters

Parameters	Calibration results
b: combined bias [µ Tesla]	[-12.069 152.87 -175.18]
A: scale factors, soft iron and misalignments [n.u.]	$\begin{bmatrix} -0.154 & 0.0141 & -0.0059 \\ 0.0141 & -0.203 & 0.0006 \\ -0.0059 & 0.0006 & -0.1584 \end{bmatrix}$

The calculated parameters of calibration A and b from the least square ellipsoid fitting algorithm are shown in Table 3.3. As described in Table 3.3, b is a vector with elements of total bias in the x, y, and z axis of the magnetometer which should be subtracted from the real raw data. After estimating the parameters of calibration with the proposed method, the experiment was performed for testing the online calibration design, the test was done in real-time, and a sampling time of 0.01 second with Simulink-based design.

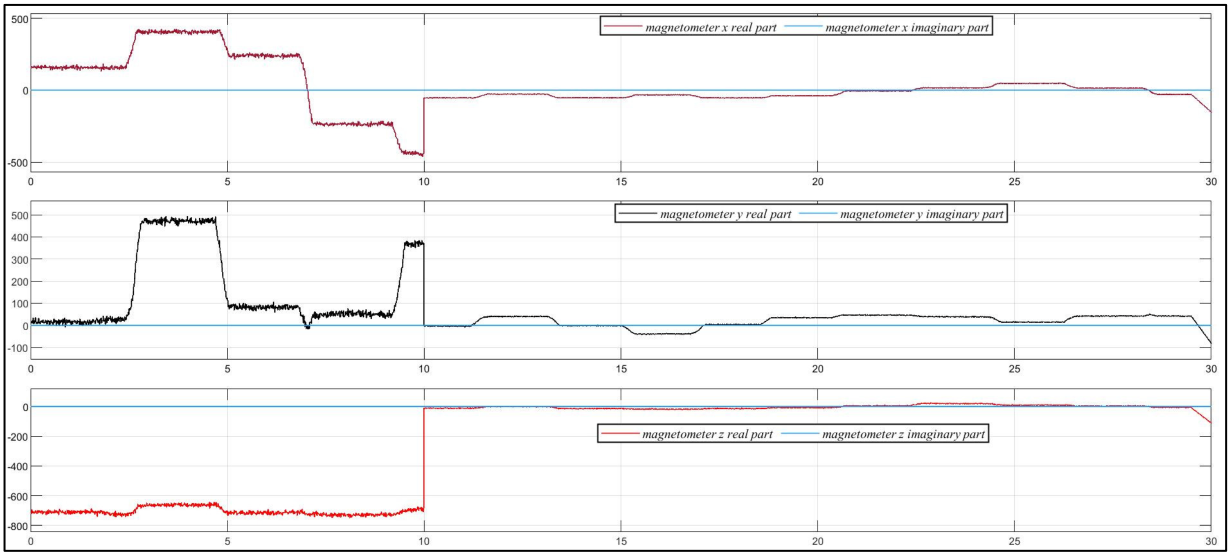


Figure 3.3 Magnetometer output after performing the online calibration algorithm

As is obvious in Figure 3.3, the plots clearly illustrate the behavior of the magnetometer's output over time regarding X, Y and Z axis. Prior to the threshold sample time of 10 seconds, the output is uncalibrated, exhibiting erratic and inconsistent values. This uncalibrated phase is denoted by a noticeable lack of stability and accuracy in the magnetometer readings. However, beyond the 10-second mark, a significant change is observed in the magnetometer's output. The plot indicates a distinct shift in the behaviour, with the output becoming calibrated. The magnetometer readings now exhibit enhanced precision and reliability, as indicated by a more consistent and accurate representation of the measured magnetic field.

The calibration process has effectively compensated for the initial inaccuracies in the magnetometer's output. Figure 3.4 presents the uncalibrated magnetometer data, depicting the raw measurements obtained prior to the calibration process. In contrast, Figure 3.5 exhibits the outcomes subsequent to the removal of specific parameters from the original magnetometer readings. Notably, the application of these calibration techniques has resulted in a significant improvement in data accuracy.

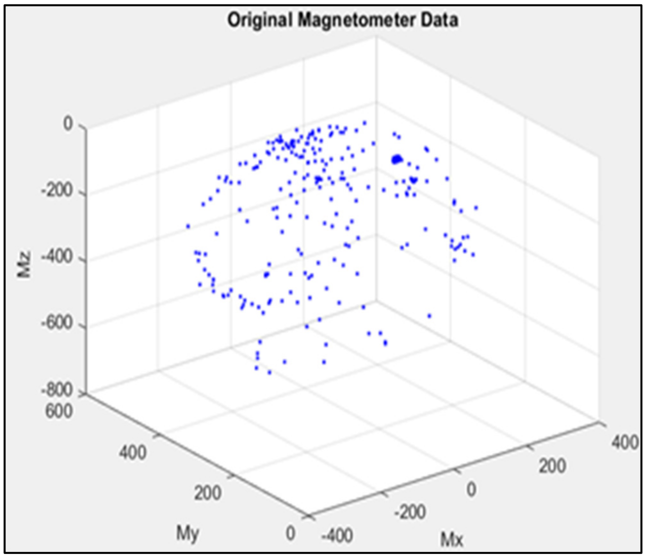


Figure 3.4 Non-Calibrated magnetometer measurements

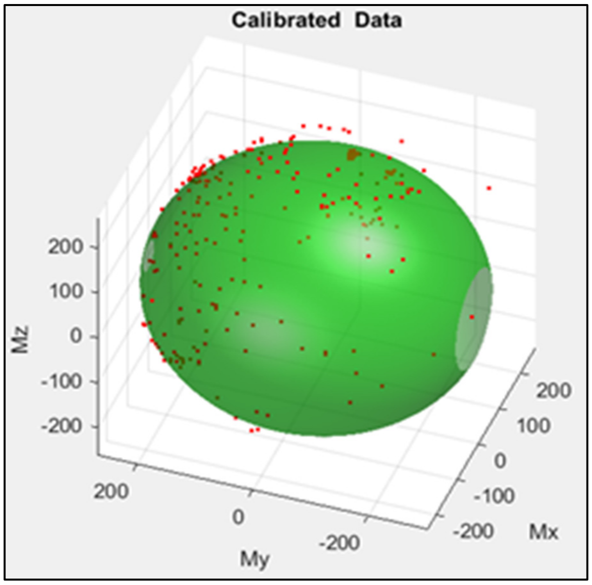


Figure 3.5 Calibrated magnetometer data

Analyzing Figure 3.5 reveals that approximately 90% of the red data points are aligned with and conform to the surface of a sphere. This correspondence between the calibrated measurements and the spherical shape indicates a calibration accuracy of 90%. This substantial level of precision is a testament to the effectiveness of the employed calibration methodology. Furthermore, the calibrated data points are distributed along the surface of an ellipsoid. The radius of this ellipsoid corresponds to the magnitude of the local magnetic Earth field, representing the strength and orientation of the magnetic field at the measurement location. By conforming to the ellipsoid's surface, the calibrated data accurately captures the characteristics of the magnetic field, allowing for more reliable analysis and interpretation of the magnetometer measurements.

Overall, the calibration process showcased in Figure 3.5 demonstrates a remarkable enhancement in the accuracy and reliability of the magnetometer data. These calibrated measurements provide a solid foundation for conducting further analysis, modeling, or applications that require precise understanding and utilization of the local magnetic field.

3.7 Conclusion

In conclusion, this paper presented a calibration method for 3-axis low-cost magnetometers using the least square ellipsoid fitting algorithm. The calibration process aimed to reduce noise and mitigate the effects of magnetic interferences and instrumentation errors, ultimately enhancing the accuracy and reliability of magnetometer measurements. By collecting data while moving the sensor in arbitrary directions, the calibration parameters were estimated, including magnetic disturbances (soft iron and hard iron effects) and instrumental errors (scale factor, nonorthogonality, and bias). The measured data were modelled as a combination of these errors, and the calibration parameters were obtained by solving a quadratic form equation using the least square ellipsoid fitting algorithm. The results demonstrated that the calibrated data exhibited a better fit to the surface of an ellipsoid compared to the original magnetometer data, indicating the effectiveness of the calibration method.

By accurately estimating and correcting the error parameters, the calibrated magnetometer data can be utilized in various applications, such as motion capture, navigation systems, robotics, and geophysics. The proposed calibration method offers several advantages, including its simplicity and cost-effectiveness, as it only requires the data collected by the magnetometer and a computer for processing. Furthermore, the real-time capability of the algorithm makes it suitable for applications that require continuous calibration, ensuring accurate and reliable measurements over time. The integration of the calibration method into the intelligent IMU Sensor (IIS) further enhances its practicality and applicability in real-world scenarios.

It is important to note that while this paper focused on low-cost magnetometers, the principles and techniques presented can be extended to other types of magnetometers with a 3-axis configuration. Additionally, future research could explore the integration of machine learning algorithms or advanced optimization techniques to further improve the calibration accuracy and robustness. Overall, the proposed calibration method using the least square ellipsoid fitting algorithm provides a valuable contribution to the field of magnetometer calibration with 90%

accuracy in magnetometer calibration of module MPU-9250, enabling more accurate and reliable measurements for a wide range of applications.

CHAPTER 4

REAL-TIME SENSOR-EMBEDDED NEURAL NETWORK FOR HUMAN ACTIVITY RECOGNITION

Ali Shakerian [a], Victor Douet [a], Amirhossein Shoaraye Nejati [a] and René Jr Landry [a]

[a] Department of Electrical Engineering, École de Technologie Supérieure,
Montréal, QC H3C 1K3, Canada

Article Published in *the journal MDPI-Sensors*[1], September 2023

4.1 Abstract

This article introduces a novel approach to human activity recognition (HAR) by presenting a sensor that utilizes a real-time embedded neural network. The sensor incorporates a low-cost micro-controller and an inertial measurement unit (IMU), which is affixed to the subject's bust to capture their movements. Through the implementation of a convolutional neural network (CNN) on the micro-controller, the sensor is capable of detecting and predicting the wearer's activities in real-time, eliminating the need for external processing devices. The article provides a comprehensive description of the sensor and the methodology employed to achieve real-time prediction of subject behaviors. Experimental results demonstrate the accuracy and high inference performance of the proposed solution for real-time embedded activity recognition.

Keywords: Convolutional neural network (CNN), micro-controller, Human Activity Recognition (HAR), Real-Time

[1] https://doi.org/10.3390/s23198127

4.2 Introduction

Human activity recognition (HAR) using wearable sensors has demonstrated remarkable achievements through the application of neural networks (Ignatov 2018; Moya Rueda et al. 2018; Zeng et al. 2014). By inputting raw data from accelerometers, gyroscopes, and occasionally magnetometers into pre-trained neural networks, it becomes possible to detect and classify human activities or motion behaviors (Choi, Jung, and Mun 2019; Silva do Monte Lima, Uchiyama, and Taniguchi 2019). However, there are notable drawbacks associated with the utilization of wearable sensors. Firstly, real-time inference on neural networks often necessitates an external processing platform, resulting in the transmission of data to a host computer. This reliance on an external device for predictions limits the potential for embedding such systems in compact and lightweight packages for practical real-world applications.

Secondly, the support for microcontrollers in popular machine learning frameworks, such as TensorFlow, has been relatively recent, accompanied by inadequate documentation and limited platform compatibility. Despite these challenges, low-cost sensors, comprising inexpensive microcontrollers and inertial measurement units (IMUs), find extensive use in various applications such as medical and health analysis (Ghasemzade and Jamshidi 2018), motion capture (Lourenço et al. 2016), and inertial navigation (Yan, Shan, and Furukawa 2017). For human activity recognition, sensors can be placed on different parts of the body to capture data using integrated inertial units. The captured data can be either stored in the sensor's memory or transmitted to an external device for real-time or post-processing analysis (Mascret et al. 2018; Ronao and Cho 2016). Alternatively, smartphones can directly process measurements from their inertial sensors (Wan et al. 2020).

Common approaches to processing this data involve traditional machine learning techniques like Support Vector Machines (SVM) (Anguita et al. 2012) or boosting (Subasi et al. 2019), employing handcrafted features (Kwapisz, Weiss, and Moore 2011), or, in more advanced systems, utilizing neural networks such as multilayer perceptrons (MLP), convolutional neural networks (CNNs), or Long Short-Term Memory networks (LSTMs). Artificial neural networks

(ANNs) are gaining popularity due to their state-of-the-art performance, capacity to learn and generalize high-level relationships in data, and their ability to process raw data without the need for handcrafted features. Typically, inference is performed on computers (Ronao and Cho 2016) or specialized systems like Nvidia's Jetson Nano, Google's Coral USB accelerator, or Intel's Neural Compute Stick 2, leveraging the processing capabilities of CPUs, GPUs, or TPUs. Inference can also be executed on non-specialized small single-board computers, such as the Raspberry Pi (Mascret et al. 2018).

In a recent study, researchers introduced an innovative wearable multi-sensor data fusion approach that combines supervised machine learning algorithms with multi-resolution time-frequency analysis of received signal strength (RSS) between wearable sensors. Their framework surpasses current state-of-the-art techniques, achieving an impressive maximum classification accuracy of 99.63% (Vidya and P 2022).

Another study focused on edge fitness and context-aware health monitoring devices, specifically designing a convolutional neural network (CNN)-based human activity recognition (HAR) system. The authors explored the impact of hyperparameters and evaluated the performance of CNN-based methods using acceleration signals from the University of California, Irvine (UCI) HAR benchmark database. They emphasized the significance of selecting appropriate activation functions and segment sizes to optimize accuracy and demonstrated the real-time feasibility of their approach using the Raspberry Pi 4 (Phukan et al. 2022).

To address the challenge of limited labeled data, researchers proposed a CapsLSTM-based model for HAR in smart healthcare (Khan, Kumar, and Kumar 2022). By leveraging accelerometer and gyroscope sensors in smartphones, they incorporated spatio-temporal information to improve recognition. The authors validated their framework on the UCI-HAR and MotionSense databases and achieved comparable test accuracies across different fractions of training data, outperforming other models such as LSTM, 1D-CNN, ConvLSTM, or CNN-LSTM (Khan et al. 2022).

In a recent survey, researchers conducted an extensive review of Wearable-Based Fall-Related Recognition Systems (WFRS). These systems are essential for monitoring and ensuring the safety of the elderly, particularly given the rising global aging population. Falls significantly affect the quality of life of older adults, leading to both physical and psychological harm. The United Nations anticipates a surge in the number of individuals aged 65 and older, increasing from 727 million (9.3%) in 2020 to 1.5 billion (16%) by 2050. According to the World Health Organization (WHO), about 28%-35% of people aged 65 and over fall each year. These falls result in unintentional injuries, such as fractures and cognitive declines, highlighting the urgency for precise and efficient Fall Recognition Systems (FRS) (J. Liu et al. 2023). Previous research has examined FRSs, focusing on sensor types and algorithms. Yet, there's a notable gap in assisting newcomers in choosing the right technology for every phase of their research. To bridge this gap, the paper reviews 48 cutting-edge research articles on WFRSs from three renowned databases. The paper delves into the entirety of WFRSs, spanning data collection to model evaluation, and assesses the advantages and limitations of various sensor counts, data preprocessing methods, feature extraction techniques, and model training approaches (J. Liu et al. 2023).

In another study, researchers compared different deep-learning techniques for HAR using ankle inertial signals (Nazari et al. 2022). They proposed several models based on artificial neural networks and achieved an average classification accuracy of up to 92.8%. Additionally, a lightweight feature extraction technique combined with pruned and quantized CNN was introduced for HAR on wearable devices. By utilizing statistical feature extraction, the authors effectively discriminated between static and dynamic activities. They employed Random Forest (RF) and CNN for specific activity classification, achieving high accuracy while reducing computational and memory consumption (Nazari et al. 2022). Another research effort explored the use of logistic regression with hyperparameter addition for HAR using smartphone sensors (Minarno, Kusuma, and Ramandita 2022). The authors' approach yielded improved accuracy results, aiming to enhance the performance of smart devices in daily life.

Additionally, a voting classifier system for HAR that combines multiple machine learning classifiers was introduced (Jindal, Sachdeva, and Kushwaha 2022). The system utilized classifiers such as Logistic Regression (LR), K-Nearest Neighbors (KNN), Random Forest (RF), Naive Bayes (NB), and Support Vector Machine (SVM). The authors evaluated the effectiveness of ensemble voting classifiers on the UCI-HAR dataset, with Voting Classifier-II outperforming other classifiers. The integration of deep learning and swarm intelligence optimization algorithms for HAR using wearable sensor data was also investigated (Helmi et al. 2023). In this research, the authors developed a robust HAR system and proposed a lightweight feature extraction approach based on a residual convolutional network and a recurrent neural network (RCNN-BiGRU). Their feature selection methods, utilizing the marine predator algorithm (MPA), demonstrated superior performance compared to other optimization algorithms.

Furthermore, another study focused on the implementation and evaluation of a real-time HAR system (H. Liu, Xue, and Schultz 2023). The researchers explored optimized machine learning techniques for HAR in cyber-physical systems, comparing various algorithms such as Random Forest, Decision Trees, and K-Nearest Neighbors with deep learning algorithms like Convolutional Neural Networks, Long Short-Term Memory, and Gated Recurrent Units. Additionally, they introduced optimization techniques such as Stochastic Gradient Descent (SGD), Adam, and RMSProp (Priyadarshini et al. 2023).

Certain applications require real-time behavior analysis without the possibility of sending data to an external device due to environmental or external constraints. Examples include analyzing athlete performance, detecting soldiers' behaviors in harsh environments, or studying diver activities underwater. In such scenarios, transmitting data at an adequate rate for external processing may not be feasible, making the adoption of sensor-embedded neural networks a compelling solution.

Real-time processing is of paramount importance in these contexts, especially when analyzing on-the-fly sensor data. Convolutional Neural Networks (CNNs) offer several distinct advantages that make them ideal for this purpose over architectures like LSTM and

CapsLSTM. One of the chief benefits is their computational efficiency. Lacking the recurrent connections of LSTMs, CNNs often demand fewer computational resources, which is pivotal for real-time applications where rapid inference is critical. Moreover, CNNs typically possess a lower model complexity when tailored for real-time purposes. They tend to be more streamlined with fewer parameters, facilitating faster model loading and quicker inference times – both essential for real-time sensor data processing.

On the other hand, LSTMs, with their intricate recurrent architecture, can often be more memory-intensive and slower in execution, especially when dealing with extensive sequences. Furthermore, CNNs excel in detecting localized patterns that are frequently observed in sensor data. This capability enables immediate recognition of emergent patterns in real-time, avoiding the waiting period associated with processing an entire sequence, a potential shortcoming of certain LSTM configurations. This article delves into real-time human activity recognition using inertial data. By harnessing a CNN and the sensor's onboard microcontroller, we enable efficient and effective inference, demonstrating the viability and superiority of this approach for real-time, on-device applications.

4.3 Materials and Methods

In this research, a wearable sensor using low-cost MEMS capable of processing inertial data using neural networks on its micro-controller was developed. A mobile application was also created to control and receive information from the sensor. Finally, a 3D printed case was made to be attached to a user's bust.

4.3.1 Hardware

In the domain of Human Activity Recognition, sensor selection plays a crucial role in system design. In one study, the researchers employed the ESP8266 sensor (ESP8266-12F), a WiFi-based sensing device, for collecting multi-dimensional received signal strength indicator (RSSI) records (Chen et al. 2021), where WiFi-based sensing coupled with hierarchical deep learning models achieved recognition accuracies of 96.45% and 94.59% for six different activities in varying environments. In contrast, another study utilizes the 3-DOF accelerometer ADXL345 for motion-based activity recognition, reporting a recognition rate of 99.2% for four key daily activities (Hong et al. 2023).

The sensor outlined in this research incorporates an ICM-20948 9-DOF IMU from InvenSense and an ESP32-WROOM-32 from Expressif Systems, both of which are compatible with the TensorFlow Lite framework for microcontrollers. As depicted in Figure 4.1, the sensor's dimensions are 4.37 cm in length, 1.2 cm in width, and 0.34 cm in height. It can be powered either by a 3.7 V 500 mAh battery or through a USB cable. To contextualize its significance, it's crucial to juxtapose our sensor with other market-available sensors in terms of specifications, price, and SWAP-C.

Our study builds upon this foundation while introducing the ibNav 6.1 sensor, which incorporates the ICM-20948. This choice offers several advantages. The ICM-20948 is a versatile and advanced motion sensor module known for its precision in capturing a wide range of motion data, including acceleration and gyroscope measurements. Its integration within the ibNav 6.1 sensor enriches our dataset with additional dimensions of information, facilitating a more comprehensive understanding of human activities in diverse real-world scenarios. By selecting the ibNav 6.1 sensor with the ICM-20948, we aim to leverage these advantages to advance the field of HAR, capitalizing on its enhanced capabilities for a more holistic and accurate recognition of human activities. Table 4.1 provides a competitive comparison analysis of sensors designed for HAR. Based on the data presented in this table, the ibNav 6.1 offers superior specifications at a competitive price when compared to its counterparts

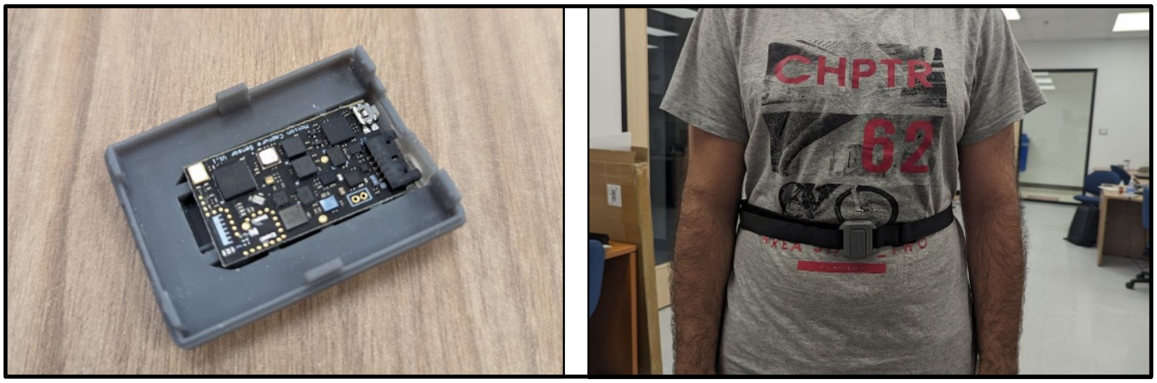


Figure 4.1 Different views of the ibNav 6.1 sensor. The board itself on the left and the sensor on the user's chest on the right

Table 4.1 Comparison of different sensors

	MetaMotionC	**MetaTracker**	**MetaMotionR**	**Notch**	**IbNav 6.1**
Manufacturer	mbientlab	mbientlab	mbientlab	notch	Lassena
Price	81.99$	60.99$	86.99$	63.16$	50$
Battery	200mAh	600mAh	100mAh	70mAh	500mAh
Memory	8MB	4MB	8MB		4MB+SdCard
Radio	BLE	BLE	BLE	BLE	BLE/BT+WiFi
Real Time Sampling	100Hz		100Hz	100Hz	100Hz
Accelerometer	X	X	X	X	2X
Gyroscope	X	X	X	X	2X
Magnetometer	X		X	X	2X
Temperature	X	X	X		X
Barometer	X	X	X		X
Humidity		X			X
Sensor Fusion	X		X		X
Enclosure	IP40	IP54	IP40	IP67	IP67
Weight	0.2oz	0.7oz	0.2oz		
Case Dimensions	25mm diameter x 4mm	52mm × 35mm x 15mm	27mm × 27mm x 4mm		43.75mm x 34mm x 12.2mm

The sensor also provides the capability of transmitting and receiving data wirelessly via the integrated onboard Wi-Fi and Bluetooth features, or alternatively, through a wired connection. In addition to the ICM-20948 and ESP32, the system necessitated a component with server functionality. This was to facilitate data reception and establish a communicative bridge between the sensor and the end-user. Moreover, this server component was required to possess adequate data storage capacity to accommodate datasets for post-processing, in addition to providing training for the convolutional neural network. The Raspberry Pi 4 Model B was selected to fulfill these needs. A summary of the ICM-20948 specifications is provided in Table 4.2.

In this study, our sensor equipped with a barometer and magnetometer provided a robust set of sensor data. However, for the specific research focus of Human Activity Recognition (HAR) using CNN and IMU sensors, we made a deliberate choice to exclude the barometer and magnetometer data. This decision was based on the fact that these sensors, although valuable for altitude measurement and orientation estimation, did not directly contribute to our central research objectives related to HAR. To streamline our dataset, reduce dimensionality, and ensure model clarity, we opted to concentrate solely on the IMU sensor data, which is a recognized and widely-used source for HAR in the literature.

Table 4.2 Specifications of the ICM-20948

Specifications	**Accelerometer**	**Gyroscope**	**Magnetometer**	**Barometer**
Measurement Range	±2 g, ±4 g, ±8 g, ±16 g	±250 dps, ±500 dps, ±1000 dps, ±2000 dps	±4900 µT	260–1260 hPa
Sensitivity	16,384 LSB/g	131 LSB/dps	0.15 µT/LSB	0.01 hPa/LSB
Noise Density	200 µg/√Hz	20 mdps/√Hz	0.15 µT/√Hz	0.01 hPa/√Hz
Output Data Rate (ODR)	4 Hz–1.12 kHz	4 Hz–8 kHz	4–100 Hz	1–200 Hz
Current Consumption (Operating)	450 µA	1.2 mA	90 µA	0.9 mA

4.3.2 System Architecture

For this system, an iOS and Android application was developed using the Unity Game Engine to control and show real-time measurements from the sensor. Both the application and the sensor connect wirelessly using WiFi to the Raspberry Pi, acting as an hotspot. For communication between components, the lightweight, publish-subscribe messaging protocol MQTT is used.

The application can control the sensor by sending simple MQTT commands to it and change, on-the-fly, various aspect of the sensor such as, activate or deactivate the activity prediction task, reset and re-calibrate the IMU and reboot the device. The application is also used to view the sensor's prediction of the activities in real-time and various statistics about the system. It is important to note that the module does not require to be connected to the other parts of the system for the HAR task and can function in standalone mode with its integrated battery. Inertial data acquisition is set at a frequency of 100 Hz, those data are then fed to the input of the convolutional neural network embedded into the micro-controller and are also transmitted over MQTT to be stored in the Raspberry Pi memory. The architecture of the system can be seen in Figure 4.2.

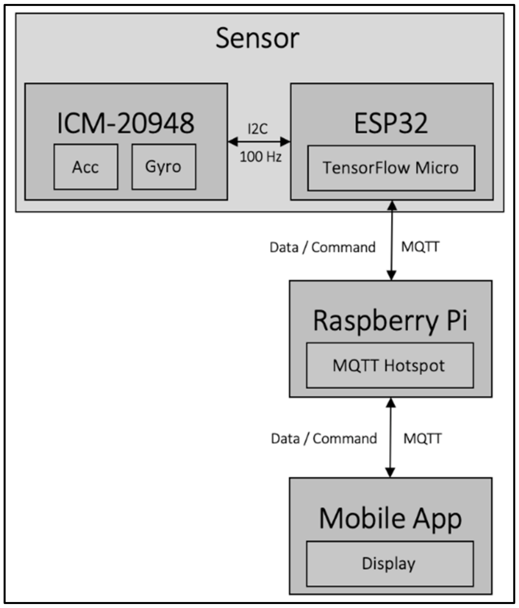


Figure 4.2 Architecture of the embedded system

4.3.3 Convolutional neural network model

For this human activity recognition task, a convolutional neural network was chosen due to its state-of-the-art results in activity classification (Wan et al. 2020) as well as in other tasks and to avoid needing handcrafted features. Additionally, The preference for Convolutional Neural Networks (CNNs) over LSTM and CapsLSTM models for our project was driven by their advantages in simplicity, faster inference times, superior local pattern detection, and efficiency on edge devices, especially with TensorFlow Lite Micro. Additionally, CNNs have a unique strength in learning spatial hierarchies, a feature not as pronounced in LSTMs. The model is composed of three 1D convolution layers with a Rectified Linear Unit (ReLu) as their activation function. Additionally, a dropout and max pooling layers are added after each of the convolutions. The output of the last convolution is then flattened and passed to the first dense layer with a ReLu activation and dropout. Finally, a last dense layer is placed at the end of the network with an output shape corresponding to the number of activities to be classified by the softmax activation function. The different layers used in the network are seen in Table 4.3 are explained next.

- *1D Convolution*: The convolution layer is used to extract features from data. It applies a convolution operation on its input using a number of filters of a set size (kernel size) to generate feature maps.
- *ReLu*: A Rectified Linear Unit is an activation function. It introduces non-linearity to the network to help learn complex mapping functions. The ReLu activation is defined as follows:

$$f(x) = max(0, x) \tag{4.1}$$

Other activations exist such as Leaky ReLu, Tanh, Swish, etc.

- *Max pooling*: A pooling layer is used to reduce the dimensionality (*downsampling*) of the feature maps and adds some translation invariance. It reduces the number of parameters and helps prevent overfitting. Max pooling is done by applying a *max filter* on the feature maps.
- *Dropout*: Dropout layers are a regularization technique to prevent overfitting by stopping neurons to learn specific features of the training data. During training of the network, a neuron is temporarily deactivated (*dropped*) with probability *p*.
- *Dense layer*: A dense or fully connected layer is a regular neural network layer where every neurons in the previous layer are connected to every neurons on the next layer. A fully connected layer expects a 1D vector as input, a flatten layer is used to flatten the output of the convolution layers.
- *Softmax*: Softmax is another activation function. It is used for classification task by normalizing its input into a probability distribution. Each class to be predicted are assigned a probability. The class with the highest probability is used as the activity predicted by the network.

More information on convolutional neural network can be found in the paper "Deep Learning" by LeCun et al. (LeCun, Bengio, and Hinton 2015).
The 3-axis accelerometer (in g's) and gyrometer (in rad/s) data from the ICM-20948 IMU are fed at a frequency of 100 Hz, as an input to the network. A timestep of 100 and a 10% sliding window are also used, resulting in an input shape of (None, 100, 6) for the first 1D convolution layer.

Although the model used in this system for the prediction task could have been more complex, the model size is limited due to the 4 MB flash memory of the ESP32-WROOM-32. At the moment, more than 94% of the flash memory is used by the model and the rest of the sensor's firmware. Adding new layers with new weights to the model would exceed the current available space.

Table 4.3 Summary of the CNN Model

Layer (type)	**Output Shape**	**Parameter #**
conv1d_1	(None, 99, 128)	1664
max_pooling1d_1	(None, 49, 128)	0
dropout_1	(None, 49, 128)	0
conv1d_2	(None, 48, 128)	32896
max_pooling1d_2	(None, 24, 128)	0
dropout_2	(None, 24, 128)	0
conv1d_3	(None, 23, 128)	32896
max_pooling1d_3	(None, 11, 128)	0
dropout_3	(None, 11, 128)	0
flatten_1	(None, 1408)	0
dense_1	(None, 128)	180352
dropout_4	(None, 128)	0
dense_2	(None, 6)	774

4.3.4 Training and Testing Methodology

To test the activity recognition task using an embedded neural network, we defined six different activities to be classified by the model. Those activities are: walking, jumping, jogging, sitting, crouching and remaining stationary (meaning no movement of the legs but slight upper-body movements are allowed). The datasets were generated on one user with each dataset corresponding to only one of the six activities. The sensor was placed on the user's bust, as seen in Figure 4.1 using an elastic strap-band. A calibration of the IMU is performed upon powering the sensor and the user is then asked to do a specific activity. Once the activity is done, the system is reset, and a new activity is performed.

Table 4.4 Training and Validation Datasets Repartition

Activity	**Train dataset (minutes)**	**Test dataset (minutes)**
Stationary	20	20
Jumping	5	2
Walking	39	13
Jogging	20	10
Sitting	9	2
Crouching	4	2
Total	97	49

Training and testing datasets were generated separately to avoid introducing any bias in our system. The Leave-One-Trial-Out process described in the paper "Human activity recognition based on wearable sensor data: A standardization of the state-of-the-art" by Jordao et al. (Jordao et al. 2018)

was used, except the 10-fold cross validation process was not applied here because, as mentioned previously, the training and testing datasets were generated at different time.
Discussing the validation approach, the "Leave-one-trial-out" method was employed to rigorously assess the model's robustness across diverse trials. The primary goal was to determine if a model trained on a specific set of trials could accurately predict activities in trials it had not encountered before. To ensure the integrity and representativeness of the dataset, each trial was meticulously designed to encompass a wide range of activities and scenarios. However, in the realm of real-time systems, the critical concerns of data integrity and potential noise due to network-related challenges take center stage. Addressing these concerns, a meticulous pre-processing and data validation pipeline was implemented.

This comprehensive process included meticulous monitoring of time-stamps and sequence integrity, the meticulous handling of missing segments during preprocessing to maintain temporal coherence, and exhaustive testing in an environment with a stable network connection. In cases where data packets encountered transmission delays, a proactive approach involved systematic logging, thorough examination, and the exclusion of any packets deemed corrupted or inauthentic. This stringent data quality control process serves as a cornerstone to guarantee the reliability and precision of results within the context of real-time system applications.

Table 4.4 shows the distribution of the activities' datasets between training and testing. Finally, training was done on 21 epochs using a learning rate of 6×10^{-4} with the Adam optimizer and a batch size of 32.

For real-time testing with the micro-controller, the model was converted in a TensorFlow Lite format. Post-training quantization was used to improve the model size and hardware latency using full integer quantization of weights and activations by supplying a representative dataset of the sensor's data to the converter. The TensorFlow Lite model was then converted into a TensorFlow Lite FlatBuffer to be integrated into the micro-controller flash memory. During real-time testing, using the sensor, the activities performed by the user were analysed and matched with the real-time predictions of the convolutional neural network transmitted from the wearable sensor to the application using the MQTT protocol. The accuracy can be calculated with the number of time the user performed an activity and the number of time the sensor predicted the correct performed activity.

4.4 Results and Discussions

This section presents the outcomes of our research, specifically focusing on the results from the training and testing phases of our neural network-based human activity recognition system. Through these results, we aim to elucidate the model's efficacy under controlled and real-world conditions. A comprehensive discussion following the presentation of each result provides insights into the implications and potential avenues for future research.

4.4.1 ibNav 6.1 Battery performance analysis

In the assessment of the battery performance for ibNav 6.1, several crucial insights were gleaned from its 500mAh battery capacity. As depicted in Figure 4.3, the charging metrics of the device stood out as particularly noteworthy. The observed charging duration demonstrated both consistency and efficiency, suggesting that the ibNav 6.1 is adeptly designed for minimal downtime, thus enhancing its readiness for swift redeployment after depletion. Shifting the focus to the battery's discharging behavior, Figure 4.4 provides a detailed account across six varied tests. These discharge trends furnish a comprehensive understanding of the device's power consumption across an array of operational scenarios. Notably, the consistency in discharge rates across these tests emphasizes the device's stability and reliability. This steadiness is particularly valuable when anticipating the device's performance in prolonged real-time applications. At the same time, any deviations or anomalies in the discharge patterns could be instrumental in identifying specific operational conditions or tasks that may strain the battery, paving the way for potential optimizations in future iterations.

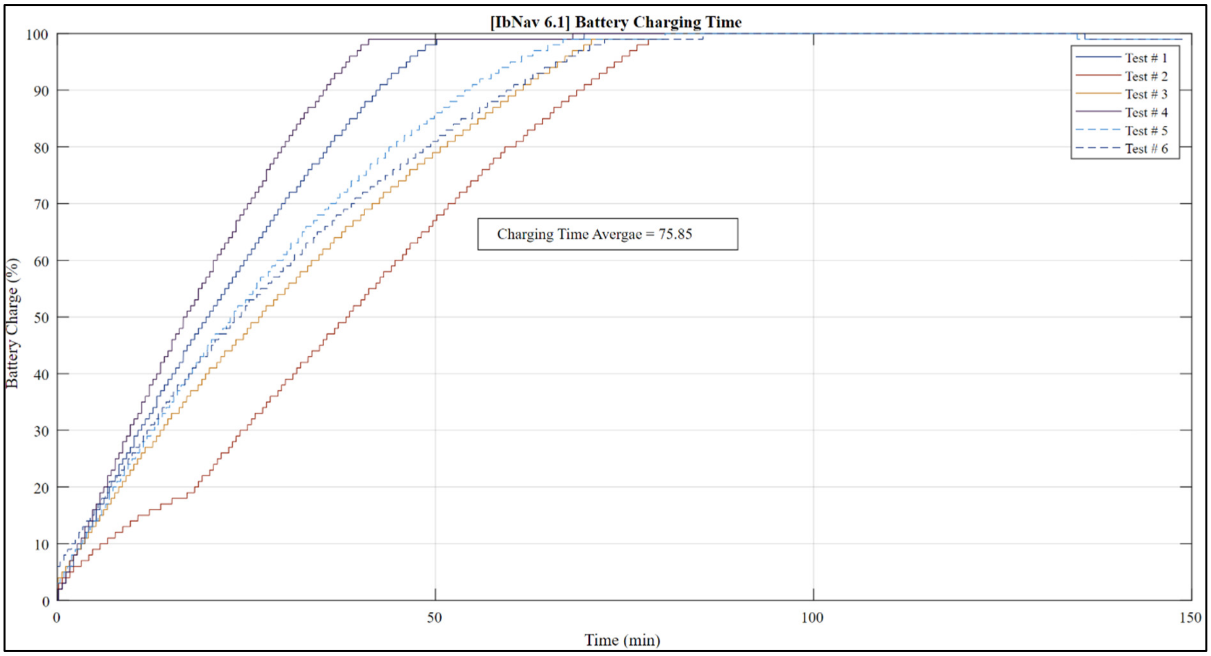


Figure 4.3 Battery charging time for six different tests

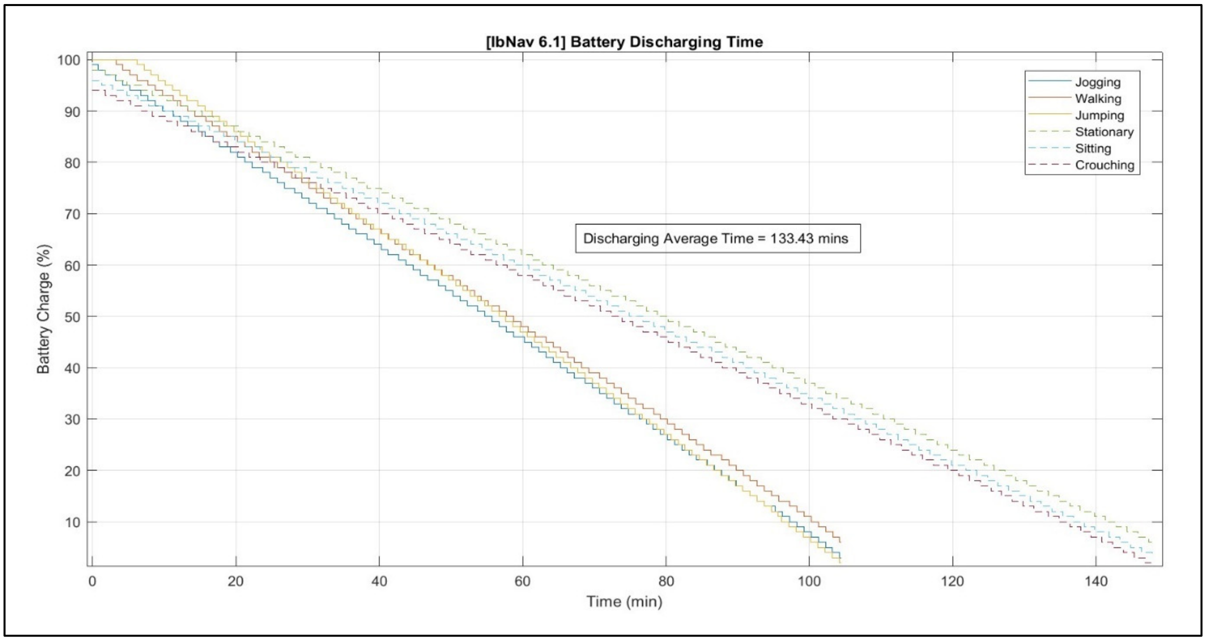


Figure 4.4 Battery Discharging time for six different tests

In essence, the battery performance results affirm that the ibNav 6.1 is not just robust in its capacity but also in its operational efficiency, making it a reliable contender for extended real-world deployment scenarios.

4.4.2 Training Results

The primary objective of this project was not to unearth the most advanced neural network for human activity recognition, but rather to develop a capability for immediate activity classification, embedded within a microcontroller, thereby eliminating the need for any external computing device. During the training and validation phases, the convolutional neural network illustrated in Figure 4.5 demonstrated an impressive maximum training accuracy of 99.97% based on a substantial training dataset consisting of 3,630,600 values, inclusive of a 10% overlap. In terms of validation, the table shows a peak accuracy of 99.47%, leveraging a validation dataset of 1,924,800 values, again with the equivalent overlap. Notably, analogous training results were acquired when employing 2D convolutional layers within the model.

Various other models were trialed; however, they were found to surpass the flash memory capacity of the microcontroller, hence proving impractical for our intended use-case. Figure 4.5 depicts a graphical representation of the training and validation accuracy.

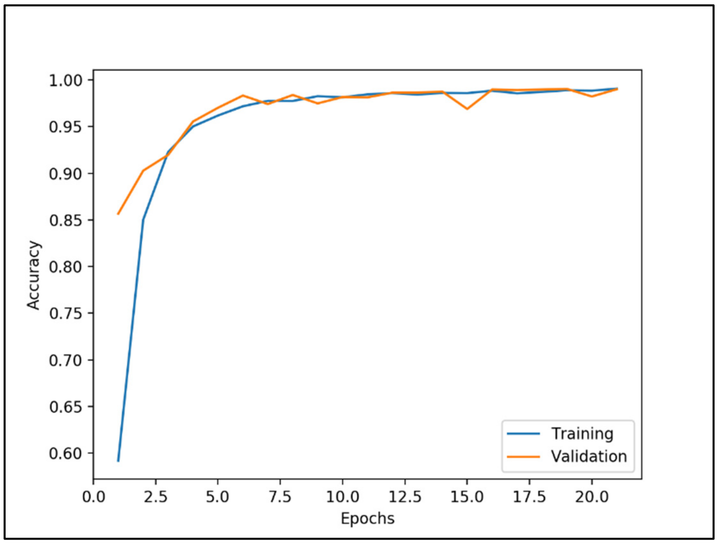


Figure 4.5 Training and validation accuracy graph

4.4.3 Testing Results

To evaluate the real-time activity classification capabilities of our system, we initiated a sequence of specific actions for a duration of 10 minutes, which was undertaken by a test subject. Subsequently, the predictions yielded by the system were relayed to the application, where a comprehensive analysis was performed to distinguish between correct and incorrect predictions.

As indicated in Table 4.5, both the sum of correct predictions and the corresponding accuracy rates for each activity, as well as overall, are tabulated. It is noteworthy that the embedded convolutional neural network (CNN) successfully executed the human activity recognition classification task with an appreciable degree of accuracy. Indeed, 90.4\% of the activities were accurately predicted by the sensor positioned on the test subject's torso. The misclassified activities are presumably attributable to the limited datasets utilized for training the neural network, which consequently compromised its ability to generalize unseen inertial data across a range of activities.

The enhancement of prediction accuracy is likely to be realized through the creation of larger activity datasets that encompass diverse environmental conditions and scenarios, coupled with an increased participant pool, and retraining the model accordingly. Additionally, our microcontroller's inference performance revealed an average processing time of 1.69 seconds for each inference, given 600 units of inertial data as input. This latency can be mitigated by fully leveraging the dual-core capabilities of the Xtensa microprocessor and optimizing the current model, for instance, by reducing the number of layers and weights in the neural network.

Table 4.5 Number of Accurately Predicted Activities During Testing

Activity	Number of predictions	Correct predictions	Accuracy
Stationary	226	222	98.2%
Jogging	220	214	97.3%
Walking	248	209	84.3%
Jumping	213	179	84%
Sitting	211	189	89.6%
Crouching	208	186	89.4%
Total	1326	1199	90.4%

The results obtained in this study demonstrate the successful implementation of a real-time sensor-embedded neural network for human activity recognition (HAR). The system achieved an overall accuracy of 90.4% in predicting various activities, showcasing its potential for practical real-world applications.

The numbers provided in Table 4.5 represent aggregated counts of specific activities captured across multiple sequences, each lasting 10 minutes, as mentioned in Section 4.3. To elaborate further, each 10-minute sequence was designed to encompass a mixture of activities. For example, a single sequence could involve a pattern such as 5 instances of jumping followed by 10 of crouching, then 4 of jogging, and so on. By aggregating these specific actions from multiple such 10-minute sequences, we arrived at the cumulative counts presented in Table 4.4. Therefore, the count of 226 for stationary activities represents the total number of stationary instances recorded across all these sequences, and similarly for other activities. However, there are several important points to consider and potential areas for improvement that should be discussed. However, there are several important points to consider and potential areas for improvement that should be discussed.

Firstly, the high training accuracy of 99.97% indicates that the neural network was able to learn and fit the training data effectively. This suggests that the model has sufficient capacity and can capture intricate patterns in the training dataset. However, the validation accuracy of 99.47% was slightly lower than the training accuracy, indicating some degree of overfitting. The model might be too complex for the available data, leading to limited generalization to unseen data. To address this, collecting a larger and more diverse dataset that encompasses various environmental conditions and participant demographics could improve the model's generalization performance.

Moreover, it is important to acknowledge that the accuracy of the system varied across different activities. While activities like "stationary," "sitting," and "crouching" achieved high accuracy, activities such as "walking" and "jumping" had lower accuracies. This discrepancy might be attributed to inherent challenges associated with distinguishing subtle differences in motion patterns between certain activities. Therefore, fine-tuning the model or exploring alternative neural network architectures specifically tailored for each activity category could potentially boost accuracy for individual activities.

Furthermore, the limited processing capabilities of the ESP32 microcontroller, with only 4 MB of flash memory, pose restrictions on the complexity of the neural network model that can be used. As a result, the model size had to be optimized, which could have affected the overall accuracy. Incorporating external flash memory or exploring specialized hardware accelerators designed for neural networks might enhance both model size and inference speed.

Real-time inference using microcontrollers inherently introduces latency due to the limited computing power compared to dedicated processing units. The average processing time of 1.69 seconds for each inference indicates a certain level of delay between the actual activity and its prediction. Exploring methods for reducing latency, such as optimizing the network architecture or leveraging hardware acceleration, is crucial to enhance the real-time capabilities of the system.

Another important consideration is the reliability of the system in various real-world scenarios. The system's performance might be influenced by external factors such as variations in sensor placement, different clothing types, or environmental conditions. Conducting extensive tests in diverse settings and with a larger pool of participants would provide valuable insights into the system's robustness and practical applicability.

In conclusion, the real-time sensor-embedded neural network for human activity recognition presented in this study represents a significant step towards wearable technologies that can perform HAR without relying on external processing devices. The system's accuracy and inference performance indicate its potential for numerous applications, including athlete performance analysis, soldier behavior detection, and underwater activities monitoring. However, challenges related to overfitting, activity-specific accuracy, model size, and latency need to be addressed for further improvements. By addressing these issues and conducting more extensive testing, the system could become a valuable tool in a wide range of real-world scenarios, benefiting fields such as healthcare, sports, and security.

4.5 Conclusions

This study presents a wearable sensor embedded with a convolutional neural network, demonstrating considerable proficiency in real-time human activity recognition tasks.

Currently, low-cost microcontrollers do not possess the performance capabilities of dedicated computing devices in terms of the complexity of the models that can be utilized. For instance, most real-time vision tasks necessitate intensive processing capabilities beyond the reach of non-specialized, economically efficient microcontrollers.

Nevertheless, the anticipated development of specialized hardware, such as the accelerator module by Coral, is projected to dramatically enhance the efficiency of wearable sensors in implementing neural networks. Future plans involve the design of an advanced board, informed by potential system improvements highlighted in the concluding section of this study. The next stage of development will also focus on creating larger datasets, incorporating multiple participants across diverse environments and encompassing a wider array of activities. The goal is to improve the performance, versatility, and robustness of the system under investigation. Moreover, it is intended to test reinforcement learning methods to enable online training capability.

Lastly, efforts will be made to establish a more robust and precise testing methodology for real-time classification analysis. The proposed improvements are expected to set a solid foundation for advancements in real-time human activity recognition, thus making significant contributions to the broader fields of wearable technologies and artificial intelligence.

A few modifications to the system used in this article could improve accuracy and performance. Currently, the ESP32 micro-controller used in this project is limited to 4 MB of flash memory for stocking both the FlatBuffer array containing the neural network and its weights, and the firmware of the sensor. Developing new boards with external flash memory to increase the total size capacity would allow to store larger neural network which could improve robustness and outperform the accuracy of the current one. Furthermore, two IMUs with different sensitivity could be used on the same board to improve activity recognition. Nowadays, dedicated computing chips are becoming more available and could be added to the sensor to improve inference capabilities and performance. Reinforcement learning methods could also be used to enable online training of the model to increase accuracy over time.

Finally, a few case studies could be done to select the best part of the body to place the sensor, for example the waist or the hip rather than on the bust.

CONCLUSION

This comprehensive analysis brings together three distinct research studies that offer innovative solutions to diverse challenges in the fields of indoor navigation, magnetometer calibration, and real-time human activity recognition. While each study operates within its unique domain, they collectively represent the evolving landscape of technology and its profound impact on our daily lives. In the realm of indoor navigation, the first study introduces a pioneering blockchain-based system tailored to GNSS-denied environments.

These settings, often characterized by their lack of access to global navigation satellite systems, have presented significant obstacles to achieving accurate and secure indoor navigation. To tackle this complex issue, the study proposes an integrated approach that combines a foot-mounted dual-inertial measurement unit (IMU) setup with a zero-velocity update (ZUPT) algorithm. This combination is augmented by the use of an extended Kalman filter (EKF) to fuse data from the IMUs and the ZUPT method, effectively rectifying errors induced by sensor drift during zero-velocity periods. A standout feature of this proposed system is its utilization of blockchain technology to secure data storage and validation.

By leveraging an immutable and distributed database, the system ensures the security and reliability of sensor data and navigation solutions. Experimental results from tests conducted at École de technologie supérieure (ÉTS) university attest to the system's impressive performance in indoor environments, with a mean root mean square error (RMSE) of 1.2 meters. Notably, this system's versatility extends its applicability across various domains, including driverless cars, robotics, and human monitoring. Furthermore, the scalability of this blockchain-based system is a key attribute. Its seamless implementation in environments with WiFi connectivity, facilitated by MQTT and gRPC protocols compatible with WiFi technology, ensures widespread deployment possibilities.

While the paper mentions that the ibnav sensor can sustain approximately 90 minutes of continuous usage, it's worth recognizing that specific use cases and sensor settings may influence this duration. To gain a deeper understanding of the sensor's operational limits and potential battery life enhancements, further research and experimentation are advisable. Additionally, the integration of LiDAR technology holds promise for future improvements in system performance.

The inclusion of LiDAR has the potential to not only augment the system's capabilities but also enhance the precision and reliability of outcomes. Further research should explore rigorous testing in diverse weather conditions to ensure the system's robustness, particularly in different temperature settings. Additionally, the incorporation of augmented reality (AR) can significantly enhance the user experience by overlaying essential information onto the real-world perspective, thereby improving not only navigation but also overall usability. Transitioning to the second study, the focus shifts to the calibration of 3-axis low-cost magnetometers.

The research presents a calibration method designed to minimize noise and address the impact of magnetic interference and instrumentation errors, ultimately enhancing the accuracy and dependability of magnetometer measurements. This calibration process involves data collection while moving the sensor in arbitrary directions, enabling the estimation of calibration parameters. These parameters encompass magnetic disturbances (soft iron and hard iron effects) as well as instrumental errors (scale factor, nonorthogonality, and bias). The collected data are modeled as a combination of these errors, and the calibration parameters are derived using the least square ellipsoid fitting algorithm. The study's findings affirm the efficacy of the calibration method, as calibrated magnetometer data exhibits a significantly improved fit to the surface of an ellipsoid compared to the original uncalibrated data. This outcome underscores the method's effectiveness in reducing errors and enhancing measurement accuracy. Furthermore, the calibration method's simplicity and cost-effectiveness are noteworthy advantages, as it relies only on magnetometer data and basic computer processing. The real-time capability of the algorithm is a pivotal asset, making it suitable for applications necessitating continuous calibration. Integrating this calibration method into the Intelligent IMU Sensor (IIS) further amplifies its practicality and suitability for real-world scenarios. Importantly, the principles and techniques presented in this research can be extended to other 3-axis magnetometers, expanding the applicability of the calibration method. Looking forward, future research endeavors may explore the integration of advanced optimization techniques and machine learning algorithms.

These innovations have the potential to further refine calibration accuracy and robustness, benefiting a wide array of applications such as motion capture, navigation systems, robotics, and geophysics. Finally, the third study ventures into the domain of wearable sensors and their role in real-time human activity recognition. It acknowledges the current limitations of low-cost microcontrollers in executing complex neural network models required for real-time vision tasks. However, the study anticipates advancements in specialized hardware, such as the Coral accelerator module, which is expected to significantly bolster the computational capabilities of wearable sensors for neural network applications.

The study's future plans encompass the development of an advanced board informed by potential system improvements outlined in the research. Expanding datasets to include multiple participants in diverse environments and activities is a core objective, with the aim of enhancing the system's performance, versatility, and robustness. An intriguing prospect highlighted in the conclusion is the exploration of reinforcement learning methods for enabling online training of the neural network model. This approach holds the promise of continuous accuracy improvement over time, adapting to evolving user behavior and activities. Furthermore, efforts will be directed toward establishing a more robust and precise testing methodology for real-time classification analysis, further enhancing the system's performance and reliability. A notable area for potential improvement is the limitation posed by the ESP32 microcontroller's finite 4 MB of flash memory. Expanding this capacity through the development of boards with external flash memory could significantly enhance the system's capabilities. Additionally, the incorporation of multiple IMUs with varying sensitivity levels could further enhance activity recognition accuracy. As dedicated computing chips become more accessible, their integration into the sensor may unlock enhanced inference capabilities and overall system performance. The conclusion also hints at the possibility of conducting case studies to determine the optimal placement of the sensor on the body, exploring alternatives such as the waist or hip instead of the bust.

In sum, these three research studies collectively represent the forefront of technological innovation, addressing diverse challenges in indoor navigation, magnetometer calibration, and real-time human activity recognition. While each study operates within a distinct domain, they share a common thread of pushing the boundaries of what technology can achieve. This synthesis of ideas underscores the interdisciplinary nature of technology and the potential for cross-pollination of insights. Moreover, these studies underscore the importance of constant innovation and the pursuit of solutions to complex problems. As technology continues to evolve, research endeavors such as these serve as crucial stepping stones toward more precise, secure, and efficient systems that can benefit society as a whole. In conclusion, these findings represent significant contributions to their respective fields, offering valuable guidance for future research and development efforts in the ibNav project each enhancing a specific part of this project. By embracing innovation and collaboration, we can continue to advance our understanding of technology's potential and unlock new horizons in a rapidly changing world.

LIST OF REFERENCES

Abosekeen, Ashraf, Aboelmagd Noureldin, Tashfeen Karamat, and Michael J. Korenberg. 2017. "Comparative Analysis of Magnetic-Based RISS Using Different MEMS-Based Sensors." Pp. 2944–59 in. Portland, Oregon.

Aggarwal, P. 2010. *MEMS-Based Integrated Navigation*. London, UK: Artech House.

Andel, Ján, Vojtech Šimák, Alžbeta Kanálikova, and Rastislav Pirník. 2022. "GNSS Based Low-Cost Magnetometer Calibration." *Sensors* 22(21):8447. doi: 10.3390/s22218447.

Anguita, Davide, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge L. Reyes-Ortiz. 2012. "Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine." Pp. 216–23 in *Ambient Assisted Living and Home Care*, edited by J. Bravo, R. Hervás, and M. Rodríguez. Berlin, Heidelberg: Springer Berlin Heidelberg.

Anon. n.d. "Introduction—Hyperledger-Fabricdocs Main Documentation." Retrieved (https://hyperledger-fabric.readthedocs.io/en/release-2.5/whatis.html).

Arora, A., and S. K. Yadav. 2019. "Batman: Blockchain-Based Aircraft Transmission Mobile Ad Hoc Network." Pp. 233–40 in *Proceedings of 2nd International Conference on Communication, Computing and Networking: ICCCN 2018, NITTTR Chandigarh, India*.

Blasch, E., R. Xu, Y. Chen, G. Chen, and D. Shen. 2019. "Blockchain Methods for Trusted Avionics Systems." Pp. 192–99 in *2019 IEEE National Aerospace and Electronics Conference (NAECON)*.

Brigante, Carmen M. N., Nunzio Abbate, Adriano Basile, Alessandro Carmelo Faulisi, and Salvatore Sessa. 2011. "Towards Miniaturization of a MEMS-Based Wearable Motion Capture System." *IEEE Transactions on Industrial Electronics* 58(8):3234–41. doi: 10.1109/TIE.2011.2148671.

Cao, Guocan, Xiang Xu, and Dacheng Xu. 2020. "Real-Time Calibration of Magnetometers Using the RLS/ML Algorithm." *Sensors* 20(2):535. doi: 10.3390/s20020535.

Casanova-Marqués, R., J. Torres-Sospedra, J. Hajny, and M. Gould. 2023. "Maximizing Privacy and Security of Collaborative Indoor Positioning Using Zero-Knowledge Proofs." *Internet Things* 22:100801. doi: 10.1016/j.iot.2023.100801.

Chen, C. L., J. Yang, W. J. Tsaur, W. Weng, C. M. Wu, and X. Wei. 2022. "Enterprise Data Sharing with Privacy-Preserved Based on Hyperledger Fabric Blockchain in IIOT's Application." *Sensors* 22:1146. doi: 10.3390/s22031146.

Chen, Jing, Xinyu Huang, Hao Jiang, and Xiren Miao. 2021. "Low-Cost and Device-Free Human Activity Recognition Based on Hierarchical Learning Model." *Sensors* 21(7):2359. doi: 10.3390/s21072359.

Chen, Xiyuan, Xiaotian Zhang, Min Zhu, Caiping Lv, Yuan Xu, and Hang Guo. 2022. "A Novel Calibration Method for Tri-Axial Magnetometers Based on an Expanded Error Model and a Two-Step Total Least Square Algorithm." *Mobile Networks and Applications* 27(2):794–805. doi: 10.1007/s11036-021-01898-z.

Choi, Ahnryul, Hyunwoo Jung, and Joung Hwan Mun. 2019. "Single Inertial Sensor-Based Neural Networks to Estimate COM-COP Inclination Angle During Walking." *Sensors (Basel, Switzerland)* 19(13):2974. doi: 10.3390/s19132974.

Coulin, Jade, Richard Guillemard, Vincent Gay-Bellile, Cyril Joly, and Arnaud De La Fortelle. 2022. "Online Magnetometer Calibration in Indoor Environments for Magnetic Field-Based SLAM." Pp. 1–8 in *2022 IEEE 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Beijing, China: IEEE.

Dharani, J., K. Sundarakantham, and K. Singh. 2022. "A Privacy-Preserving Framework for Endorsement Process in Hyperledger Fabric." *Comput. Secur.* 116:102637. doi: 10.1016/j.cose.2022.102637.

Gaba, P., R. S. Raw, M. A. Mohammed, J. Nedoma, and R. Martinek. 2022. "Impact of Block Data Components on the Performance of Blockchain-Based VANET Implemented on Hyperledger Fabric." *IEEE Access* 10:71003–18. doi: 10.1109/ACCESS.2022.3188296.

Ghasemzade, S. V., and F. Jamshidi. 2018. "Applications of Inertial Navigation Systems in Medical Engineering." *Journal of Biomedical Physics and Engineering* 8(3):325–32.

Helmi, Ahmed M., Mohammed A. A. Al-qaness, Abdelghani Dahou, and Mohamed Abd Elaziz. 2023. "Human Activity Recognition Using Marine Predators Algorithm with Deep Learning." *Future Generation Computer Systems* 142:340–50. doi: https://doi.org/10.1016/j.future.2023.01.006.

Hong, Nhung Tran Thi, Giang L. Nguyen, Nguyen Quang Huy, Do Viet Manh, Duc-Nghia Tran, and Duc-Tan Tran. 2023. "A Low-Cost Real-Time IoT Human Activity Recognition System Based on Wearable Sensor and the Supervised Learning Algorithms." *Measurement* 218:113231. doi: https://doi.org/10.1016/j.measurement.2023.113231.

Hu, XiaoWen, GuoQiang Wang, ZongHao Pan, Chinese Academy of Sciences Key Laboratory of Geospace Environment, School of Earth and Space Sciences, University of Science and Technology of China, Hefei 230026, China, and Institute of Space Science and Applied Technology, Harbin Institute of Technology, Shenzhen 518055, China. 2022. "Automatic Calculation of the Magnetometer Zero Offset Using the Interplanetary

Magnetic Field Based on the Wang–Pan Method." *Earth and Planetary Physics* 6(1):52–60. doi: 10.26464/epp2022017.

Ignatov, Andrey. 2018. "Real-Time Human Activity Recognition from Accelerometer Data Using Convolutional Neural Networks." *Applied Soft Computing* 62:915–22. doi: https://doi.org/10.1016/j.asoc.2017.09.027.

Jao, C. S., and A. M. Shkel. 2021. "ZUPT-Aided INS Bypassing Stance Phase Detection by Using Foot-Instability-Based Adaptive Covariance." *IEEE Sens. J.* 21:24338–48. doi: 10.1109/JSEN.2021.3112140.

Jindal, Sonika, Monika Sachdeva, and Alok KS Kushwaha. 2022. "Performance Evaluation of Machine Learning Based Voting Classifier System for Human Activity Recognition: 10.48129/Kjs. Splml. 19189." *Kuwait Journal of Science*.

Jordao, Artur, Antonio C. Nazare Jr, Jessica Sena, and William Robson Schwartz. 2018. "Human Activity Recognition Based on Wearable Sensor Data: A Standardization of the State-of-the-Art."

Khan, Pritam, Yogesh Kumar, and Sudhir Kumar. 2022. "CapsLSTM-Based Human Activity Recognition for Smart Healthcare With Scarce Labeled Data." *IEEE Transactions on Computational Social Systems* 1–10. doi: 10.1109/TCSS.2022.3223343.

Kwapisz, Jennifer R., Gary M. Weiss, and Samuel A. Moore. 2011. "Activity Recognition Using Cell Phone Accelerometers." *SIGKDD Explor. Newsl.* 12(2):74–82. doi: 10.1145/1964897.1964918.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. 2015. "Deep Learning." *Nature* 521(7553):436–44. doi: 10.1038/nature14539.

Li, Supeng, Defu Cheng, Yi Wang, and Jing Zhao. 2023. "Calibration of Strapdown Magnetic Vector Measurement Systems Based on a Plane Compression Method." *Measurement Science and Technology* 34(5):055115. doi: 10.1088/1361-6501/acbab0.

Lin, H. Y., and J. R. Zhan. 2023. "GNSS-Denied UAV Indoor Navigation with UWB Incorporated Visual Inertial Odometry." *Measurement* 206:112256. doi: 10.1016/j.measurement.2022.112256.

Liu, Hui, Tingting Xue, and Tanja Schultz. 2023. "On a Real Real-Time Wearable Human Activity Recognition System." Pp. 16–18 in *Proceedings of the 16th International Joint Conference on Biomedical Engineering Systems and Technologies, Lisbon, Portugal*.

Liu, Jiawei, Xiaohu Li, Shanshan Huang, Rui Chao, Zhidong Cao, Shu Wang, Aiguo Wang, and Li Liu. 2023. "A Review of Wearable Sensors Based Fall-Related Recognition Systems." *Engineering Applications of Artificial Intelligence* 121:105993. doi: https://doi.org/10.1016/j.engappai.2023.105993.

Lourenço, João, Leonardo Martins, Rui Almeida, Claudia Quaresma, and Pedro Vieira. 2016. “Low Cost Inertial Measurement Unit for Motion Capture in Biomedical Applications.” Pp. 151–58 in *Technological Innovation for Cyber-Physical Systems*, edited by L. M. Camarinha-Matos, A. J. Falcão, N. Vafaei, and S. Najdi. Cham: Springer International Publishing.

Mascret, Q., M. Bielmann, C. Fall, L. J. Bouyer, and B. Gosselin. 2018. “Real-Time Human Physical Activity Recognition with Low Latency Prediction Feedback Using Raw IMU Data.” Pp. 239–42 in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*.

Minarno, Agus Eko, Wahyu Andhyka Kusuma, and Rizalwan Ardi Ramandita. 2022. “Classification of Activity on the Human Activity Recognition Dataset Using Logistic Regression.” in *AIP Conference Proceedings*. Vol. 2453. AIP Publishing.

Moya Rueda, Fernando, René Grzeszick, Gernot Fink, Sascha Feldhorst, and Michael ten Hompel. 2018. “Convolutional Neural Networks for Human Activity Recognition Using Body-Worn Sensors.” *Informatics* 5(2):26. doi: 10.3390/informatics5020026.

Naheem, K., and M. S. Kim. 2022. “A Low-Cost Foot-Placed UWB and IMU Fusion-Based Indoor Pedestrian Tracking System for IoT Applications.” *Sensors* 22:8160. doi: 10.3390/s22218160.

Nazari, Farhad, Darius Nahavandi, Navid Mohajer, and Abbas Khosravi. 2022. “Comparison of Deep Learning Techniques on Human Activity Recognition Using Ankle Inertial Signals.” Pp. 2251–56 in *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*.

Nilsson, J. O., I. Skog, and P. Handel. 2010. “Performance Characterisation of Foot-Mounted ZUPT-Aided INSs and Other Related Systems.” Pp. 1–7 in *2010 International Conference on Indoor Positioning and Indoor Navigation*.

Norrdine, A., Z. Kasmi, and J. Blankenbach. 2016. “Step Detection for ZUPT-Aided Inertial Pedestrian Navigation System Using Foot-Mounted Permanent Magnet.” *IEEE Sens. J.* 16:6766–73. doi: 10.1109/JSEN.2016.2585599.

Ouni, Mohamed Ayoub, and Rene Landry. 2016. “Partide Swarm Optimization Algorithm in Calibration of MEMS-Based Low-Cost Magnetometer.” Pp. 27–33 in *2016 IEEE/ION Position, Location and Navigation Symposium (PLANS)*. Savannah, GA: IEEE.

Peregrina-Pérez, M. J., J. Lagares-Galán, and J. Boubeta-Puig. 2023. “Hyperledger Fabric Blockchain Platform.” *Distributed Computing to Blockchain* 283–95. doi: 10.1016/B978-0-323-96146-2.00014-0.

Pešić, S., M. Radovanović, M. Ivanović, M. Tošić, O. Iković, and D. Bošković. 2019. “Hyperledger Fabric Blockchain as a Service for the IoT: Proof of Concept.” Pp. 172–

83 in *Model and Data Engineering, Proceedings of the 9th International Conference, MEDI 2019, Toulouse, France, 28–31 October 2019*.

Phukan, Nabasmita, Shailesh Mohine, Achinta Mondal, M. Sabarimalai Manikandan, and Ram Bilas Pachori. 2022. "Convolutional Neural Network-Based Human Activity Recognition for Edge Fitness and Context-Aware Health Monitoring Devices." *IEEE Sensors Journal* 22(22):21816–26. doi: 10.1109/JSEN.2022.3206916.

Priyadarshini, Ishaani, Rohit Sharma, Dhowmya Bhatt, and M. Al-Numay. 2023. "Human Activity Recognition in Cyber-Physical Systems Using Optimized Machine Learning Techniques." *Cluster Computing* 26(4):2199–2215.

Pylvänäinen, Timo. 2008. "Automatic and Adaptive Calibration of 3D Field Sensors." *Applied Mathematical Modelling* 32(4):575–87. doi: 10.1016/j.apm.2007.02.004.

Ravi, D., S. Ramachandran, R. Vignesh, V. R. Falmari, and M. Brindha. 2022. "Privacy Preserving Transparent Supply Chain Management through Hyperledger Fabric." *Blockchain Res. Appl.* 3:100072. doi: 10.1016/j.bcra.2022.100072.

Renaudin, Valérie, Muhammad Haris Afzal, and Gérard Lachapelle. 2010. "Complete Triaxis Magnetometer Calibration in the Magnetic Domain." *Journal of Sensors* 2010:1–10. doi: 10.1155/2010/967245.

Ronao, Charissa Ann, and Sung-Bae Cho. 2016. "Human Activity Recognition with Smartphone Sensors Using Deep Learning Neural Networks." *Expert Systems with Applications* 59:235–44. doi: https://doi.org/10.1016/j.eswa.2016.04.032.

Ru, Xu, Nian Gu, Hang Shang, and Heng Zhang. 2022. "MEMS Inertial Sensor Calibration Technology: Current Status and Future Trends." *Micromachines* 13(6):879. doi: 10.3390/mi13060879.

Salimi, S., P. T. Morón, J. P. Queralta, and T. Westerlund. 2022. "Secure Heterogeneous Multi-Robot Collaboration and Docking with Hyperledger Fabric Blockchain." *IEEE Access* 10:1–7. doi: 10.1109/WF-IoT54382.2022.10152244.

Sangenis, E., C. S. Jao, and A. M. Shkel. 2022. "SVM-Based Motion Classification Using Foot-Mounted IMU for ZUPT-Aided INS." P. 4 in *2022 IEEE Sensors*.

Sartayeva, Y., and H. C. Chan. 2023. "A Survey on Indoor Positioning Security and Privacy." *Comput. Secur.* 131:103293. doi: 10.1016/j.cose.2023.103293.

Silva do Monte Lima, João Paulo, Hideaki Uchiyama, and Rin-Ichiro Taniguchi. 2019. "End-to-End Learning Framework for IMU-Based 6-DOF Odometry." *Sensors (Basel, Switzerland)* 19(17):3777. doi: 10.3390/s19173777.

Simon Colomar, D., J. O. Nilsson, and P. Handel. 2012. "Smoothing for ZUPT-Aided INSs." Pp. 1–5 in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*.

Styp-Rekowski, Kevin, Ingo Michaelis, Claudia Stolle, Julien Baerenzung, Monika Korte, and Odej Kao. 2022. "Machine Learning-Based Calibration of the GOCE Satellite Platform Magnetometers." *Earth, Planets and Space* 74(1):138. doi: 10.1186/s40623-022-01695-2.

Subasi, Abdulhamit, Asalah Fllatah, Kholoud Alzobidi, Tayeb Brahimi, and Akila Sarirete. 2019. "Smartphone-Based Human Activity Recognition Using Bagging and Boosting." *Procedia Computer Science* 163:54–61. doi: https://doi.org/10.1016/j.procs.2019.12.086.

Tao, X., F. Zhu, X. Hu, W. Liu, and X. Zhang. 2022. "An Enhanced Foot-Mounted PDR Method with Adaptive ZUPT and Multi-Sensors Fusion for Seamless Pedestrian Navigation." *GPS Solut.* 26:13. doi: 10.1007/s10291-021-01196-x.

Teslabs, Engineering. n.d. "A Way to Calibrate a Magnetometer." Retrieved (https://teslabs.com/articles/magnetometer-calibration/).

Vidya, B., and Sasikumar P. 2022. "Wearable Multi-Sensor Data Fusion Approach for Human Activity Recognition Using Machine Learning Algorithms." *Sensors and Actuators A: Physical* 341:113557. doi: https://doi.org/10.1016/j.sna.2022.113557.

Wagner, J. F., M. Kohl, and B. Gyorfi. 2022. "Reevaluation of Algorithmic Basics for ZUPT-Based Pedestrian Navigation." *IEEE Access* 10:118419–37. doi: 10.1109/ACCESS.2022.3220629.

Wahdan, Ahmed, Jacques Georgy, Walid F. Abdelfatah, and Aboelmagd Noureldin. 2014. "Magnetometer Calibration for Portable Navigation Devices in Vehicles Using a Fast and Autonomous Technique." *IEEE Transactions on Intelligent Transportation Systems* 15(5):2347–52. doi: 10.1109/TITS.2014.2313764.

Wan, Shaohua, Lianyong Qi, Xiaolong Xu, Chao Tong, and Zonghua Gu. 2020. "Deep Learning Models for Real-Time Human Activity Recognition with Smartphones." *Mobile Networks and Applications* 25(2):743–55. doi: 10.1007/s11036-019-01445-x.

Wang, Y., S. Askari, and A. M. Shkel. 2019. "Study on Mounting Position of IMU for Better Accuracy of ZUPT-Aided Pedestrian Inertial Navigation." *IEEE Trans. Aerosp. Electron. Syst.* 56:2280–91. doi: 10.1109/TAES.2019.2946506.

Wang, Y., A. Chernyshoff, and A. M. Shkel. 2018. "Error Analysis of ZUPT-Aided Pedestrian Inertial Navigation." Pp. 206–12 in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*.

Wang, Y., C. S. Jao, and A. M. Shkel. 2021. “Scenario-Dependent ZUPT-Aided Pedestrian Inertial Navigation with Sensor Fusion.” *Gyroscopy Navig.* 12:1–16. doi: 10.1134/S2075108721010119.

Wang, Yusheng, Andrei Chernyshoff, and Andrei M. Shkel. 2020. “Study on Estimation Errors in ZUPT-Aided Pedestrian Inertial Navigation Due to IMU Noises.” *IEEE Transactions on Aerospace and Electronic Systems* 56(3):2280–91. doi: 10.1109/TAES.2019.2946506.

Wang, Z., H. Zhao, S. Qiu, and Q. Gao. 2015. “Stance-Phase Detection for ZUPT-Aided Foot-Mounted Pedestrian Navigation System.” *IEEE Trans. Mechatron.* 20:3170–81. doi: 10.1109/TMECH.2015.2430357.

Xu, X., X. Wang, Z. Li, H. Yu, G. Sun, S. Maharjan, and Y. Zhang. 2021. “Mitigating Conflicting Transactions in Hyperledger Fabric-Permissioned Blockchain for Delay-Sensitive IoT Applications.” *IEEE Internet Things J.* 8:10596–607.

Yan, Hang, Qi Shan, and Yasutaka Furukawa. 2017. “RIDI: Robust IMU Double Integration.” *CoRR* abs/1712.09004.

Yan, Rui, Fubin Zhang, and Huihui Chen. 2019. “A MEMS-Based Magnetometer Calibration Approach in AUV Navigation System.” Pp. 1–6 in *OCEANS 2019 - Marseille*. Marseille, France: IEEE.

Yang, H., P. Vijayakumar, J. Shen, and B. B. Gupta. 2022. “A Location-Based Privacy-Preserving Oblivious Sharing Scheme for Indoor Navigation.” *Future Gener. Comput. Syst.* 137:42–52. doi: 10.1016/j.future.2022.06.016.

Yang, M., R. Zhu, Z. Xiao, and B. Yan. 2021. “Symmetrical-Net: Adaptive Zero Velocity Detection for ZUPT-Aided Pedestrian Navigation System.” *IEEE Sens. J.* 22:5075–85. doi: 10.1109/JSEN.2021.3094301.

Yang, Y., X. Ren, X. Jia, and B. Sun. 2023. “Development Trends of the National Secure PNT System Based on BDS.” *Sci. China Earth Sci.* 66:929–38. doi: 10.1007/s11430-022-1069-7.

Zemam, S., and H. Benzerrouk. 2010. “Improvement and Evaluation of Zero-Velocity Detectors for Foot-Mounted Inertial Navigation System- PART I.” P. 12 in *2010 International Conference on Indoor Positioning and Indoor Navigation*.

Zeng, M., L. T. Nguyen, B. Yu, O. J. Mengshoel, J. Zhu, P. Wu, and J. Zhang. 2014. “Convolutional Neural Networks for Human Activity Recognition Using Mobile Sensors.” Pp. 197–205 in *6th International Conference on Mobile Computing, Applications and Services*.

Zhao, T., and M. J. Ahamed. 2021. “Pseudo-Zero Velocity Re-Detection Double Threshold Zero-Velocity Update (ZUPT) for Inertial Sensor-Based Pedestrian Navigation.” *IEEE Sens. J.* 21:13772–85. doi: 10.1109/JSEN.2021.3070144.

Zhao, Z. 2022. “Comparison of Hyperledger Fabric and Ethereum Blockchain.” Pp. 584–87 in *2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*.