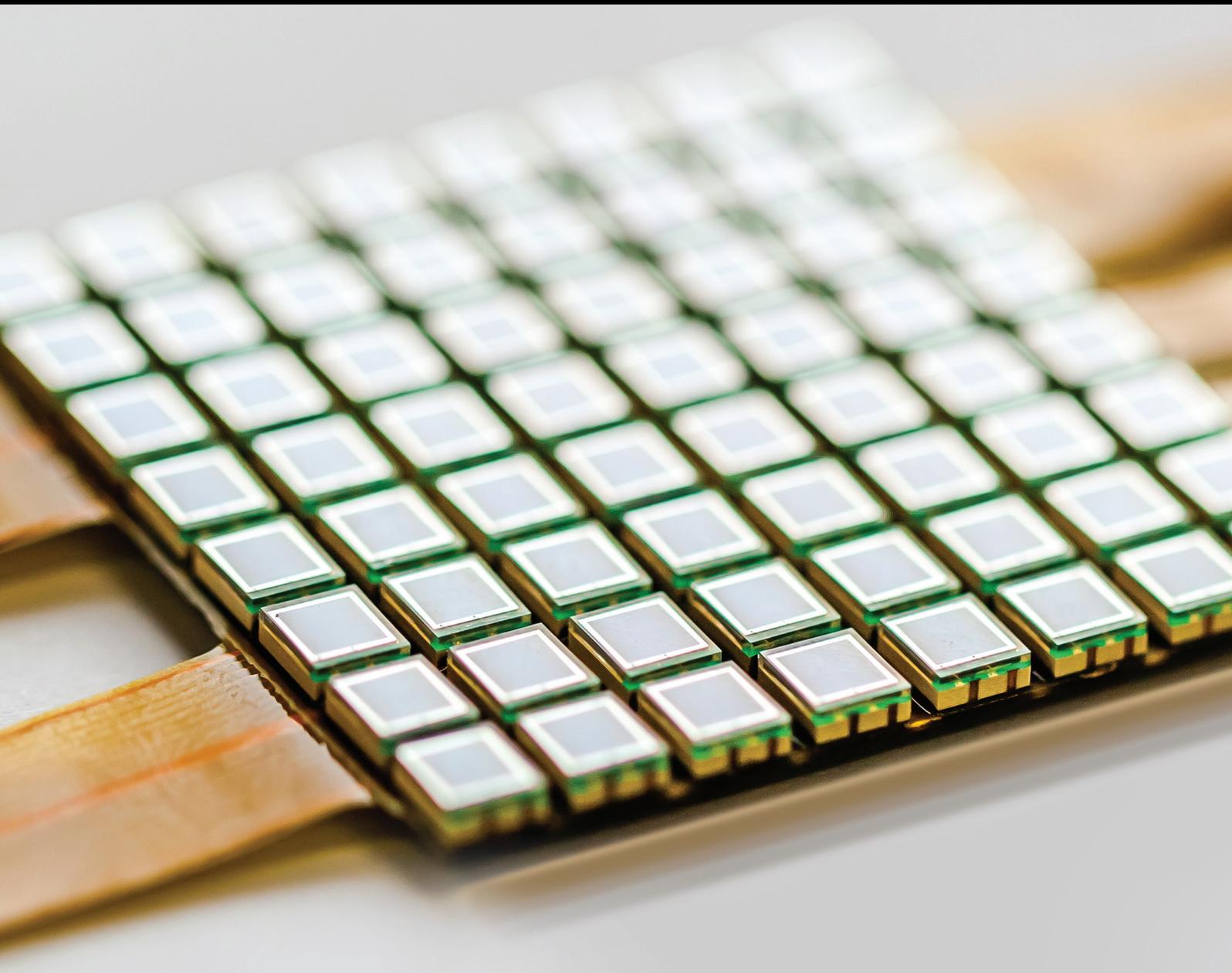


# Indoor Tracking, Mapping, and Navigation: Algorithms, Technologies, and Applications

Lead Guest Editor: Jacky C. K. Chow

Guest Editors: Michael Peter, Marco Scaioni, and Mohannad Al-Durgham





---

# **Indoor Tracking, Mapping, and Navigation: Algorithms, Technologies, and Applications**

**Indoor Tracking, Mapping, and Navigation:  
Algorithms, Technologies, and Applications**

Lead Guest Editor: Jacky C. K. Chow

Guest Editors: Michael Peter, Marco Scaioni,  
and Mohannad Al-Durgham



---

Copyright © 2018 Hindawi. All rights reserved.

This is a special issue published in "Journal of Sensors." All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## Editorial Board

Harith Ahmad, Malaysia  
M. İlhan Akbaş, USA  
Manuel Aleixandre, Spain  
Bruno Andò, Italy  
Fernando Benito-Lopez, Spain  
Romeo Bernini, Italy  
Shekhar Bhansali, USA  
Wojtek J. Bock, Canada  
Paolo Bruschi, Italy  
Belén Calvo, Spain  
Stefania Campopiano, Italy  
Domenico Caputo, Italy  
Sara Casciati, Italy  
Gabriele Cazzulani, Italy  
Chi Chiu Chan, Singapore  
Nicola Cioffi, Italy  
Marco Consales, Italy  
Jesus Corres, Spain  
Andrea Cusano, Italy  
Antonello Cutolo, Italy  
Dzung Dao, Australia  
Manel del Valle, Spain  
Francesco Dell’Olio, Italy  
Franz L. Dickert, Austria  
Giovanni Diraco, Italy  
Nicola Donato, Italy  
Mauro Epifani, Italy  
Abdelhamid Errachid, France  
Stephane Evoy, Canada  
Vittorio Ferrari, Italy  
Luca Francioso, Italy  
Manel Gasulla, Spain  
Carmine Granata, Italy

Alfredo Guemes, Spain  
Banshi D. Gupta, India  
Clemens Heitzinger, Austria  
María del Carmen Horrillo, Spain  
Syed K. Islam, USA  
Stephen James, UK  
Hai-Feng Ji, USA  
Sang Sub Kim, Republic of Korea  
Antonio Lazaro, Spain  
Laura M. Lechuga, Spain  
Chengkuo Lee, Singapore  
Chenzong Li, USA  
Eduard Llobet, Spain  
Jaime Lloret, Spain  
Yu-Lung Lo, Taiwan  
Oleg Lupan, Moldova  
Frederick Mailly, France  
Santiago Marco, Spain  
Vincenzo Marletta, Italy  
Eugenio Martinelli, Italy  
Antonio Martínez Olmos, Spain  
Jose R. Martínez-De-Dios, Spain  
Giuseppe Maruccio, Italy  
Yasuko Y. Maruo, Japan  
Mike McShane, USA  
Fanli Meng, China  
Stephen. J. Mihailov, Canada  
Heinz C. Neitzert, Italy  
Calogero M. Oddo, Italy  
Marimuthu Palaniswami, Australia  
Alberto J. Palma, Spain  
Lucio Pancheri, Italy  
Roberto Paolesse, Italy

Alain Pauly, France  
Giorgio Pennazza, Italy  
Michele Penza, Italy  
Salvatore Pirozzi, Italy  
Antonina Pirrotta, Italy  
Stavros Pissadakis, Greece  
Biswajeet Pradhan, Malaysia  
Armando Ricciardi, Italy  
Christos Riziotis, Greece  
Maria Luz Rodriguez-Mendez, Spain  
Jerome Rossignol, France  
Carlos Ruiz, Spain  
Josep Samitier, Spain  
Giorgio Sberveglieri, Italy  
Andreas Schütze, Germany  
Sandra Sendra, Spain  
Woosuck Shin, Japan  
Pietro Siciliano, Italy  
Vincenzo Spagnolo, Italy  
Sachin K. Srivastava, India  
Stefano Stassi, Italy  
Vincenzo Stornelli, Italy  
Weilian Su, USA  
Tong Sun, UK  
Raymond Swartz, USA  
Hidekuni Takao, Japan  
Guiyun Tian, UK  
Suna Timur, Turkey  
H. Vaisocherova - Lisalova, Czech Republic  
Xavier Vilanova, Spain  
Luca Vollero, Italy  
Qihao Weng, USA  
Hai Xiao, USA

# Contents

## **Indoor Tracking, Mapping, and Navigation: Algorithms, Technologies, and Applications**

Jacky C. K. Chow , Michael Peter , Marco Scaioni , and Mohannad Al-Durgham   
Editorial (3 pages), Article ID 5971752, Volume 2018 (2018)

## **An Improved Particle Filter Algorithm for Geomagnetic Indoor Positioning**

He Huang , Wei Li , De An Luo, Dong Wei Qiu , and Yang Gao  
Research Article (9 pages), Article ID 5989678, Volume 2018 (2018)

## **Practical In Situ Implementation of a Multicamera Multisystem Calibration**

Ivan Detchev , Ayman Habib, Mehdi Mazaheri, and Derek Lichti  
Research Article (12 pages), Article ID 5351863, Volume 2018 (2018)

## **An Analysis of Multiple Criteria and Setups for Bluetooth Smartphone-Based Indoor Localization Mechanism**

Manuel Castillo-Cara, Jesús Lovón-Melgarejo, Gusseppe Bravo-Rocca, Luis Orozco-Barbosa, and Ismael García-Varea  
Research Article (22 pages), Article ID 1928578, Volume 2017 (2018)

## **Linear Kalman Filter for Attitude Estimation from Angular Rate and a Single Vector Measurement**

Shangqiu Shan, Zhongxi Hou, and Jin Wu  
Research Article (9 pages), Article ID 9560108, Volume 2017 (2018)

## **Distributed Monocular SLAM for Indoor Map Building**

Ruwan Egodagamage and Mihran Tuceryan  
Research Article (11 pages), Article ID 6842173, Volume 2017 (2018)

## **Integrated SINS/WSN Positioning System for Indoor Mobile Target Using Novel Asynchronous Data Fusion Method**

Hai Yang, Wei Li, Chengming Luo, Jinyao Zhang, and Zhuoyin Si  
Research Article (11 pages), Article ID 7879198, Volume 2017 (2018)

## **Sequential Monte Carlo Localization Methods in Mobile Wireless Sensor Networks: A Review**

Ammar M. A. Abu Znaid, Mohd. Yamani Idna Idris, Ainuddin Wahid Abdul Wahab, Liana Khamis Qabajeh, and Omar Adil Mahdi  
Review Article (19 pages), Article ID 1430145, Volume 2017 (2018)

## Editorial

# Indoor Tracking, Mapping, and Navigation: Algorithms, Technologies, and Applications

Jacky C. K. Chow <sup>1</sup> Michael Peter <sup>2</sup> Marco Scaioni <sup>3</sup> and Mohannad Al-Durgham <sup>4</sup>

<sup>1</sup>Aston Business School, Aston University, Birmingham B4 7ET, UK

<sup>2</sup>Department of Earth Observation Science, Faculty ITC, University of Twente, 7514 AE Enschede, Netherlands

<sup>3</sup>Department of Architecture, Built Environment and Construction Engineering, Politecnico Milano, 20133 Milano, Italy

<sup>4</sup>Geomni, Jacksonville, FL 32225, USA

Correspondence should be addressed to Jacky C. K. Chow; [jckchow@ucalgary.ca](mailto:jckchow@ucalgary.ca)

Received 18 December 2017; Accepted 24 December 2017; Published 17 April 2018

Copyright © 2018 Jacky C. K. Chow et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The necessity for an accurate, precise, and quick positioning capability is growing in modern societies. This feature is needed on the one side by actively moving objects like people, vehicles, and robots. On the other side, passively moving objects may also need to be continuously located and tracked, such as merchandise and transported goods, kids, and people with disabilities. Some examples include (1) finding the nearest exit during emergency events by taking into consideration the obstacles on the ground, (2) continuous georeferencing both in outdoor and indoor environments to receive location-based information, (3) tracking people and goods for surveillance purposes, and (4) monitoring of people's physical activities to encourage an active lifestyle or to facilitate medical rehabilitation.

Notwithstanding the big progresses of outdoor positioning technologies developed in the last decades, which have already influenced our lifestyle, new solutions are continuously in demand. Navigation systems should be able to operate ubiquitously in challenging conditions, including the possibility of indoor tracking. This function requires the integration of multiple positioning systems to facilitate seamless and robust navigation. In indoor environments, the support from *Global Navigation Satellite Systems* (GNSSs, see [1]) cannot be exploited and alternative solutions have to be explored. For instance, great attention is focused on low-cost technologies that may be widely used in a sustainable way. Concurrently, the exploitation of those devices that are already commonly carried by people (such as smartphones)

gives the opportunity to develop solutions prone to be quickly adopted by a large number of users.

On the other side, the development of technology for indoor positioning should be complemented by the accurate knowledge of the environment's geometry and semantics, which are both needed to make intelligent decisions and plan optimal actions. Spatial 3D information is crucial in understanding the scene and context to make technologies truly autonomous. In the case of modern newly designed buildings, the use of existing Building Information Models (BIMs) may provide extensive data sources for indoor geometry to support real-time positioning and navigation [2, 3]. Unfortunately, in most existing buildings, the planimetric information is not available in digital form. Thus, indoor maps and 3D models should be collected on purpose. Even though technologies for indoor mapping [4] have reached a high degree of maturity, due to factors such as scene obstructions, movement variations, sensor limitations, and model uncertainty, indoor mapping remains to be a highly challenging problem. Popular techniques for indoor 3D surveying based on integrated photogrammetric [5] and computer vision [6] methods, as well as static [7] and mobile [8] 3D scanning systems, may quickly provide dense and accurate point clouds. On the other hand, discrete point clouds might not be directly suitable to support indoor navigation, but they require a complex modelling stage to obtain 3D vector models, whose derivation is not yet a fully automated process [9]. While public and important buildings with high human

traffic may be worthwhile to organize specific projects for 3D indoor modeling, the large majority of indoor spaces require other solutions, such as crowdsourced mapping [10]. Last but not least, standards for sharing 3D indoor digital mapping data have to consolidate and be widely used [11]. Beyond the necessity of such information for indoor positioning and navigation, the integration of indoor 3D models georeferenced in the geodetic coordinate systems and topologically connected to the outdoor maps is one of the major challenges to be resolved in order to support the emerging technology implemented in smart cities. It is also noteworthy to mention the establishment of a benchmarking on indoor modelling by the *International Society of Photogrammetry and Remote Sensing* (ISPRS); see more details in [12].

In this special issue, seven papers that illustrate advances in indoor tracking, mapping, and navigation are published. Three papers deal with different aspects related to the localization of nodes in *wireless sensor networks* (WSNs, [13]), two papers discuss the state-of-the-art methods for navigation estimation, and two papers are focused on mapping using optical instruments.

In “Sequential Monte Carlo Localization Methods in Mobile Wireless Sensor Networks: A Review,” the authors first propose a new classification of localization schemes in mobile WSNs, where the chance of changing spatial position allows to expand coverage using a smaller number of sensors while reducing power consumption. In the three suggested categories, the range-free scheme exploits network connectivity to work out the position, without the need of an external device such as an antenna. The paper specifically provides a survey of the *sequential Monte Carlo* method to estimate the location of nodes in range-free WSNs [14]. Three metrics are used for classifying these methods, that is, localization accuracy, computational cost, and communication cost.

In the paper “An Analysis of Multiple Criteria and Setups for Bluetooth Smartphone-Based Indoor Localization Mechanism,” the problem of fading impairments in localization of Bluetooth Low Energy (BLE) 4.0 beacons is addressed. BLE beacons have become a relevant alternative to Wi-Fi-based indoor localization mechanisms [15]. After reviewing the literature about this technology, the paper identifies the main system parameters to be considered during the design of this type of localization sensors: the mean localization error, local prediction accuracy, and global prediction accuracy. A series of experimental tests prove this conclusion.

The integration of strapdown inertial navigation system (SINS) and WSN is the topic of the paper “Integrated SINS/WSN Positioning System for Indoor Mobile Target Using Novel Asynchronous Data Fusion Method.” SINS sensors contained in a mobile target can continuously provide ubiquitous positioning without depending on any external interaction with other devices. On the other hand, while the positioning accuracy is very high in the short term, SINS sensors suffer from drift within time [16]. When GNSS signals are available, the integration of both technologies is used to complement the short-term stability of SINS by the long-term stability of GNSS, but this does not work in indoor environments. Some authors have already proposed the integration of SINS and WSN to compensate for the drift [17]

and to obtain continuous indoor positioning. In this paper, the authors focus on this method by developing an asynchronous data-fusion technique based on an unscented Kalman filter [18]. Some simulations and experiments with real data show that this method may perform better than traditional techniques for both asynchronous and synchronous data.

A topic that is closely related to the previous article is presented in “Linear Kalman Filter for Attitude Estimation from Angular Rate and a Single Vector Measurement.” Indeed, attitude estimation is fundamental in inertial indoor navigation [19]. In the past, several filtering methods have been used, mainly on the basis of different versions of the Kalman filter and complementary filter [20]. In particular, the latter has been used because the former is difficult to apply with relatively low-cost hardware. Here the authors propose the implementation of a Kalman-filtering scheme along with gyroscopic data and a single vector observation. Using quaternion algebra, the attitude is estimated together with its variance-covariance information. The Single Vector Observation Linear Kalman filter (SVO-LKF) features a flexible design that makes its application faster than other filters based on linearization, as also demonstrated in simulations and experiments reported in the paper.

Besides using a variation of the Kalman filter to estimate the navigation solution, for scenarios where the noise is not normally distributed and the mathematical model is nonlinear, a particle filter may be considered. In the article “An Improved Particle Filter Algorithm for Geomagnetic Indoor Positioning,” the authors presented an improved navigation solution using the ambient magnetic field processed by a particle filter. They proposed an improved particle filter algorithm based on initial positioning error constraint, inspired by the Hausdorff distance measurement point set matching theory. Since the operating range of the particle filter cannot exceed the magnitude of the initial positioning error, it avoids the adverse effect of sampling particles with the same magnetic intensity but away from the target during the iteration process on the positioning system.

As mentioned before, 3D mapping of indoor environments is one of the aspects strictly related to navigation and tracking. One solution to this problem is to use distributed mapping agents (e.g., people or robots) that move in and out of an indoor environment and concurrently may generate partial maps. In a second stage, independently and partially reconstructed maps need to be unified into a highly accurate global map. This topic is presented in the paper “Distributed Monocular SLAM for Indoor Map Building.” Here the *simultaneous localization and mapping* (SLAM) technique is adopted for the acquisition of relatively dense 3D models by using distributed mobile sensors [21]. On the other hand, constructed maps are used to help navigation in an interactive way (tracking and mapping problem). Here the use of monocular visual SLAM is considered for distributed indoor mapping. If an agent moves into an area that has already been mapped, it may use the existing geometric information to support navigation. In the case that an agent enters an area that has not yet been mapped, it starts the local mapping process and localizes itself as a part of the SLAM process. All independently and locally built maps are then

merged together to complement each other. In particular, the paper introduces and discusses a distributed framework to compute map overlaps also in the case no prior knowledge of the starting positions of the agents is available.

The problem of mapping the indoor environment is also the final aim of the solution proposed in “Practical In Situ Implementation of a Multicamera Multisystem Calibration.” Systems made up of multiple consumer-grade cameras may offer a low-cost and effective solution for image-based indoor mobile mapping. In this paper, first, some methods for the geometric calibration of multicamera systems are reviewed, including the determination of both interior and exterior orientation parameters for each sensor [22]. Then, the authors introduce a novel, versatile methodology for multicamera system geometric calibration, capable of concurrently handling more camera systems as well.

## Acknowledgments

We would like to thank all the authors for their valuable contributions and the reviewers for their help. It is everybody’s contributions that allowed us to complete this special issue with high-quality papers. Finally, we would like to thank all members of the editorial board for approving this special issue.

Jacky C. K. Chow  
Michael Peter  
Marco Scaioni  
Mohannad Al-Durgham

## References

- [1] B. Hofmann-Wellenhof, H. Lichtenegger, and E. Wasle, *GNSS — Global Navigation Satellite Systems*, Springer Verlag, Vienna, 2008.
- [2] C. Eastman, P. Teicholz, R. Sacks, and K. Liston, *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Architects, Engineers, Contractors, and Fabricators*, John Wiley & Sons Inc, Hoboken, NJ, USA, 3rd edition, 2018.
- [3] L. Barazzetti, “Parametric as-built model generation of complex shapes from point clouds,” *Advanced Engineering Informatics*, vol. 30, no. 3, pp. 298–311, 2016.
- [4] J. Kárník and J. Streit, “Summary of available indoor location techniques,” *IFAC-PapersOnLine*, vol. 49, no. 25, pp. 311–317, 2016.
- [5] G. Forlani, R. Roncella, and C. Nardinocchi, “Where is photogrammetry heading to? State of the art and trends,” *Rendiconti Lincei*, vol. 26, Supplement 1, pp. 85–S96, 2015.
- [6] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, UK, 2006.
- [7] G. Vosselman and H.-G. Maas, *Airborne and Terrestrial Laser Scanning*, vol. 318, Taylor and Francis Group, Boca Raton, FL, USA, 2010.
- [8] C. Thomson, G. Apostolopoulos, D. Backes, and J. Böhm, “Mobile laser scanning for indoor modelling,” in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II-5/W2, pp. 289–293, Antalya, Turkey, 2013.
- [9] L. Previtali, R. B. Barazzetti, and M. Scaioni, “Towards automatic indoor reconstruction of cluttered building rooms from point clouds,” in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II-5, pp. 281–288, Riva del Garda, Italy, 2014.
- [10] C. Heipke, “Crowdsourcing geospatial data,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 65, no. 6, pp. 550–557, 2010.
- [11] G. Gröger, T. Kolbe, C. Nagel, and K. Häfele, *OpenGIS City Geography Markup Language (CityGML) Encoding Standard v2.0.0*, Open Geospatial Consortium Standard, Open Geospatial Consortium, OGC, 2008.
- [12] K. Khoshelham, L. Díaz Vilariño, M. Peter, Z. Kang, and D. Acharya, “The ISPRS benchmark on indoor modelling,” in *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-2/W7, pp. 367–372, Wuhan, China, 2017.
- [13] J. Yick, B. Mukherjee, and D. Ghosal, “Wireless sensor network survey,” *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [14] L. Mihaylova, D. Angelova, and A. Zvikhachevskaya, “Sequential Monte Carlo methods for localization in wireless networks,” in *Advances in Intelligent Signal Processing and Data Mining*, p. 354, Springer-Verlag, Berlin Heidelberg, 2013.
- [15] Z. Farid, R. Nordin, and M. Ismail, “Recent advances in wireless indoor localization techniques and system,” *Journal of Computer Networks and Communications*, vol. 2013, Article ID 185138, 12 pages, 2013.
- [16] Q. Yuan and I.-M. Chen, “3-D localization of human based on an inertial capture system,” *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 806–812, 2013.
- [17] C. Luo, W. Li, H. Yang, B. Ying, and G. Xin, “Implementation of mobile target positioning technology integrating SINS with WSN measurements,” *Journal of Sensors*, vol. 2014, Article ID 673179, 12 pages, 2014.
- [18] J. Wang, A. Hu, X. Li, and Y. Wang, “An improved PDR/magnetometer/floor map integration algorithm for ubiquitous positioning using the adaptive unscented Kalman filter,” *ISPRS International Journal of Geo-Information*, vol. 4, no. 4, pp. 2638–2659, 2015.
- [19] X. Yun, J. Calusdian, E. R. Bachmann, and R. B. McGhee, “Estimation of human foot motion during normal walking using inertial and magnetic sensor measurements,” *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 7, pp. 2059–2072, 2012.
- [20] W. T. Higgins, “A comparison of complementary and Kalman filtering,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-11, no. 3, pp. 321–325, 1975.
- [21] R. Smith, M. Self, and P. Cheeseman, “Estimating uncertain spatial relationships in robotics,” in *Autonomous Robot Vehicles*, I. J. Cox and G. T. Wilfong, Eds., pp. 167–193, Springer-Verlag, New York, N.Y., USA, 1990.
- [22] T. Luhmann, S. Robson, S. Kyle, and J. Böhm, *Close Range Photogrammetry: 3D Imaging Techniques*, Walter De Gruyter Inc., Germany, 2nd edition, 2014.

## Research Article

# An Improved Particle Filter Algorithm for Geomagnetic Indoor Positioning

He Huang <sup>1,2</sup> Wei Li <sup>1</sup> De An Luo,<sup>1,2</sup> Dong Wei Qiu <sup>1,2</sup> and Yang Gao<sup>3</sup>

<sup>1</sup>School of Geomatics and Urban Spatial Information, Beijing University of Civil Engineering and Architecture, Beijing 102616, China

<sup>2</sup>Beijing Advanced Innovation Center for Future Urban Design, Beijing 100044, China

<sup>3</sup>The University of Calgary, Department of Geomatics Engineering, Calgary, AB, Canada T2N 1N4

Correspondence should be addressed to Wei Li; 1159928759@qq.com and Dong Wei Qiu; qiudw@bucea.edu.cn

Received 1 April 2017; Revised 7 July 2017; Accepted 13 December 2017; Published 19 March 2018

Academic Editor: Mohannad Al-Durgham

Copyright © 2018 He Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Geomagnetic indoor positioning is an attractive indoor positioning technology due to its infrastructure-free feature. In the matching algorithm for geomagnetic indoor localization, the particle filter has been the most widely used. The algorithm however often suffers filtering divergence when there is continuous variation of the indoor magnetic distribution. The resampling step in the process of implementation would make the situation even worse, which directly lead to the loss of indoor positioning solution. Aiming at this problem, we have proposed an improved particle filter algorithm based on initial positioning error constraint, inspired by the Hausdorff distance measurement point set matching theory. Since the operating range of the particle filter cannot exceed the magnitude of the initial positioning error, it avoids the adverse effect of sampling particles with the same magnetic intensity but away from the target during the iteration process on the positioning system. The effectiveness and reliability of the improved algorithm are verified by experiments.

## 1. Introduction

With the rapid development of location-based services, indoor positioning is receiving increased attention. Indoor positioning technology can be applied to public security, location tracking, intelligent transportation, and so forth. For example, a hospital can locate and monitor patients using indoor positioning technology. Indoor positioning can solve the problem of finding cars in large complexes and underground parking lots; it can also quickly find an optimal shopping route in large shopping malls. Indoor positioning is currently based on technologies such as Wi-Fi, RFID, Bluetooth, UMB, and geomagnetism. Among different technologies for indoor positioning, the geomagnetic indoor positioning approach has broad application prospects because it is infrastructure-free and has geomagnetic signal stability [1].

As early as the 1950s, the geomagnetic positioning technology based on geomagnetic field vector matching was

applied to large-scale outdoor environment navigation, including those of surface ships, submarines, and missiles. In the past, the geomagnetic positioning techniques typically used the correlation matching algorithms [2–4] or the recursive filtering algorithms [5, 6]. More recent use of the geomagnetic positioning technology in the indoor environment is considered an important breakthrough. The geomagnetic field in a modern building is disturbed by the steel-reinforced concrete, steel structure, power systems, electronic equipment, and other artificial sources, which forms a unique and spatially continuous geomagnetic signature [7, 8]. The indoor magnetic field can be collected to construct a fingerprint model based on Wi-Fi fingerprint technology. In general, the construction of geomagnetic perturbation fingerprint model can use a variety of characteristic variables, including a single variable (total geomagnetic intensity), double variables ( $H$  and  $V$  direction magnetic field components), or three variables ( $X$ ,  $Y$ , and  $Z$  direction magnetic field components). The more characteristic variables are included

for the position determination, the more improvement can be made on the positioning accuracy [9]. The indoor positioning technology based on the magnetic field matches the magnetic field intensity, which is obtained by the indoor moving carrier, with the geomagnetic disturbance fingerprint model to enable indoor positioning.

At present, the particle filter algorithm,  $k$ -nearest neighbor [10], and extended Kalman filter (EKF) [11] have been applied in indoor positioning systems based on a fingerprint model. The particle filter developed by Monte Carlo techniques has been widely used as one effective matching algorithm, because of its advantages of broad adaptability in nonlinear and non-Gaussian systems [12–14]. Grand and Thrun [12] presented a real-time indoor localization method that utilizes a single 3-axis magnetometer to estimate the position of a handheld device. Using a particle filter, a localization accuracy of 0.7 meters in position and 25 degrees in orientation was achieved for a simple straight-line trajectory. Haverinen and Kemppainen [13] equipped a robot with a single magnetic sensor that measured three planes. They ran their robot through a corridor, had it collect data at a set of locations to create a map of the hallway, and used Monte Carlo localization (a particle filter) to accurately determine its location from any starting point. The maximum error was about 28 centimeters, but the robot needed to travel 25 meters in order to localize itself in general. Kim et al. [15] proposed an indoor positioning system using smartphones. They used a particle filter to estimate the users' location based on geomagnetic anomalies [15] and used the distance between the user and the wall as a conditional constraint. The experimental results in the corridor showed that the positioning accuracy is within 3 meters.

The use of particle filters requires the carrier to complete a continuous recursive filter location over a period of time, and it suffers the problem of filtering divergence [16]. The filter convergence however is affected by the distribution of indoor geomagnetic, and a divergence would occur if it lacks indoor geomagnetism signatures. The sparser the characteristics of the indoor geomagnetic field, as many locations have similar magnetic field patterns, the more likely the divergence of the particle filter algorithm occurs, which eventually leads to excessive positioning error.

In response to this challenge, this paper proposes the use of point set matching in the Hausdorff distance measurement method to improve the particle filter algorithm as well as the method of positioning error constraints. In the new particle filter algorithm, the initial positioning error is used as the distance constraint, only allowing particles that meet the constraints to iterate, and resampling. Coupled with the convergence characteristics of the resampling step, we can suppress the effect of persistent divergence. The main contributions of this work include (1) the development of a geomagnetic data acquisition platform, (2) an improved particle filter algorithm to prevent filter divergence, and (3) the design of an offline test system to estimate the location of mobile robots carrying magnetic sensors.

The paper is organized as follows. The classical particle filter algorithm is described in Section 2.1. Section 2.2 briefly

outlines the improved algorithm based on the Hausdorff distance. In Section 3, the data acquisition robot is first briefly described, then the data collection is presented, and finally the analysis of the positioning results is provided. The conclusion is given in Section 4.

## 2. An Improved Particle Filtering Algorithm

**2.1. Classical Particle Filter Algorithm.** The particle filter algorithm is an optimal Bayesian estimation method based on Monte Carlo's idea [17]. It is often used to estimate the position of a target moving along the corridor from an unknown location. The particle filter uses Monte Carlo localization (MCL) method to approximate the posterior probability distribution  $p(x_t|z_t)$  when it is too complex to directly sampled, but the prior probability density  $p(x_t|x_{t-1}^i)$  can be sampled and the measurement density  $p(z_t|x_t^i)$  can be evaluated, where  $x_t$  represents the target positioning,  $x_t^i$  represents the particle positioning, and  $z_t$  is the observation. The entire particle filtering process consists of three steps, namely, forecasting, updating, and resampling.

The particle filter algorithm follows the general framework of a sequential importance sampling (SIS) algorithm, and it adds the sampling importance resampling (SIR) to solve the sample impoverishment caused by the iterative process. It first generates a set of samples with  $N$  particles from the prior probability density and then generates the weight  $w^i$  according to the measurement density  $p(z_t|x_t^i)$  of each particle. The weights are normalized to make their sum equal to one before the resampling. The basic idea of resampling is to remove particles with small weight and concentrate on particles with large weight. Due to easy implementation, the particle filter widely uses such resampling algorithm. The two-dimensional motion model is given by

$$x_t^i = x_{t-1}^i + Hl, \quad (1)$$

where  $l \sim U(0, L)$ ,  $U(0, L)$  which obeys a uniform distribution;  $L$  is the moving distance; and

$$H = \begin{bmatrix} \sin \theta & 0 \\ 0 & \cos \theta \end{bmatrix}. \quad (2)$$

In (2),  $\theta$  is the moving direction. We can roughly choose the size of  $L$  according to the length of the person's step. The measurement density  $p(z_t|x_t^i)$  is based on the single variable Gaussian probability density function, which is given by

$$p(z_t|x_t^i) = \frac{1}{\sigma_r \sqrt{2\pi}} \exp \left[ -\frac{(z_t - f(x_t^i))^2}{2\sigma_r^2} \right], \quad (3)$$

where  $\sigma_r$  is the covariance of the observation  $z_t$ . The function  $f(x_t^i)$  returns the magnetic field intensity of the position  $x_t^i$  in

the fingerprint model. Finally, the position  $\hat{x}_t$  of the target is estimated based on the weight  $w$  of the sample particles, which is given by

$$\hat{x}_t = \sum_{i=1}^N w_t^i x_t^i, \quad (4)$$

and the positioning error is given by

$$\text{err} = \sqrt{(x_t - \hat{x}_t)^2}. \quad (5)$$

For geomagnetic indoor positioning based on the particle filter, the particle weight is determined by (3) according to the magnetic field intensity of the target and particle in which the closer the geomagnetic intensity, the greater the weight. After removing the particles with the small weights, the estimated position is then calculated by (4). When the difference in the weight of the particle is not large, the deviation of the distance between the particle position  $x_t^i$  and the target position is large [16]. If the particle position is far from the target position, the error between the estimated position and the target position is large. In other words, the estimated position is obtained by weighted sum of the individual particle positions. The greater the value of the Euclidean distance  $d_h$  between the target and particles with large weight, the greater the value of  $\hat{x}_t$ . Finally, it will cause greater positioning error as shown by (5). We have further explained it in Section 3.2 by an example.

## 2.2. An Improved Algorithm Based on Hausdorff Distance.

Increasing the characteristic elements of the geomagnetic matching can effectively solve the divergence problem in the particle filter algorithm, but it will also greatly increase the complexity of the original algorithm and the time required for positioning. Inspired by the Hausdorff distance matching, an improved particle filter algorithm based on the Hausdorff distance is proposed and described in the following.

We know that the magnetic field at any point is stable in the static environment and it is unique within a certain range. If the particle filter starts within a reasonable range, the deviation of estimated position will be reduced. The Hausdorff distance can then be applied to determine this reasonable range for the particle filter. The Hausdorff distance is the maximum value of the distance between the two sets of points in space which is defined as follows [4]:

$$d_H(A, B) = \max(d_h(A, B), d_h(B, A)), \quad (6)$$

where  $d_h$  represents the maximum distance between  $a_i$  and  $b_j$  in the two point sets of  $d_h(A, B) = \max_{a \in A} (\min_{b \in B} \|a - b\|)$  and  $d_h(B, A) = \min_{b \in B} (\max_{a \in A} \|b - a\|)$ ,  $\|\bullet\|$  is a distance norm,  $A = \{a_1, a_2, \dots, a_p\}$  and  $B = \{b_1, b_2, \dots, b_q\}$  represent

**Step 1.** Initialization:

$t = 0$ , draw particles  $x_0^i$  uniformly with  $w^i = 1/N$

**Step 2.** Calculate the positioning error:

$t = 1$ , the initial positioning error is treated as the  $d_H$

**Step 3.** Particle iteration:

For time steps  $t = 1, 2, \dots$

For particle numbers  $i = 1: N$

IF  $d_h < d_H$

The particle state is shifted:  $x_t^i \sim p(x_t | x_{t-1}^i)$

Get mobile vector online observations:  $z_t$

Weight update:  $w_t^i = p(z_t | x_t^i)$

END IF

END FOR

**Step 4.**

Make CDF of  $w_t^i$  and

For particle numbers  $i = 1: N$

Draw  $u_i = U(0, 1)$

Resample  $\tilde{x}_t^i = \text{CDF}(u_i)$

END FOR

**Step 5.** Output the weighted position:

$$\hat{x}_t = \sum_{i=1}^N w_t^i x_t^i.$$

ALGORITHM 1

two finite sets of points. The Hausdorff distance represents the degree of matching between  $A$  and  $B$ .

In general, in the geomagnetic feature matching based on classical particle filtering, we can control the weight of the particles that participate in calculating the estimated position in (4) when their weight becomes large. However,  $d_h$  is difficult to control, once its value becomes too large. This will cause filtering divergence. Since  $d_H$  is the maximum distance between each particle and the target, it can be controlled. As a result, if this maximum value can be controlled within a reasonable range, we can avoid the problem that  $d_h$  becomes large. In order to avoid the effect of human factors, we use the positioning system to estimate the initial position of the target. The constraint to be applied is the initial positioning error between the estimated initial position and the true position of the target in the measurement environment. Let the initial positioning error as  $d_H$ ; the following initial positioning error constraint can be established:

$$d_h < d_H. \quad (7)$$

The position  $\hat{x}_t$  of the target estimated based on the weight  $w$  of the sample particles in (4) must satisfy the condition in (7). Using the improved particle filter algorithm, the divergence problem can be avoided because we have removed those particles that had a larger weight but were far from the target. As a result, the positioning results of the positioning system can converge more quickly. The implementation steps of the improved algorithm is given in Algorithm 1 and the flowchart of the implementation process is given in Figure 1.

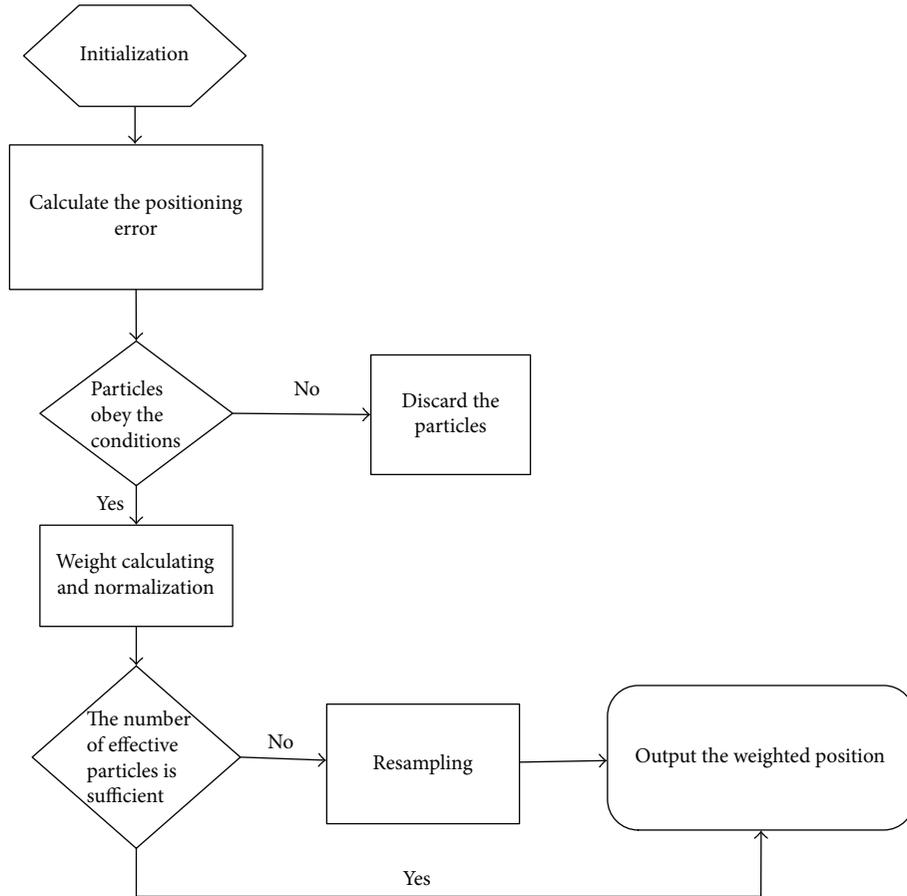


FIGURE 1: Flowchart of improved particle filter algorithm.

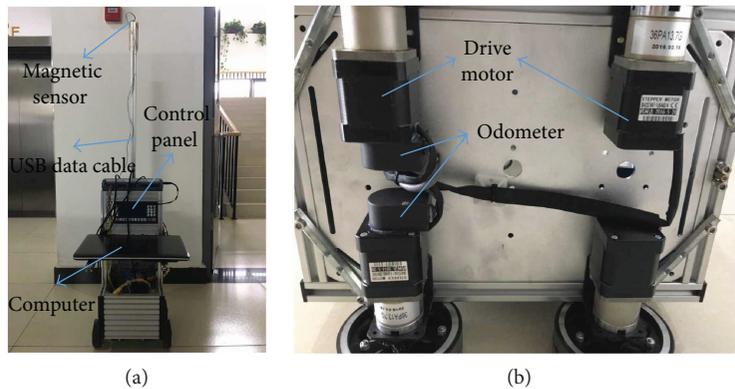


FIGURE 2: Geomagnetic data acquisition robot platform. (a) Front view. (b) Bottom view.

### 3. Experiments and Analysis

**3.1. Data Collection System Development: A Geomagnetic Data Acquisition Robot.** The construction of a geomagnetic fingerprint model is a prerequisite for geomagnetic indoor positioning technology. In order to quickly and effectively establish a fingerprint model, a data acquisition robot has been developed by Urban Surveying and Mapping Institute which includes data acquisition module and control module as shown in Figure 2 [18]. The robot is loaded with a HMC5983 three-axis magnetic sensor placed at a

height of 1.3 meters above the floor. The placement of sensor at such a height is to separate the magnetic sensor to avoid interference by the robot itself. The robot is controlled by the control module. Data through the USB cable real-time are imported to the computer, through openMAT software to save geomagnetic data and record, while the magnitude was calculated. Finally, we use the MATLAB-based software to conduct position determination in the computer. Since the magnetic sensor itself will produce a relatively stable interference field when energized [19], we must correct the HMC5983 magnetic sensor

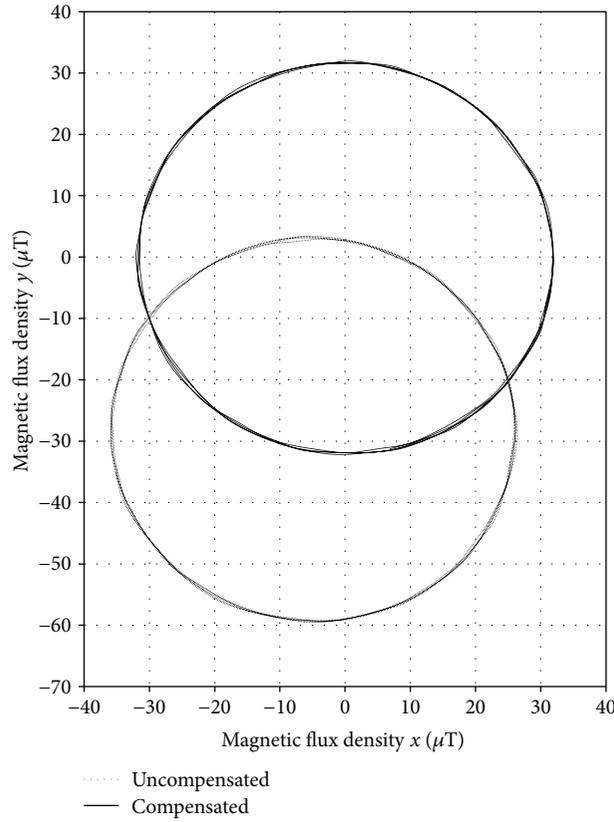


FIGURE 3: Comparison of magnetic field values after magnetic sensor compensation.

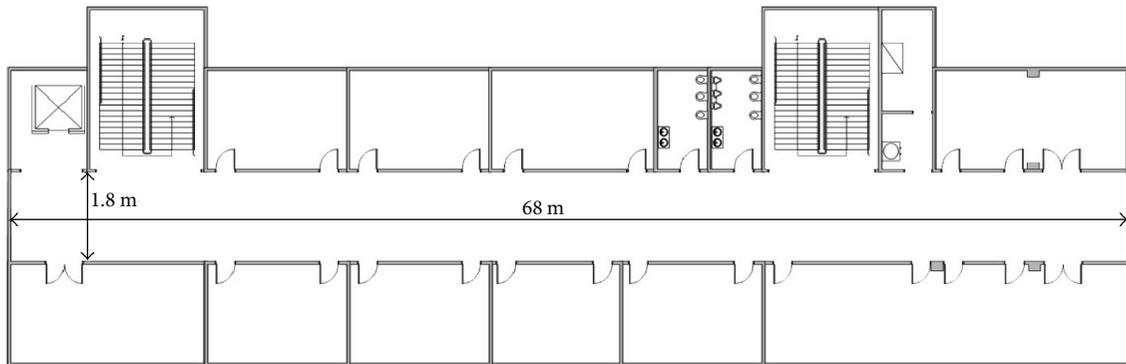


FIGURE 4: Floor map of the Institute of Surveying and Mapping.

in the data acquisition module to ensure the authenticity of the data before collecting data. The rotation of the magnetic sensor around the  $z$ -axis of its own coordinate system will generate the corresponding circle based on the magnitude of the  $x$ -axis and the  $y$ -axis magnetic field. However, due to the presence of an interference field, the location of the center of the circle will move a short distance. The interference can be eliminated by adding a fixed value to the output value of the magnetic sensor, as shown in Figure 3.

The method to correct the value  $(x_{sf}, y_{sf})$  of the HMC5983 magnetic sensor and the  $x$ -axis and  $y$ -axis magnetic field offset  $(x_{off}, y_{off})$  is given as follows:

$$\begin{aligned}
 x_{sf} &= \max\left(1, \left(\frac{y_{\max} - y_{\min}}{x_{\max} - x_{\min}}\right)\right), \\
 y_{sf} &= \max\left(1, \left(\frac{x_{\max} - x_{\min}}{y_{\max} - y_{\min}}\right)\right), \\
 x_{off} &= \left(\frac{x_{\max} - x_{\min}}{2 - x_{\max}}\right)x_{sf}, \\
 y_{off} &= \left(\frac{y_{\max} - y_{\min}}{2 - y_{\max}}\right)y_{sf}, \\
 x_v &= h_x x_{sf} + x_{off}, \\
 y_v &= h_y y_{sf} + y_{off},
 \end{aligned} \tag{8}$$

where  $(x_{\max}, x_{\min})$  and  $(y_{\max}, y_{\min})$  are the maximum and minimum values, respectively;  $(h_x, h_y)$  is the original magnetic field value on the  $x$ -axis and  $y$ -axis; and  $(x_v, y_v)$  is the compensation of the  $x$ -axis and  $y$ -axis direction of the magnetic field value.

**3.2. Experimental Data Collection and Fingerprint Model Construction.** The experimental data acquisition was conducted in the 2nd floor of the Institute of Surveying and Mapping at Beijing University of Civil Engineering and Architecture (BUCEA). The frames of the building are reinforced concrete. Figure 4 shows a simple plan of the corridor with a total length of 68 meters. The magnetic field was measured every 25 Hz producing a three-dimensional vector  $\mathbf{m} = [m_x, m_y, m_z]$ , in units of  $\mu\text{T}$ .

In order to prove the defect of particle filter algorithm, we randomly collected a row of data in the corridor. Figure 5 shows the magnetic field distribution in the corridor which is linear and continuous, and the magnetic field intensity of different locations is likely to be the same. This proves that when the particle position  $x'_i$  is far from the target, it will lead to biased estimate  $\hat{x}_i$  in the particle filter algorithm. An example is shown in Figure 6.

In the corridor, we collected geomagnetic data along four lines of 60 cm apart. For building the fingerprint model, we control the acquisition robot along the planned route to move forward which the step length of robot is 0.2 m with geomagnetic data measurement at a sampling rate of 5 Hz. To perform localization, we use the norm of the magnetic field  $\mathbf{m} = [m_x, m_y, m_z]$  as the observation, because magnetic field intensity  $\|\mathbf{m}\|$  is a rotation invariant scalar quantity. The  $\|\mathbf{m}\|$  is given by

$$\|\mathbf{m}\| = \sqrt{m_x^2 + m_y^2 + m_z^2}. \quad (9)$$

The final fingerprint model is created by applying a linear interpolation to the magnetic field intensity using a 0.02 m step size in the computer, as shown in Figure 7, and the map adds the grid coordinates as its relative coordinate frame after it is generated.

## 4. Positioning Results

The acquisition robot used to take measurements was controlled to travel straight along the corridor, with a step length of 0.6 m. It recorded data every second, iteration times  $t = 16$ . In the positioning system, the number of sample particles  $N$  is set to 400; each test set was conducted using the same value of the standard deviation of the measurement model  $\sigma_r = 5.0\mu\text{T}$ , and a total of 20 time tests. The SIS algorithm and the particle filter algorithm with the SIR algorithm are used to investigate the geomagnetic matching precision and the degree of continuous filtering divergence.

The burr in Figure 8 shows that the SIS algorithm has the problem of filtering divergence and geomagnetic matching instability in the geomagnetic matching process. Although the positioning error shows an overall decreasing trend, there is no convergence.

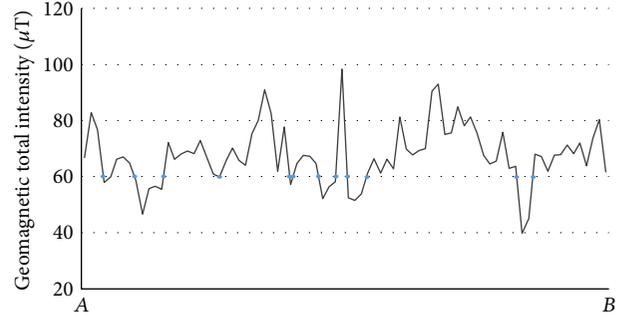


FIGURE 5: Distribution of the magnitude of the magnetic field in a row of the corridor. Blue spots indicate points with the same magnetic field intensity.

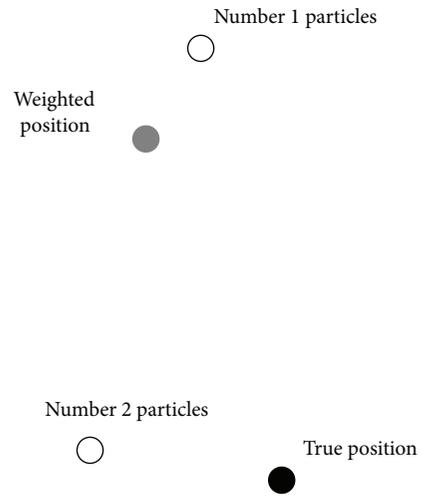


FIGURE 6: According to (4), we find that if the  $w$  of number 1 is the same as that of number 2, but the distance between number 1 and the true position is very far, will cause the estimated position away from the true position.

As shown in Figure 9, the addition of the SIR algorithm makes the filtering divergence more serious, and the continuous filtering divergence phenomenon results in the loss of positioning (see ①, ②, ③ in Figure 9). Most of the experiments began to converge after  $t = 5$ . The average matching accuracy was 1.78 m and the average run time was 10.9 s. We can see from the comparison of Figures 8 and 9 that after the SIR algorithm is added, the particle is convergent after iteration calculation for a period of time due to its resampling step. The matching accuracy is far superior to that of the SIS algorithm, although positioning loss is also inevitable.

The idea of the positioning error constraint method is shown in Figures 10 and 11. In Figure 10, the dots indicate the sample particles that are randomly selected in the positioning area; the hollow circle represents the initial position of the weighted position and the \* sign indicates the position of the moving carrier at time  $t = 1$ . In Figure 11, the circle which is deformed due to the limitation of the geomagnetic fingerprint model range represents the initial positioning

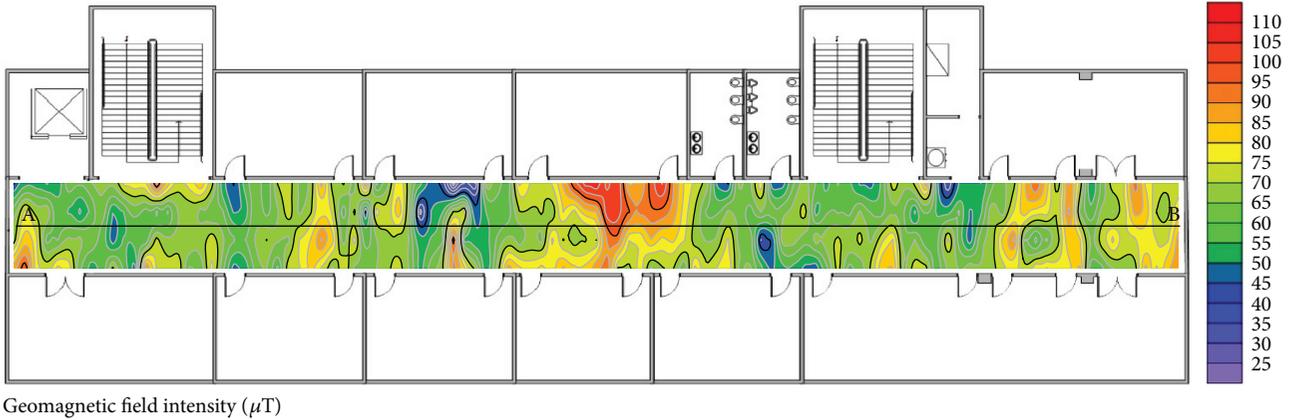


FIGURE 7: Geomagnetic fingerprint model. Straight-line AB is any one row of the corridors.

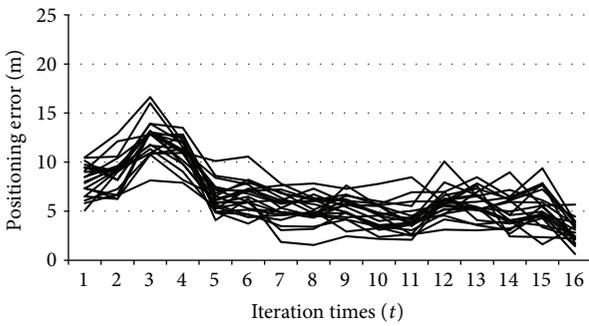


FIGURE 8: Positioning error based on SIS algorithm.

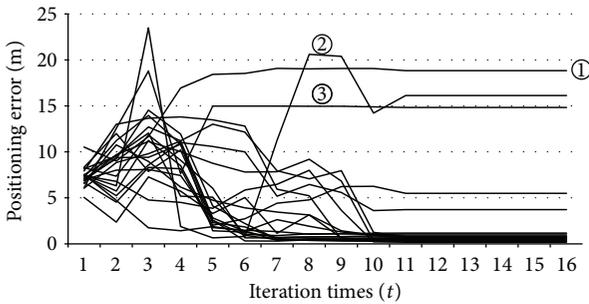


FIGURE 9: Positioning error based on particle filter algorithm with the SIR.

error range. It can be concluded from Figures 10 and 11 that with the establishment of the positioning error constraint after the initial positioning, the sampling particles have a restriction on their maximum distance  $d_H$ . It does not use the particles which are far from the true location of target in the positioning process. Due to the reduction in the number of particles, the time of the whole system iterating 16 times is saved (the average running time is 9.1 s).

Figure 12 shows that the initial positioning errors are all within 10 m, which is acceptable for indoor positioning. The experimental results and the positioning error analysis based on the improved particle filter algorithm are shown

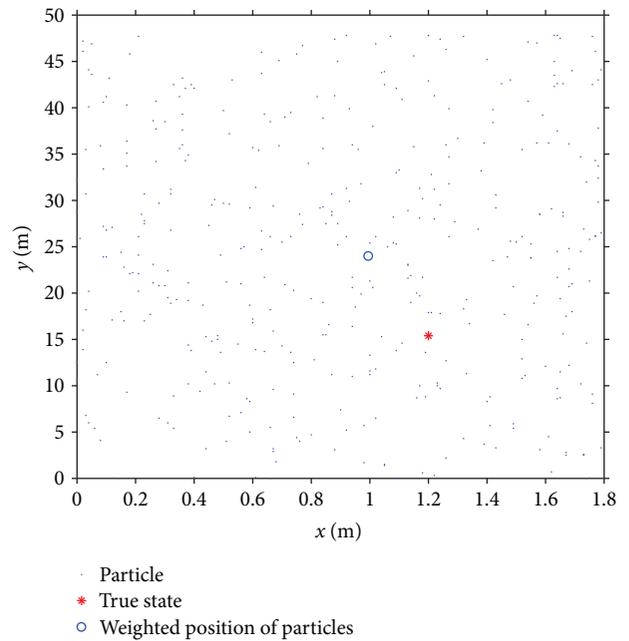


FIGURE 10: Initial positioning of  $t = 1$ .

in Figure 13. In the experiments using the improved algorithm, most particles began to converge at  $t = 2$  and have no loss of positioning as in Figure 9, with an average matching accuracy improved to 0.62 meters. Compared to the particle filter with the SIR algorithm, the improved algorithm can effectively prevent the filter divergence, eliminating location losses. Its convergence rate is faster than that of the classic particle filter algorithm, with an average running time of the whole system iterating 16 times is reduced by 16.15%.

### 5. Conclusions and Future Work

Indoor geomagnetic disturbance prevents the classical particle filter algorithm from stably finding the location in real time. Although the particle filter algorithm has a strong convergence in the matching accuracy, the particle iteration

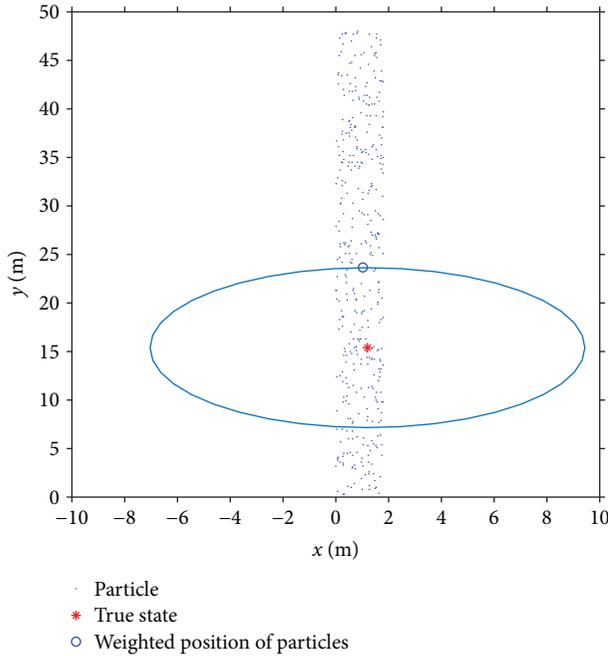


FIGURE 11: Positioning error constraint.

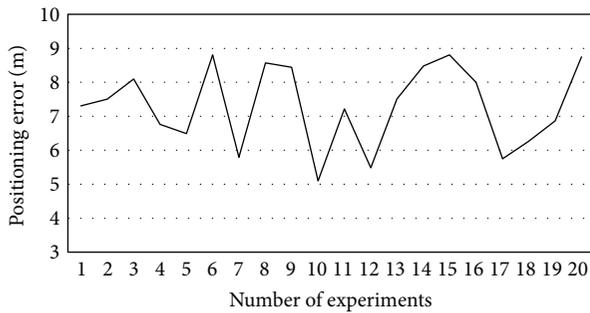


FIGURE 12: Initial positioning errors. Maximum value: 8.81 m, average value: 7.30 m, and minimum value: 5.10 m.

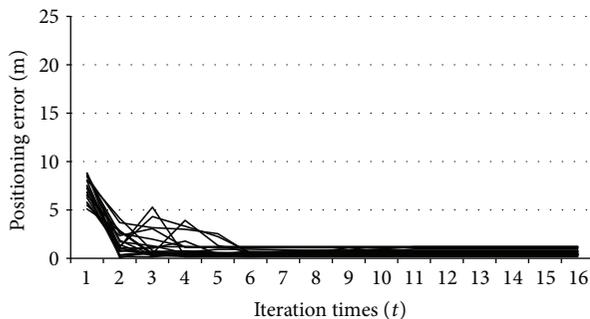


FIGURE 13: Positioning errors based on improved algorithm.

process is accompanied by the loss of positioning. In this paper, we add the position error constraint to the particle filter algorithm and compare and analyze the advantages and disadvantages of the classical particle filter algorithm

with the improved algorithm in matching accuracy, running time, and other aspects by real-field test. The results show that the improved algorithm can solve the persistent divergence problem in the particle filter and avoid the loss of positioning. Under the premise of the single variable feature, it can improve the indoor positioning speed and effectively solve the technical defect in geomagnetic matching. In the future work we will try to combine WLAN and other technologies to achieve effective initial positioning accuracy better than 5 meters.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This study was supported by National Key Research and Development Program of China (no. 2017YFB0503702).

## References

- [1] M. Angermann, M. Frassl, M. Doniec, B. J. Julian, and P. Robertson, "Characterization of the indoor magnetic field for applications in localization and mapping," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–9, Sydney, NSW, Australia, November 2012.
- [2] C. Deng, C. Huang, H. Zhao, and H. Tian, "Review of geomagnetic matching navigation algorithm," *Science Technology and Engineering*, vol. 12, no. 24, pp. 6125–6131, 2012.
- [3] Y. Qiao, S. Wang, J. Zhang, and Z. Wang, "Simulation of geomagnetic matching algorithm based on correlation analysis for flying objects," *Progress in Geophysics*, vol. 24, no. 2, pp. 721–727, 2009.
- [4] Y. Kong, L. Hao, and J. Xu, "Study on geomagnetic matching algorithm based on Hausdorff distance," *Aero Weaponry*, vol. 4, pp. 26–29, 2011.
- [5] X. Wang and Y. Tian, "Autonomous navigation based geomagnetic research," *Chinese Journal of Geophysics*, vol. 53, no. 11, pp. 2724–2731, 2010.
- [6] G. Yang, S. Li, and Z. Jiang, "Data fusing algorithm in geomagnetic aided INS," *Journal of Chinese Inertial Technology*, vol. 15, no. 1, pp. 47–50, 2007.
- [7] L. C. Boles and K. J. Lohmann, "True navigation and magnetic maps in spiny lobsters," *Nature*, vol. 421, no. 6918, pp. 60–63, 2003.
- [8] W. Storms, J. Shockley, and J. Raquet, "Magnetic field navigation in an indoor environment," in *2010 Ubiquitous Positioning Indoor Navigation and Location Based Service*, pp. 1–10, Kirkkonummi, Finland, October 2010.
- [9] B. Li, T. Gallagher, A. G. Dempster, and C. Rizos, "How feasible is the use of magnetic field alone for indoor positioning?," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–9, Sydney, NSW, Australia, November 2012.
- [10] J. Chung, M. Donahoe, C. Schmandt, I. J. Kim, P. Razavai, and M. Wiseman, "Indoor location sensing using geo-magnetism," in *Proceedings of the 9th international conference on Mobile systems, applications, and services - MobiSys '11*, pp. 141–154, Bethesda, MD, USA, June–July 2011.

- [11] R. Hostettler and S. Särkkä, "IMU and magnetometer modeling for smartphone-based PDR," in *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–8, Alcalá de Henares, Spain, October 2016.
- [12] E. Le Grand and S. Thrun, "3-axis magnetic field mapping and fusion for indoor localization," in *2012 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 358–364, Hamburg, Germany, September 2012.
- [13] J. Haverinen and A. Kemppainen, "Global indoor self-localization based on the ambient magnetic field," *Robotics and Autonomous Systems*, vol. 57, no. 10, pp. 1028–1035, 2009.
- [14] J. Haverinen and A. Kemppainen, "A global self-localization technique utilizing local anomalies of the ambient magnetic field," in *2009 IEEE International Conference on Robotics and Automation*, pp. 3142–3147, Kobe, Japan, May 2009.
- [15] S. E. Kim, Y. Kim, J. Y. Yoon, and E. S. Kim, "Indoor positioning system using geomagnetic anomalies for smartphones," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–5, Sydney, NSW, Australia, November 2012.
- [16] Z. Guo, L. Luan, H. Zhu, and P. Cai, "Geomagnetic filtering navigation algorithm based on single spot matching precorrection," in *2012 Fifth International Symposium on Computational Intelligence and Design*, pp. 543–546, Hangzhou, China, October 2012.
- [17] S. Q. Hu, *The Principle and Application of Particle Filter*, Science Press Co. Ltd, 2010.
- [18] H. Huang, Y. Zhao, C. L. Wang, and G. Yu, "Design of the acquisition system of indoor positioning reference map based on magnetic field data," *Bulletin of Surveying and Mapping*, vol. 2, pp. 54–59, 2017.
- [19] W. Shao, F. Zhao, C. Wang et al., "Location fingerprint extraction for magnetic field magnitude based indoor positioning," *Journal of Sensors*, vol. 2016, Article ID 1945695, 16 pages, 2016.

## Research Article

# Practical In Situ Implementation of a Multicamera Multisystem Calibration

Ivan Detchev <sup>1</sup>, Ayman Habib,<sup>2</sup> Mehdi Mazaheri,<sup>1</sup> and Derek Lichti<sup>1</sup>

<sup>1</sup>Department of Geomatics Engineering, University of Calgary, 2500 University Dr NW, Calgary, AB, Canada T2N 1N4

<sup>2</sup>Lyles School of Civil Engineering, Purdue University, 550 Stadium Mall Dr West, Lafayette, IN 47907-2051, USA

Correspondence should be addressed to Ivan Detchev; [i.detchev@ucalgary.ca](mailto:i.detchev@ucalgary.ca)

Received 24 May 2017; Revised 4 October 2017; Accepted 31 October 2017; Published 7 February 2018

Academic Editor: Marco Scaioni

Copyright © 2018 Ivan Detchev et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Consumer-grade cameras are generally low-cost and available off-the-shelf, so having multicamera photogrammetric systems for 3D reconstruction is both financially feasible and practical. Such systems can be deployed in many different types of applications: infrastructure health monitoring, cultural heritage documentation, bio-medicine, as-built surveys, and indoor or outdoor mobile mapping for example. A geometric system calibration is usually necessary before a data acquisition mission in order for the results to have optimal accuracy. A typical system calibration must address the estimation of both the interior and the exterior, or relative, orientation parameters for each camera in the system. This article reviews different ways of performing a calibration of a photogrammetric system consisting of multiple cameras. It then proposes a methodology for the simultaneous estimation of both the interior and the relative orientation parameters which can work in several different types of scenarios including a multicamera multisystem calibration. A rigorous in situ system calibration was successfully implemented and tested. The same algorithm is able to handle the equivalent to a traditional-style bundle adjustment, that is, a network solution without constraints, for a single or multicamera calibrations, and the proposed bundle adjustment with built-in relative orientation constraints for the calibration of a system or multiple systems of cameras.

## 1. Introduction

A photogrammetric system consists of one or multiple digital cameras. In the case of a single camera [1], it would have to be moving or sequentially occupying varying camera stations. This scenario would only work for objects that remain shape-invariant throughout the image recording session due to the time lapse between the varying camera station exposures. Preferably, an array or a cluster of cameras should be used in the scenarios where the shape of the object of interest may be changing with time [2]. The availability of inexpensive digital cameras has made the use of such multisensor systems more and more common. Their employment in mobile mapping applications [3, 4], dense matching of imagery [5, 6], biomedical and motion-capture metric applications [2, 7], infrastructure health monitoring [8, 9],

and the generation of photo scenes from multiple sensors [10] has become frequent occurrence.

Sensor calibration is known to be a critical quality assurance measure to maximize photogrammetric accuracy. This is even more so in the case of a multicamera system. A correct system calibration is essential for the accurate reconstruction of 3D object space required in photogrammetric applications. For this purpose, a mathematical model based on built-in relative orientation constraints (ROCs) is reviewed and further improved by modifying it to handle both single and multiple reference camera(s).

The next section summarizes the different types of system calibrations depending on whether the cameras are precalibrated or calibrated in situ and whether the estimation of the relative orientation parameters is performed in a two- or a one-step procedure. The preferred system calibration

method for the simultaneous estimation of all system calibration parameters is proposed and tested.

## 2. System Calibration Methodology

The geometric calibration of a system comprising multiple cameras has two components: a camera calibration of each camera in the system and an estimation of the position and orientation of the cameras involved in the system with respect to a reference camera. The next subsections discuss various options for accomplishing the estimation of a system calibration.

**2.1. Solution for the Interior Orientation Parameters.** The camera calibration necessitates estimating the interior orientation parameters (IOPs) of each camera, which include the principal distance, the principal point offset, and any necessary distortion or additional parameters. This has been heavily addressed in photogrammetric literature. For consumer-grade digital cameras that can be purchased off-the-shelf, the preferred procedure for estimating the IOPs is a bundle adjustment with self-calibration [11, 12]. The distortion models are found in Brown [13] and Brown [14], while the analytical basis for the adjustment is published in Kenefick et al. [15] and Granshaw [16]. Clarke and Fryer [17] and Remondino and Fraser [18] include recommendations on how to carry out the calibration procedure for a single camera. In addition, Chandler et al. [19] and Fraser [20] show examples for the calibration of digital cameras that are specifically low-cost/off-the-shelf.

The calibration process can be performed in a specialized laboratory or on the job (i.e., in situ). Which type of calibration should be chosen depends on the project specifics. In the case of a photogrammetric system comprising a few cameras (e.g., two to three), the calibration process can successfully be carried out individually for each camera before the commencement of any data collection. Yet, if many cameras are involved in the system in question (e.g., four or more), precalibrating each one of them individually might be too time-consuming. Additionally, it may not be desirable to dismount and then remount the cameras from the system platform every time they need to be recalibrated. In such circumstances, performing the camera calibration on the job or in situ may be more practical and/or feasible. The challenge of such an in situ system calibration is the first order network design problem, that is, having a network geometry that will produce a solution for the unknown parameters with an acceptable variance-covariance matrix. For instance, sufficient number of target points with a well-distributed spread within the image format must be present for each camera. At the same time, multistation convergent images must be taken such that isotropic coordinate precision for the object space reconstruction is achieved. For a stationary camera system, this network configuration can be emulated by conducting numerous translations and rotations of a portable test field within the field of view of the cameras in the system [21–25]. It is worth clarifying that even though the camera system is physically stationary and the test field is

the one translated and rotated, the adjustment is handled inversely as if the test field is kept stationary and the camera system is the one moving. In this way, for each instance of translation and rotation, the number of exterior orientation parameters (EOPs) added to the adjustment is lesser than the number of 3D coordinates for the object space target points, and the total number of unknowns in the adjustment is thus minimized. Note that ideally this portable test field should be in 3D so that any projective compensation or high correlations within and between the IOPs and EOPs can be decoupled [15, 22–25].

**2.2. Solution for the Camera Mounting Parameters.** Before the beginning of a data collection campaign, the position and the orientation of the cameras in the system must be estimated in addition to the IOPs. This can be done with respect to a reference camera or some other type of a reference frame (e.g., an IMU body frame). These parameters are referred to as the relative orientation parameters (ROPs) or the camera mounting parameters (CMPs), that is, the parameters describing how each camera is attached to the system platform. Assuming that the CMPs are defined relative to a reference camera, they consist of positional,  $r$ , and rotational,  $R$ , offsets between each camera and the reference camera. These components can also be referred to as the lever arm (baseline) and (angular) boresight, respectively. The estimation of the CMPs can be done in a two-step or a one-step process [26]. Both type of processes are reviewed in the next two subsections.

**2.2.1. Two-Step CMP Estimation.** The first step in the two-step procedure for providing a solution for the CMPs is estimating the EOPs for each of the cameras in the system. A traditional-style bundle adjustment, that is, a network solution based on the collinearity equations and without any constraints, is normally used for the purpose of completing this first step (see (1) and (2)). Note that time dependency is assumed here in order to have a general model, that is, a model that can handle both stationary or moving sensors or objects.

$$r_I^m = r_{c_k}^m(t) + \lambda_i^{c_k} \cdot R_{c_k}^m(t) \cdot r_i^{c_k}(t), \quad (1)$$

where  $r_I^m$  contains the coordinates of object space point,  $I$ , with respect to the mapping frame  $m$ ;  $r_{c_k}^m(t)$  and  $R_{c_k}^m(t)$  are the time-dependent positional and rotational parameters or the EOPs of camera  $c_k$  with respect to the mapping frame  $m$  at time  $t$ ;  $\lambda$  is the image to object space scale; and the expression

$$r_i^{c_k}(t) = \begin{bmatrix} x_i^{c_k} - x_p^{c_k} - \Delta x_i^{c_k} \\ y_i^{c_k} - y_p^{c_k} - \Delta y_i^{c_k} \\ -c^{c_k} \end{bmatrix} \quad (2)$$

contains the distortion-free coordinates of image space point,  $i$ , or the distortion-free projection of point  $I$  in the frame of camera  $c_k$ , where  $(x_i^{c_k}, y_i^{c_k})$  are the observed or distorted image coordinates for point  $i$ ;  $(x_p^{c_k}, y_p^{c_k})$  is the principal point

offset;  $c_k$  is the principal distance; and  $(\Delta x_i^{c_k}, \Delta y_i^{c_k})$  are the image space distortions for point  $i$ .

The second step in this process uses the estimated EOPs at time  $t$  to compute the CMPs using (3) and (4) [4].

$$r_{c_k}^{c_r}(t) = \left( R_{c_r}^m(t) \right)^T \cdot \left( r_{c_k}^m(t) - r_{c_r}^m(t) \right), \quad (3)$$

where  $r_{c_k}^{c_r}(t)$  is the time-dependent 3D lever arm/positional offset or translation between camera  $c_k$  and the reference camera  $c_r$ , that is,  $[\Delta X_{c_k}^{c_r} \ \Delta Y_{c_k}^{c_r} \ \Delta Z_{c_k}^{c_r}]^T$ ;  $r_{c_r}^m(t)$ ; and  $R_{c_r}^m(t)$  are the time-dependent positional and rotational parameters or the EOPs of the reference camera  $c_r$  with respect to the mapping frame  $m$ ; and  $r_{c_k}^m(t)$  is the time-dependent positional component of the EOPs of camera  $c_k$ .

$$R_{c_k}^{c_r}(t) = \left( R_{c_r}^m(t) \right)^T \cdot R_{c_k}^m(t), \quad (4)$$

where  $R_{c_k}^{c_r}(t)$  is the time-dependent 3D boresight/rotational offset between camera  $c_k$  and the reference camera  $c_r$ , which is a function of  $\Delta \omega_{c_k}^{c_r}$ ,  $\Delta \phi_{c_k}^{c_r}$ , and  $\Delta \kappa_{c_k}^{c_r}$ , and  $R_{c_k}^m(t)$  is the time-dependent rotational component of the EOPs of camera  $c_k$ .

If the EOPs are estimated in a single observation epoch, there would accordingly be a single set of computed CMPs. If the EOPs are, however, estimated in two or more observation epochs, the resultant redundant sets of time-dependent CMPs can be averaged and their standard deviations can be calculated [4, 22–25]. Note, however, that rotation averaging should not be performed with Euler angles [27]. The point of averaging is to compute the best estimate of a random variable while minimizing the sum of squared errors. Averaging Euler angles, that is, sequential rotational parameters, does not minimize a meaningful cost function. Instead, for sound rotation averaging, quaternions or angle-axis representation must be used as they can minimize several cost functions, which are listed in Hartley et al. [27]. In addition, the rotational and positional parameters are often correlated. It should also be highlighted that if the reference camera does not observe the test field in certain observation epochs, these observation epochs cannot contribute to the estimations of the CMPs. Moreover, if the field of view of a particular camera does not overlap with the one for the reference camera (i.e., the two cameras cannot observe the test field simultaneously in any of the observation epochs), the CMPs of the camera in question cannot be directly estimated as in (3) and (4). A work-around procedure exploiting the overlap with other cameras in the system must be then implemented.

Using constraint equations for the EOPs in the network solution in order to enforce an invariant geometrical relationship between the cameras at different times [10, 28–33] may mitigate some of the mentioned problems. The benefit of using EOP constraint equations in the bundle adjustment is that it is not necessary to perform any averaging. This is the case since no matter which observation epoch is used, the same values for the CMPs would be computed with (3) and (4). The disadvantages of this method are that it is still technically a two-step process, a work-around procedure is necessary for computing the CMPs of any cameras, which

do not overlap with the reference camera in any of the observation epochs, and the complexity of the implementation procedure intensifies with the increase of the number of cameras in the system and the number of observation epochs [4]. Given these drawbacks, especially the complexity consideration, this type of network solution is not implemented as part of this research.

**2.2.2. One-Step CMP Estimation.** A one-step procedure for estimating the CMPs is desired in order to avoid a separate estimation step for the CMPs and potential work-around procedures for addressing situations where a camera does not have any overlap with the reference camera in any observation epoch. This can be accomplished by directly incorporating ROCs among all cameras and the reference camera in the collinearity-equation-based network solution [4, 29, 30].

$$r_I^m = r_{c_r}^m(t) + R_{c_r}^m(t) \cdot r_{c_k}^{c_r} + \lambda_i^{c_k} \cdot R_{c_r}^m(t) \cdot R_{c_k}^{c_r} \cdot r_i^{c_k}(t). \quad (5)$$

The CMPs,  $r_{c_k}^{c_r}$  and  $R_{c_k}^{c_r}$ , in (5) are now explicitly treated as time-independent parameters. In other words, it is assumed that the CMPs for the system of multiple cameras remain stable during all observation epochs within a given system calibration campaign. Also, the EOPs of the reference camera,  $r_{c_r}^m(t)$  and  $R_{c_r}^m(t)$ , now represent the EOPs of the system platform. This model is relatively straightforward to implement as it preserves its simplicity regardless of the number of cameras employed or the number of observation epochs used. As the number of cameras and the number of observation epochs increase, it significantly reduces the number of unknowns to solve compared to the traditional-style bundle adjustment. It should be noted that when the observation equations for the reference camera,  $c_r$ , are established, (5) reduces to (6), because the lever arm vector is set to zero,  $r_{c_r}^{c_r} = [0 \ 0 \ 0]^T$ , and the boresight rotation matrix is set to identity,  $R_{c_r}^{c_r} = [1 \ 0 \ 0; \ 0 \ 1 \ 0; \ 0 \ 0 \ 1]$ . The difference between the mathematical models for the traditional-style bundle adjustment (1) and the one with built-in ROCs (5) is visually summarized in Figure 1.

$$r_I^m = r_{c_r}^m(t) + \lambda_i^{c_k} \cdot R_{c_r}^m(t) \cdot r_i^{c_k}(t). \quad (6)$$

**2.3. Simultaneous IOP and CMP Estimation.** There are four possible types of a calibration of a system with multiple cameras that can be performed given the options for estimating the IOPs and the CMPs. Recall that individual cameras can be either precalibrated or multiple cameras can be calibrated in situ and that the CMPs can be either estimated via the EOPs in a two-step process or directly in a one-step process. The four scenarios are summarized in Table 1 and explained in more detail as follows:

- (1) The first system calibration scenario involves precalibrating all cameras individually and computing the CMPs in a two-step procedure via the EOPs. It is not desirable, because it may not be practical to precalibrate many cameras and/or the IOPs of the precalibrated cameras may not be stable. In addition, in situations where the reference camera does not

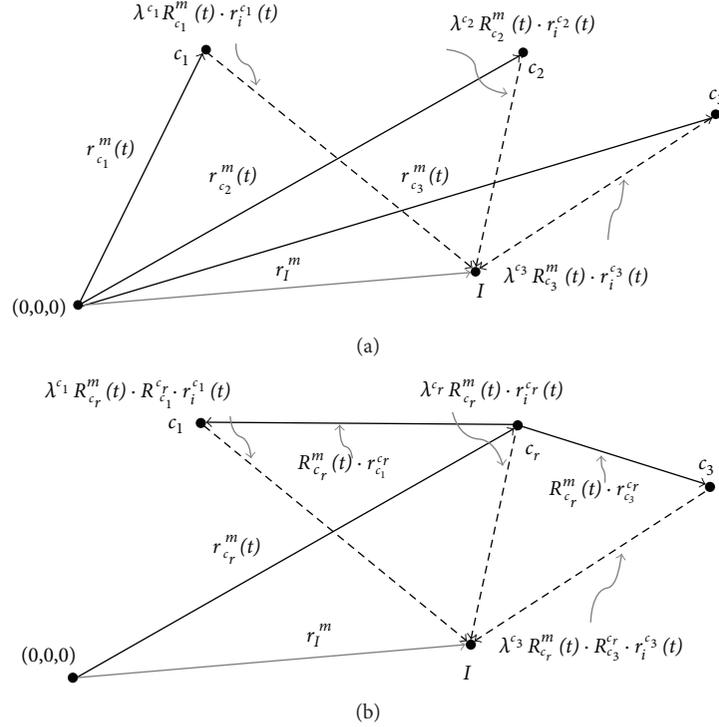


FIGURE 1: Mathematical model for 3D reconstruction using the traditional-style collinearity-equation-based bundle adjustment with no constraints (a) versus the one using built-in ROCs (b);  $(0,0,0)$  denotes the origin of the mapping frame;  $c_1$ ,  $c_2$ , and  $c_3$  are example cameras 1, 2, and 3, respectively; and  $c_r$  is the reference camera.

TABLE 1: Possible types of system calibrations.

IOPs/CMPs	Two-step CMP estimation	One-step CMP estimation
Individual cameras, precalibrated	(1) Not desirable (IOPs $\rightarrow$ EOPs $\rightarrow$ CMPs)	(3) May be acceptable (IOPs $\rightarrow$ CMPs)
Multiple cameras, calibrated in situ	(2) More practical but still not preferred (IOPs & EOPs $\rightarrow$ CMPs)	(4) Most desirable (IOPs & CMPs)

observe the test field in certain observation epochs, or certain cameras do not have any overlap with the reference camera, the CMP estimation may be ambiguous or not even possible.

- (2) The second system calibration scenario involves calibrating all cameras in situ and computing the CMPs in a two-step procedure via the EOPs. Note that the IOPs and the EOPs here are estimated simultaneously. As long as the necessary network configuration can be provided, this procedure is more practical in terms of the camera calibration aspect. It is, however, still not a preferred approach due to the drawbacks listed with regard to the two-step CMP estimation.
- (3) The third system calibration scenario involves precalibrating all cameras individually and estimating the CMPs in a single step. This procedure may be acceptable for a system with a low number of cameras (e.g., one to three) as long as the individual camera IOPs remain stable from the point in time when the IOPs

are estimated to the point in time when the cameras are set up for the CMP estimation.

- (4) The fourth system calibration scenario involves calibrating all cameras in situ and estimating the CMPs in a single step. Note that the IOPs and the CMPs here are estimated simultaneously. Again, as long as the necessary network configuration can be provided, this procedure is the most desirable one because all unknown parameters are estimated simultaneously in a single step.

**2.4. Multicamera Multisystem Calibration.** The ability to handle multiple reference cameras in the bundle adjustment with built-in relative orientation constraints is a valuable contribution of the proposed procedure for system calibration. Essentially, all cameras participating in the system calibration adjustment are assigned to or selected as a reference camera, and there can be as many reference cameras as the total number of participating cameras. Thus, the adjustment is not limited to a single reference camera.

This is significant, because it provides the flexibility of performing the following types of calibrations within the same algorithm:

- (i) Calibration of a single camera—the camera in question is selected as a reference camera; this scenario reduces to a regular single camera calibration, where the unknowns are the IOPs of the camera, the EOPs for each image, and the X,Y,Z coordinates for all the target points.
- (ii) Calibration of a single system of multiple cameras—all the available cameras are assigned to the same reference camera; other than the IOPs for all the cameras and the X,Y,Z coordinates for all the target points, the unknowns here are the EOPs for the one reference camera and the CMPs for all other cameras with respect to the reference camera.
- (iii) Calibration of multiple systems with multiple cameras within the same adjustment—the cameras are divided into groups where each group is assigned a different reference camera; note that in the most general case, each group corresponds to a physically different system. Other than the IOPs and X,Y,Z coordinates already mentioned, the unknowns in this scenario are the EOPs for the multiple reference cameras and the CMPs for the remaining cameras in each system with respect to their corresponding reference camera; note that the observations coming from a particular camera affect the EOPs of only its corresponding reference camera.
- (iv) Calibration of multiple cameras without any ROCs within the same adjustment—if multiple cameras need to be calibrated, but they are not employed within a stable system, each camera is selected as a reference camera; this scenario reduces to a regular multiple camera calibration, and it is basically a special case of the previous one where each camera is treated as its own system.

### 3. Experiment Setup

A multicamera system was employed in a structures laboratory for the purposes of this research initiative. This system is described here, as are both a newly designed calibration test field and a routine for acquiring calibration data.

**3.1. Example System Setup.** The photogrammetric system employed in this research work consisted of eight digital cameras. The camera bodies used were Canon EOS 1000D/Rebel XS DSLR with Canon EF-S 18–55 mm  $f/3.5-5.6$  zoom kit lenses. A 22.2 mm  $\times$  14.8 mm complementary metal oxide semiconductor (CMOS) solid state sensor was present in each camera body [34]. Image resolution of 10.1 mega pixels (i.e., 3888 pixels along the image width and 2592 pixels along the image height) was possible. A square pixel size with a nominal dimension of 5.71  $\mu\text{m}$  was assumed. The pixel size was computed by dividing the dimensions of the effective

sensor size by the number of pixels in the corresponding direction. For this research project, the cameras were mounted on a steel frame via tripod heads with three degrees of freedom. The tripod heads were necessary to point the cameras towards the specimen of interest. The cameras were configured and synchronized so that nonblurred images of both stationary/static and kinematic/dynamic objects could be taken. After the cameras were focused on the specimen surface, the focus and zoom rings of the lenses were locked with electrical tape, and the auto-focus, the vibration reduction, and the sensor cleaning functions of the cameras were disabled in order to minimize any potential camera instability as much as possible.

The camera system was suspended from an overhanging metal frame (see Figure 2). This configuration was chosen so that the system can observe the top surface of a concrete beam while it was being deformed by a hydraulic actuator. The array of cameras was set up in such a way that the optical axes of the cameras were converging as much as possible at the object of interest. The convergence angle between consecutive cameras varied from 0° to 20° and was approximately 70° between the first and last cameras (see Figure 3). This convergent network geometry allows for near-isotropic coordinate precision to be achieved. The nominal principal distance of the cameras varied between 22 mm and 28 mm. Principal distances on the lower end were used for the central cameras, while ones on the higher end were set for the end cameras.

**3.2. Portable Calibration Test Field and Routine for In Situ System Calibration.** Before the commencement of any actual experiments, in situ calibration data had to be collected by the system. The time allocated for the collection of the calibration data was approximately 15 to 30 minutes. The in situ calibration data collection involved translating and rotating a portable test field above the object(s)/surface(s) of interest. The test field used in this research was a 2D reinforced plywood sheet with a four by three grid of coded targets and an eleven by eight grid of checkerboard targets (see Figure 4(a)). While a 3D test field would have been preferred, the use of a 2D test field was more practical given the laboratory conditions and the object of interest at hand. The coded targets were only used for automating the target labelling and solving the image-point correspondence problem. The target coding system and the software used to automatically identify the codes were created in-house. The image point measurements for the checkerboard targets were first made with the Harris corner detector. Then, a subpixel optimization algorithm in OpenCV [35] was used to further refine them. The centre of the test field was the origin of the local coordinate system. If the test field was placed in front of the system, the orientation of the X-, Y-, and Z-axes was chosen so that the Z-axis would point in the general direction of the z-axis of the reference camera (see Figure 4(b)). Additionally, a piece of the test field was cut out by design. This allowed for the test field to fit around the piston of the hydraulic actuator at the cost of losing only three checkerboard targets (see Figure 5).

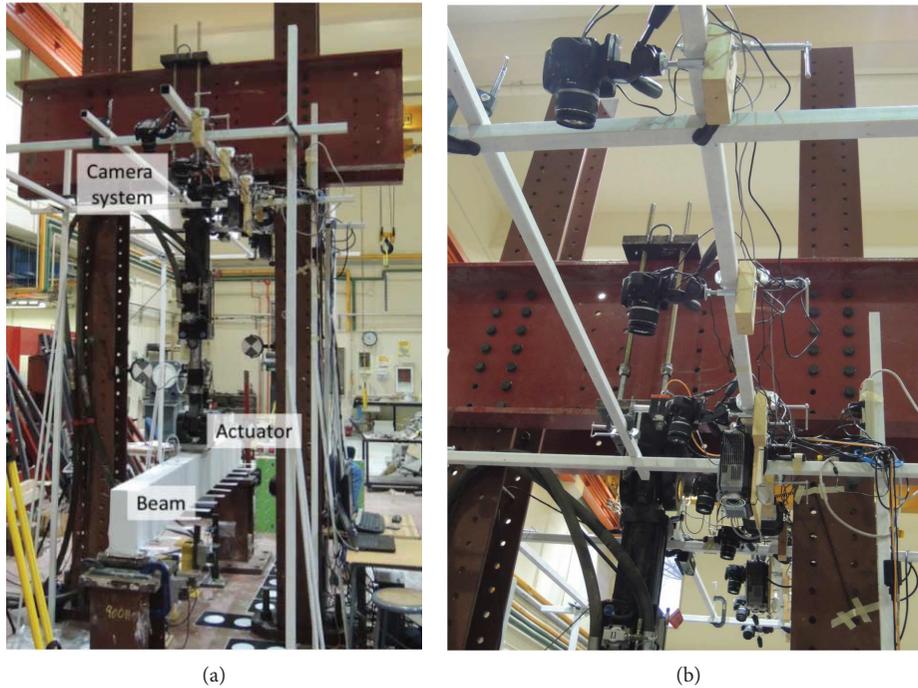


FIGURE 2: Suspended setup showing the hydraulic actuator, the concrete beam specimen (a), and a close-up of the camera system (b); note that the circular targets on the floor and the circular checkerboard targets on the actuator frame were placed for another project and are not relevant for this research.

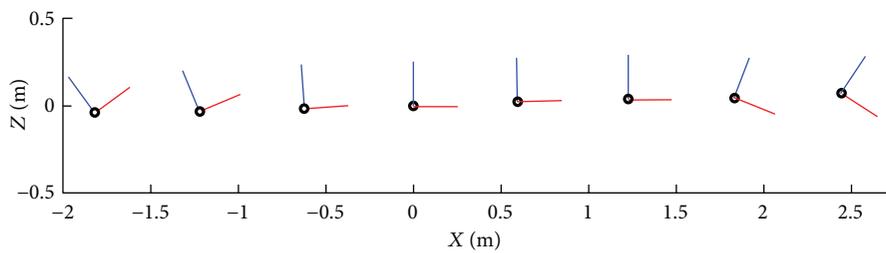


FIGURE 3: Camera configuration for the suspended system setup; the camera stations are shown as black circles, and the convention used for the camera axes is as follows:  $x$ -axis (red) is along the object of interest,  $y$ -axis (not shown) is across the object of interest, and  $z$ -axis (blue) is away from the object of interest.

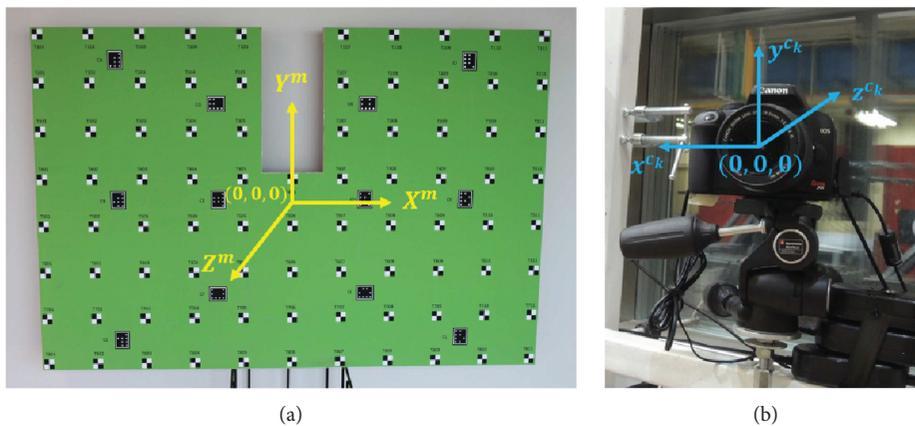


FIGURE 4: The portable calibration test field showing the origin and the orientation of the local coordinate system (a) and the coordinate system used for a particular camera (b).

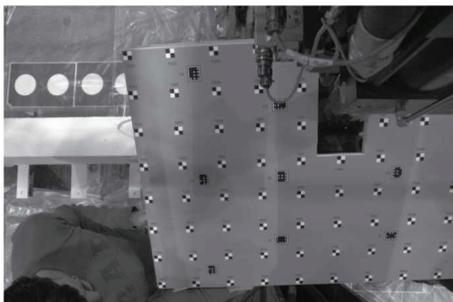


FIGURE 5: Sample placement of the portable test field around the piston of the hydraulic actuator.

Series of images of the test field were taken simultaneously from as many cameras as possible. Given the test field employed was in 2D, care was taken to partially mitigate any projective compensation or high correlations between the interior and exterior orientation parameters, namely, convergent network geometry, and rolls of the test field were implemented. In each observation epoch, the position and orientation of the test field with respect to the reference camera were altered. For example, with  $\kappa \cong 0^\circ$  or  $180^\circ$  (i.e., in “landscape” mode), the test field was translated under each camera with  $\omega$  and  $\varphi$  being nominally  $0^\circ$ . Four more translation rounds were repeated where the test field was rotated around the  $X$ - and  $Y$ -axes in such a way that the range in  $\omega$  and  $\varphi$  was between  $30^\circ$  and  $65^\circ$ . The test field was then rotated with  $\kappa \cong +90^\circ$  or  $-90^\circ$  (i.e., in “portrait” mode), and five more similar translation rounds were performed.

In order to evenly fill out the entire usable image format of each camera with targets, the described translations and rotations of the test field were performed under each camera. This target distribution was required so that the lens distortion coefficients for each camera could be estimated reliably. Figure 6 shows two examples of all the image coordinate measurements acquired within the format of a particular camera superimposed on a photograph taken by that camera.

## 4. Experimental Results

The experimental results related to the proposed multicamera multisystem calibration methodology are presented next. The tests for estimating the IOPs, the CMPs, or the IOPs and CMPs simultaneously are as follows:

- (i) Estimation of IOPs in an individual camera calibration versus calibration of multiple cameras simultaneously with no ROCs versus calibration of multiple cameras as a system with built-in ROCs
- (ii) Estimation of CMPs in a two-step calibration versus a one-step calibration with or without prior knowledge of the IOPs
- (iii) Estimation of all system calibration parameters (i.e., IOPs and CMPs) simultaneously for individual multicamera systems versus for multiple multicamera systems within the same adjustment

Comparison of the various calibration parameter outcomes is also performed using a previously developed system stability analysis tool [26].

Note that for all the bundle adjustments run in the experimental results, the coordinate frame was always the same. The datum was defined by fixing six coordinates and by including observations for the six spatial distances between the four outermost targets on the test field. Thus, the datum definition was minimally constrained (with redundancy for the scale). Also note that except for the six fixed coordinates, all coordinates for the targets on the board were used as unknowns in the adjustments. Also, note that while not all targets were observed in each acquired image, all targets were used in all solutions.

**4.1. IOP Estimation Test.** After an in situ system calibration data set was acquired using the portable test field, the data were processed in three different ways in order to assess the estimation quality of the IOPs.

- (i) A separate bundle adjustment was run for each camera in order to calibrate each camera individually.
- (ii) A single bundle adjustment without using any ROCs was run with all cameras in order to calibrate all the cameras simultaneously.
- (iii) A single bundle adjustment with a reference camera and built-in ROCs was run in order to calibrate the cameras as a system.

The results from the IOP estimation test are summarized in Table 2. A “per camera normalization” was applied to some of the adjustment quantities in the multiple camera calibration and system calibration columns, that is, the quantity in question was divided by the number of cameras for a more objective comparison. Note that even though some variation in the quality of the image coordinate measurements was present in the different adjustments, a thorough investigation as to whether it is necessary to perform any reweighting of the measurements has not yet been conducted. While the three types of adjustments had similar overall image coordinate measurement precision,  $RMSE_{xy}$ , the range for the IOP standard deviation values,  $\sigma$ , improved for the multiple camera calibration and for the system calibration. The extra intersecting light rays, coming from the increased number of image point observations for the same object space targets, increased the redundancy and strengthened the adjustment geometrically [36]. Figure 7 shows a comparison between the network geometry for a single camera calibration versus all cameras involved in the system. Additionally, the decrease in the number of unknowns for the system calibration further improved the redundancy compared to the multiple camera calibration. It can be thus concluded that in terms of the IOP estimation, the proposed system calibration yielded the strongest solution.

The same additional parameter (AP) model was used in all three types of bundle adjustments for the conducted IOP test. It was decided that the  $k_1$ ,  $k_2$ ,  $p_1$ , and  $p_2$  coefficients were sufficient to adequately model the systematic error present in

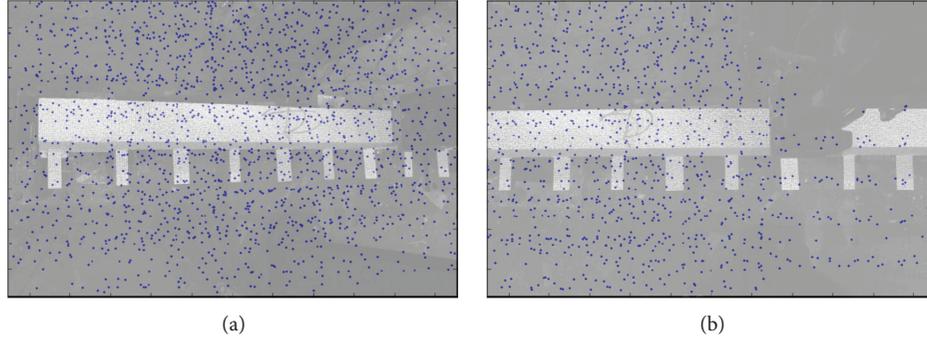


FIGURE 6: Examples of an entire image format (a) and of a usable portion of an image format (b) for different cameras being filled with targets.

TABLE 2: Results for the IOP estimation test of the cameras belonging to the system.

Adjustment quantities/types of camera calibration	Individual camera calibration	Multiple camera calibration	System calibration
RMSE <sub>xy</sub> [ $\mu\text{m}$ ]	0.78–1.05	0.90	1.03
Number of images	17–18	139	139
Norm'd number of images	17–18	17.4	17.4
Number of image points	783–1086	7649	7649
Norm'd number of image points	783–1086	956.1	956.1
Average number of points/images	45.7–63.9	55.0	55.0
Number of epochs	17–18	41	41
Average number of images/epochs	1	3.4	3.4
$c^k, x_p, y_p, \sigma$ range [ $\mu\text{m}$ ]	2–8	2–5	2–4
Number of unknowns	364–370	1145	599
Norm'd number of unknowns	364–370	143.1	74.9
Redundancy	1196–1808	14,153	14,699
Norm'd redundancy	1196–1808	1769.1	1837.4

the image coordinate observations for the cameras involved in the system. Without these APs, the size of some of the residuals was several times the size of a pixel, which was deemed unacceptably large.

**4.2. CMP Estimation Tests.** The quality of the CMP estimation was also tested with the in situ system calibration data set. Four system calibration scenarios were tested.

- (1) A traditional-style bundle adjustment where the IOPs for all the cameras were taken from the individual camera calibrations in the previous subsection and were kept as fixed constants
- (2) A self-calibrating traditional-style bundle adjustment
- (3) The proposed bundle adjustment with a reference camera and built-in ROCs where the IOPs for all cameras were again taken from the individual camera calibrations in the previous subsection and were kept as fixed constants
- (4) The proposed bundle adjustment with a reference camera and built-in ROCs in self-calibrating mode

Note that these four system calibration scenarios correspond to the types of system calibrations described in Section 2.3 and listed in Table 1. The results from the CMP estimation test are summarized in Table 3. As previously explained, the former two approaches require two steps for the CMP estimation. For this system, there were many observation epochs where the reference camera did not observe the test field, and the majority of the cameras did not have any overlap with the reference camera in any of the observation epochs; in fact, the test field was only seen by two to four cameras at a time. Thus, none of the CMPs can be derived for all observation epochs, and the CMPs of some cameras can only be derived through a daisy chain with one or more intermediate cameras. It is suggested that some sort of weighted averaging or an additional higher level adjustment must be performed in order to avoid an arbitrary solution, that is, to achieve a proper CMP estimation. Such solution was however not implemented for this paper.

The latter two approaches, as previously explained, were able to solve for the CMPs in one step. It should be emphasized that while the method where precalibrated cameras were used seemed to have better CMP standard deviation values,  $\sigma$ , especially for the rotational CMPs; the method with

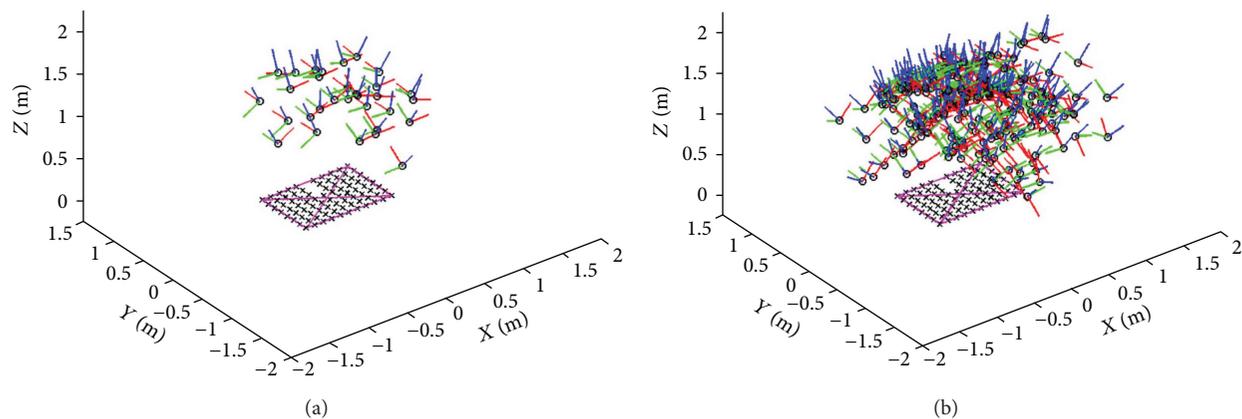


FIGURE 7: Example network geometry for the calibration of a single camera (a) versus the simultaneous calibration of all cameras involved in the system (b). The camera stations are shown as black circles, the camera  $x$ -axis is in red, its  $y$ -axis is in green, and its  $z$ -axis is in blue. The checkerboard targets on the portable test field are shown as black crosses, and the magenta lines indicate distance measurements used for scale definition.

TABLE 3: Results for the CMP estimation test.

Adjustment quantities/types of system calibration	(1) Pre + trad	(2) Trad + self	(3) Pre + prop	(4) Prop + self
RMSE <sub>xy</sub> [ $\mu$ m]	0.99	0.90	1.20	1.03
Number of images	139	139	139	139
Number of image points	7661	7649	7661	7649
Average number of points/images	55.1	55.0	55.1	55.0
Number of epochs	41	41	41	41
Average number of images/epochs	3.4	3.4	3.4	3.4
Positional CMP $\sigma$ range [mm]	N/A	N/A	0.02–0.4	0.04–0.3
Rotational CMP $\sigma$ range ["]	N/A	N/A	2–12	4–40
Number of unknowns	1089	1145	543	599
Redundancy	14,233	14,153	14,779	14,699

pre: precalibration of individual cameras; trad: traditional-style bundle adjustment; prop: proposed bundle adjustment with built-in ROCs; self: self-calibration mode.

self-calibration is still the preferred one. This is because the system consists of eight cameras, and it is not practical to run nine different adjustments in order to solve for all the IOPs and CMPs. Also, if the cameras were truly precalibrated (i.e., they were calibrated in a different location prior to being installed in the structures laboratory), there would be no guarantee that the previously estimated IOPs would still be valid.

In addition to analyzing the standard deviations of the estimated CMPs, the two one-step system calibration approaches can also be compared using a method for system stability analysis referred to as “object space parallax in image space units” [26]. While the issue here is not a matter of actual (in)stability, this system stability analysis tool can be used to check the compatibility between the two different system calibration approaches. This method quantifies/provides a numerical measure of the differences between two sets of calibration parameters based on a number of image and object space simulations. The results are shown in Table 4. Note that Habib et al. [26] output an image space RMSE only,

TABLE 4: Check for the image and object space compatibility between the two system calibration approaches (i.e., calibration scenarios three and four).

Camera pairs/total RMSEs	Image space RMSE [px]	Object space RMSE [mm]
Cams 1 & 2	1.00	0.37–0.44
Cams 2 & 3	0.49	0.20–0.24
Cams 3 & 4	0.36	0.19–0.23
Cams 4 & 5	0.47	0.25–0.30
Cams 5 & 6	0.26	0.11–0.13
Cams 6 & 7	0.36	0.15–0.17
Cams 7 & 8	0.63	0.23–0.28

while here an object space RMSE is also shown. The object space RMSE is computed by scaling the image space RMSE by the ratio of the object-to-camera depth over the average value of the principal distance. Since the object-to-camera depth could vary, reasonable “near field” and “far field”

TABLE 5: Results for the individual versus multisystem calibration test.

Adjustment quantities/system calibrations	Day 1	Day 2	Day 3	Days 1 + 2 + 3
RMSE <sub>xy</sub> [ $\mu\text{m}$ ]	1.33	1.27	1.27	1.43
Number of images	220	229	217	666
Norm'd number of images	220	229	217	222
Number of image points	12,653	13,188	12,526	38,330
Norm'd number of image points	12,653	13,188	12,526	12,776.7
Average number of points/image	57.5	57.6	57.7	57.6
Number of epochs	68	67	66	201
Norm'd number of epochs	68	67	66	67.0
Norm'd number of images/epochs	3.2	3.4	3.3	3.3
$c^k, x_p, y_p, \sigma$ range [ $\mu\text{m}$ ]	1.4–3.8	1.3–3.4	1.3–3.5	1.4–3.9
Positional CMP $\sigma$ range [mm]	0.04–0.44	0.04–0.41	0.04–0.41	0.04–0.47
Rotational CMP $\sigma$ range ["]	3.6–40.6	3.4–37.5	3.5–37.7	3.6–42.0
Number of unknowns	755	749	743	1749
Norm'd number of unknowns	755	749	743	583
Redundancy	24,551	25,627	24,309	74,911
Norm'd redundancy	24,551	25,627	24,309	24,970.3

values are picked and a range of object space RMSEs is reported. The total image space RMSE value for any camera pair was one pixel or less, while the total object space RMSE value ranged from 0.11 mm to 0.44 mm depending on the camera pair and the camera to object distance. Since the 3D reconstruction for the beam in this research initiative was based on pixel level image matching, and the sought after object space precision was 0.5 mm, the compatibility between the two types of system calibration solutions was considered satisfactory.

*4.3. Individual versus Multisystem Calibration Tests.* Additional system calibration data sets were acquired during a multiday beam deflection experiment. Each calibration data set was collected before any actual deflection observations were made on that particular day. Three calibration data sets referred to as Day 1, Day 2, and Day 3 were processed in two different ways which are as follows:

- (i) As three individual multicamera system calibrations where each calibration data set was treated as a separate system
- (ii) As a multicamera multisystem calibration where the three calibration data sets were combined in a single adjustment, but the data sets from the different days were treated as separate systems

The results from the individual system calibrations can be seen under the Day 1, Day 2, and Day 3 columns of Table 5. The proposed multicamera system calibration was applied, where the IOPs and CMPs for all the cameras in the system were estimated simultaneously. Since the network geometry of the different data sets was kept consistent, the standard deviations for the estimated parameters had similar ranges. The only noticeable differences in the adjustment quantities were in the total number of image points. This

was due to the slight difference in the number of observation epochs and thus the number of images.

The result from the multisystem calibration can be seen under the Days 1+2+3 column of Table 5. Note that a “per system normalization” was applied to some of the adjustment quantities in the last column of the table, that is, some of the quantities were divided by the number of systems in order to make the comparisons more objective. It should also be noted that the object space coordinates of the portable test field were what tied the three systems in this adjustment. Thus, the assumption for this type of solution was that the test field did not deform during the multiday experiment. In the individual versus multicamera calibration test in Section 4.1, there was an improvement in both the network geometry and the redundancy of the bundle adjustment. Since the network geometry between the different systems was similar, in this individual versus multisystem calibration test, the improvement was only in the redundancy. Due to the greater redundancy, it would be expected that the precision values for the IOP and CMP ranges would improve for the Days 1+2+3 calibration compared to the calibrations for the individual days. According to Table 5, however, this was not the case. A speculation as to why this was would be that the test field did experience some level of deformation during the multiday experiment. Nevertheless, the multisystem calibration is still a practical option as multiple data sets can be run in the same adjustment (i.e., three versus one runs in this case). Also, since all the data sets share the same object space, potentially more realistic system stability analysis can be performed.

In addition to analyzing the adjustment quantities from the two system calibration approaches, the results from the individual versus multisystem calibrations can be compared again via a system stability analysis method. More specifically, the Day 1 individual versus Day 1 multisystem, Day 2 individual versus Day 2 multisystem, and Day 3

TABLE 6: Check for image and object space compatibility between the individual versus multisystem calibration approaches.

Cam pairs/total RMSEs	Image space RMSE [px]			Object space RMSE [mm]		
	Day 1	Day 2	Day 3	Day 1	Day 2	Day 3
Day number						
Cams 1 & 2	0.37	0.25	0.37	0.14–0.16	0.09–0.11	0.14–0.16
Cams 2 & 3	0.19	0.14	0.31	0.08–0.09	0.06–0.07	0.13–0.15
Cams 3 & 4	0.27	0.19	0.33	0.11–0.13	0.08–0.09	0.14–0.16
Cams 4 & 5	0.58	0.26	0.89	0.24–0.29	0.11–0.13	0.37–0.44
Cams 5 & 6	0.28	0.24	0.53	0.12–0.14	0.10–0.12	0.22–0.26
Cams 6 & 7	0.30	0.13	0.31	0.12–0.15	0.05–0.06	0.13–0.15
Cams 7 & 8	0.22	0.18	0.54	0.08–0.10	0.07–0.08	0.20–0.24

individual versus Day 3 multisystem comparisons are listed in Table 6. The total image space RMSE value for any camera pair was under one pixel, while the total object space RMSE ranged from 0.05 mm to 0.44 mm. Again, the compatibility between the two types of system calibration solutions was considered acceptable for the work in this research initiative.

## 5. Conclusions

A mathematical model for a more straightforward-to-implement multicamera system calibration was described in this paper. The model was based on the use of a single or multiple reference camera(s) and built-in relative orientation constraints where the IOPs and the CMPs for all the cameras were explicitly estimated simultaneously in a single step. The complexity of the system calibration model is not affected by the number of cameras in the system or the number of observation epochs. The implemented adjustment for this model was able to handle an individual camera calibration, a calibration of multiple nonconstrained cameras, an individual multicamera system calibration, and a multicamera multisystem calibration. Moreover, an in situ multicamera system calibration routine was carried out, where a newly designed portable calibration test field was used. A notable feature of the test field was that it had a piece cut out of it in order for it to fit around otherwise obstructing objects. This routine fulfilled the requirements for a successful calibration, that is, it ensured that a suitable network geometry is present and that the usable image format for all cameras involved in the system had an even distribution of targets. It also proved that the multicamera system or multicamera multisystem calibration yielded the most practical and the strongest adjustment solutions due to the increased number of observations, the reduced number of unknowns, and the improved network geometry.

While the precision achieved by the system was considered sufficient for the 3D reconstruction in this research work, there are a few aspects of the calibration that can be improved.

- (i) Explore different AP models in order to further optimize the precision of the results.
- (ii) Investigate if the image coordinate measurement precision differs between cameras or images.
- (iii) Experiment with modifying the test field in order to turn it from a 2D one to a 3D one so as to reduce the level of projective compensation or high correlations between and within the interior and exterior orientation parameters and also to make it less prone to any deformations.
- (iv) Perform RMSE analysis for the 3D object space coordinates of the target points or distances between them; the ground truth values should be provided by a coordinate measuring machine or laser interferometry, that is, techniques which can achieve accuracy at the micron level.
- (v) Test if the calibration procedure can be run on a divergent multicamera system; this is important as some multicamera systems are being set up with divergent geometry due to preferences of coverage over network strength.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

## Acknowledgments

The authors would like to thank Jeremy Steward and Dr. Hervé Lahamy for their help during the data collection and Dr. Zahra Lari for her development of the coded targets. In addition, the authors are grateful to Dr. Mamdouh El-Badry, his students, and the civil engineering support staff for the opportunity to work in the structures laboratory at the University of Calgary. Portions of this paper come from the doctoral dissertation of the first author [37].

## References

- [1] F. Remondino, “3-D reconstruction of static human body shape from image sequence,” *Computer Vision and Image Understanding*, vol. 93, no. 1, pp. 65–85, 2004.
- [2] N. D’Apuzzo, “Surface measurement and tracking of human body parts from multi-image video sequences,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 56, no. 5–6, pp. 360–375, 2002.

- [3] C. Ellum and N. El-Sheimy, "Land-based mobile mapping systems," *Photogrammetric Engineering & Remote Sensing*, vol. 68, no. 1, pp. 13–28, 2002.
- [4] J. Y. Rau, A. F. Habib, A. P. Kersting et al., "Direct sensor orientation of a land-based mobile mapping system," *Sensors*, vol. 11, no. 12, pp. 7243–7261, 2011.
- [5] F. Remondino, S. F. El-Hakim, A. Gruen, and L. Zhang, "Turning images into 3-D models," *IEEE Signal Processing Magazine*, vol. 25, no. 4, pp. 55–65, 2008.
- [6] F. Remondino and S. El-Hakim, "Image-based 3d modelling: a review," *The Photogrammetric Record*, vol. 21, no. 115, pp. 269–291, 2006.
- [7] I. Datchev, A. Habib, and Y. C. Chang, "Image matching and surface registration for 3d reconstruction of a scoliotic torso," *Geomatica*, vol. 65, no. 2, pp. 175–187, 2011.
- [8] I. Datchev, A. Habib, and M. El-Badry, "Dynamic beam deformation measurements with off-the-shelf digital cameras," *Journal of Applied Geodesy*, vol. 7, no. 3, pp. 147–157, 2013.
- [9] E. Kwak, I. Datchev, A. Habib, M. El-Badry, and C. Hughes, "Precise photogrammetric reconstruction using model-based image fitting for 3d beam deformation monitoring," *Journal of Surveying Engineering*, vol. 139, no. 3, pp. 143–155, 2013.
- [10] A. M. Tommaselli, M. Galo, M. V. De Moraes, J. Marcato, C. R. Caldeira, and R. F. Lopes, "Generating virtual images from oblique frames," *Remote Sensing*, vol. 5, no. 4, pp. 1875–1893, 2013.
- [11] C. S. Fraser, "Digital camera self-calibration," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 52, no. 4, pp. 149–159, 1997.
- [12] A. F. Habib and M. F. Morgan, "Automatic calibration of low-cost digital cameras," *Optical Engineering*, vol. 42, no. 4, pp. 948–955, 2003.
- [13] D. C. Brown, "Decentering distortion of lenses," *Photogrammetric Engineering & Remote Sensing*, vol. 32, no. 3, pp. 444–462, 1966.
- [14] D. C. Brown, "Close-range camera calibration," *Photogrammetric Engineering & Remote Sensing*, vol. 37, no. 8, pp. 855–866, 1971.
- [15] J. F. Kenefick, M. S. Gyer, and B. F. Harp, "Analytical self-calibration," *Photogrammetric Engineering & Remote Sensing*, vol. 38, no. 11, pp. 1117–1126, 1972.
- [16] S. I. Granshaw, "Bundle adjustment methods in engineering photogrammetry," *The Photogrammetric Record*, vol. 10, no. 56, pp. 181–207, 1980.
- [17] T. A. Clarke and J. G. Fryer, "The development of camera calibration methods and models," *The Photogrammetric Record*, vol. 16, no. 91, pp. 51–66, 1998.
- [18] F. Remondino and C. Fraser, "Digital Camera Calibration Methods: Considerations and Comparisons," in *ISPRS Archives*, vol. XXXVI-5, pp. 266–272, Dresden, Germany, 2006.
- [19] J. H. Chandler, J. G. Fryer, and A. Jack, "Metric capabilities of low-cost digital cameras for close range surface measurement," *The Photogrammetric Record*, vol. 20, no. 109, pp. 12–26, 2005.
- [20] C. S. Fraser, "Automatic camera calibration in close range photogrammetry," *Photogrammetric Engineering & Remote Sensing*, vol. 79, no. 4, pp. 381–388, 2013.
- [21] I. Datchev, M. Mazaheri, S. Rondeel, and A. Habib, "Calibration of Multi-Camera Photogrammetric Systems," in *ISPRS Archives*, vol. XL-1, pp. 101–108, Denver, Colorado, 2014.
- [22] E. Harvey and M. Shortis, "A system for stereo-video measurement of sub-tidal organisms," *Marine Technology Society Journal*, vol. 29, no. 4, pp. 10–22, 1996.
- [23] E. S. Harvey and M. R. Shortis, "Calibration stability of an underwater stereo-video system: implications for measurement accuracy and precision," *Marine Technology Society Journal*, vol. 32, no. 2, pp. 3–17, 1998.
- [24] M. R. Shortis and E. S. Harvey, "Design and calibration of an underwater stereo-video system for the monitoring of marine fauna populations," in *ISPRS Archives*, vol. XXXII-5, pp. 792–799, Hakodate, Japan, 1998.
- [25] M. R. Shortis, S. Miller, E. S. Harvey, and S. Robson, "An analysis of the calibration stability and measurement accuracy of an underwater stereo-video system used for shell surveys," *Geomatics Research Australasia*, vol. 73, pp. 1–24, 2000.
- [26] A. Habib, I. Datchev, and E. Kwak, "Stability analysis for a multi-camera photogrammetric system," *Sensors*, vol. 14, no. 8, pp. 15084–15112, 2014.
- [27] R. Hartley, J. Trunpf, Y. Dai, and H. Li, "Rotation averaging," *International Journal of Computer Vision*, vol. 103, no. 3, pp. 267–305, 2013.
- [28] G. He, K. Novak, and W. Feng, "Stereo camera system calibration with relative orientation constraints," in *SPIE Videometrics*, vol. 1820, pp. 2–8, 1993.
- [29] B. King, "Methods for the photogrammetric adjustment of bundles of constrained stereopairs," in *ISPRS Archives*, vol. XXX, pp. 473–480, 1994.
- [30] B. King, "Bundle adjustment of constrained stereo pairs - mathematical models," *Geomatics Research Australasia*, vol. 63, pp. 67–91, 1995.
- [31] J. L. Lerma, S. Navarro, M. Cabrelles, and A. E. Seguí, "Camera calibration with baseline distance constraints," *The Photogrammetric Record*, vol. 25, no. 130, pp. 140–158, 2010.
- [32] D. D. Lichti, G. B. Sharma, G. Kuntze, B. Mund, J. E. Beveridge, and J. L. Ronsky, "Rigorous geometric self-calibrating bundle adjustment for a dual fluoroscopic imaging system," *IEEE Transactions on Medical Imaging*, vol. 34, no. 2, pp. 589–598, 2015.
- [33] S. Zheng, R. Huang, B. Guo, and K. Hu, "Stereo-camera calibration with restrictive constraints," *Cehui Xuebao/Acta Geodaetica et Cartographica Sinica*, vol. 41, no. 6, pp. 877–885, 2012.
- [34] Canon Inc, *EOS Rebel XS/EOS 1000d Instruction Manual*, 2008.
- [35] OpenCV.org, "Feature detection—OpenCV 2.4.13.0 documentation," 2014, [http://docs.opencv.org/2.4/modules/imgproc/doc/feature\\_detection.html](http://docs.opencv.org/2.4/modules/imgproc/doc/feature_detection.html).
- [36] C. S. Fraser, M. R. Shortis, and G. Ganci, "Multisensor system self-calibration," in *SPIE Videometrics IV*, vol. 2598, pp. 2–18, 1995.
- [37] I. Datchev, "Image-based Fine-scale Infrastructure Monitoring," PhD dissertation, UCGE report #20474, Department of Geomatics Engineering, University of Calgary, Calgary, Alberta, Canada, 2016.

## Research Article

# An Analysis of Multiple Criteria and Setups for Bluetooth Smartphone-Based Indoor Localization Mechanism

**Manuel Castillo-Cara,<sup>1</sup> Jesús Lovón-Melgarejo,<sup>1</sup> Gusseppe Bravo-Rocca,<sup>1</sup> Luis Orozco-Barbosa,<sup>2</sup> and Ismael García-Varea<sup>2</sup>**

<sup>1</sup>Center of Information and Communication Technologies, Universidad Nacional de Ingeniería, Lima, Peru

<sup>2</sup>Albacete Research Institute of Informatics, Universidad de Castilla-La Mancha, Albacete, Spain

Correspondence should be addressed to Manuel Castillo-Cara; [mcastillo@uni.edu.pe](mailto:mcastillo@uni.edu.pe)

Received 2 June 2017; Revised 20 August 2017; Accepted 17 September 2017; Published 23 October 2017

Academic Editor: Jacky C. K. Chow

Copyright © 2017 Manuel Castillo-Cara et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Bluetooth Low Energy (BLE) 4.0 beacons will play a major role in the deployment of energy-efficient indoor localization mechanisms. Since BLE4.0 is highly sensitive to fast fading impairments, numerous ongoing studies are currently exploring the use of supervised learning algorithm as an alternative approach to exploit the information provided by the indoor radio maps. Despite the large number of results reported in the literature, there are still many open issues on the performance evaluation of such approach. In this paper, we start by identifying, in a simple setup, the main system parameters to be taken into account on the design of BLE4.0 beacons-based indoor localization mechanisms. In order to shed some light on the evaluation process using supervised learning algorithm, we carry out an in-depth experimental evaluation in terms of the mean localization error, local prediction accuracy, and global prediction accuracy. Based on our results, we argue that, in order to fully assess the capabilities of supervised learning algorithms, it is necessary to include all the three metrics.

## 1. Introduction

A large number of proposals have been reported in the literature aiming to develop accurate indoor localization mechanisms. Most recent studies are being developed using the Received Signal Strength Indication (RSSI) of various reference wireless transmitters as a mean of estimating the position of a smartphone device. Among the technologies being considered, Wi-Fi networks have attracted the attention of many researchers and practitioners over the last years. Many experimental studies have been conducted to construct radio maps and models enabling the estimation of the distance between a reference transmitter and a smartphone device. Due to the characteristics of the wireless signal, the use of Kalman filters [1, 2], among others, have been required to remove the noise. Novel Bluetooth Low Energy (BLE) devices have become a strong alternative to Wi-Fi-based indoor location mechanisms. Their lower cost, low energy consumption, and size of the Bluetooth devices are among the

most important design features involving battery-operated smartphone devices, mainly smartphones and tablets.

Several studies have been conducted aiming to develop RSSI-based localization systems [3, 4] or simply computer vision using Kalman [5] or particle filters [6]. Early studies limited the use of Bluetooth localization mechanism to determine the locations of stationary smartphone devices at a room level [7]. Moreover, recent studies have shown that BLE4.0 signals are very susceptible to fast fading impairments making it difficult to apply the RSSI/distance models commonly used in the development of Wi-Fi-based localization mechanisms [8, 9]. In [10] the authors explore various methods used in smartphone-based indoor localization with different techniques and technologies, analyzing in-depth map, trilateration, and fingerprint techniques.

Other recent studies reported in the literature have explored alternative methods. In [11], Pei et al. have proposed a hybrid method combining fingerprinting with trilateration. In [12], the same authors have explored the impact of the

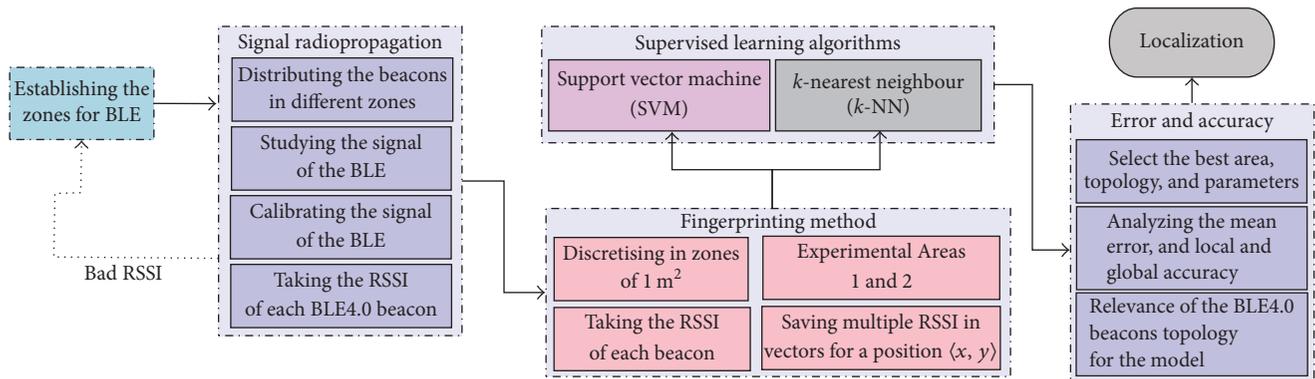


FIGURE 1: Overall schema proposal.

presence of people over the wireless signal used on the development of the localization mechanism. In [13], Guo et al. have analyzed the RSSI in different indoor environments, improving the accuracy and mean positioning error for smartphones with BLE4.0 beacons. The authors of [14] evaluate the mean localization error under various scenarios. In [15], the authors distribute efficiently the BLE4.0 beacons and make use of the information provided by additional sensors attached to the smartphone devices. Finally, Pagano et al. have proposed a system based on the ranging time of arrival, between anchor nodes and BLE4.0 beacon node [16].

In this work, in order to properly justify our proposal, we first study the signal propagation of the BLE4.0 beacon. From this first analysis, we justify the use of supervised learning algorithm as a feasible methodology to characterize the BLE4.0 beacon signal propagation to be used as a basis to develop indoor localization mechanisms. Later, we analyze the main configurable parameters based on BLE4.0 beacons and algorithms. The results obtained in a real-world scenario validate the proposal.

Figure 1 shows the overall schema proposal in this work.

The rest of the paper is organized as follows. Section 2 reviews the related work and describes the main contribution of our work. Section 3 analyzes the BLE4.0 signal propagation and justifies the use of classification algorithms on developing BLE4.0 beacons-based indoor localization mechanisms. Section 4 shows an in-depth RSSI attenuation study analyzing the impact of physical materials in our laboratory and noise introduced by other peripheral devices. Subsequently, Section 5 describes the experimental tools including the challenges to be faced when developing a BLE4.0 fingerprint-based localization mechanism. We also include a brief description of the two classification algorithms used on our proposal, experimental setups and survey campaign characteristics. Based on these preliminary results, Sections 6.1 and 6.2 present the results obtained in two different scenarios using BLE4.0 beacons as transmitter and smartphone as receiver. Moreover, we analyze the performance of the two algorithms in terms of three main metrics: (i) global accuracy; (ii) mean positioning error; and (iii) local accuracy. Section 7 briefs the results obtained in the two previous experimental areas and highlights the

main findings and lessons learned in two main areas: (i) the system configuration of a BLE4.0 beacons-based indoor localization setup and (ii) the performance evaluation of supervised learning algorithms on the development of indoor localization mechanisms. Finally, we conclude the paper in Section 8 exhibiting the final conclusions and our future work plan.

## 2. Related Work

Wireless indoor localization is a hot topic of research nowadays. Depending on the wireless sensor network technology, the use of a technique/algorithm may be more suitable or feasible with respect to others. In this section, we first overview the major trends and results. We then overview the various approaches being explored when using BLE4.0 beacons.

**2.1. Standard Wireless Positioning.** Nowadays, the main three technologies being explored to develop indoor localization mechanisms are ZigBee [17, 18], Wi-Fi, and Bluetooth. Main localization techniques for indoor localization are based on trilateration or indoor channel propagation models [9] or by means of classification algorithms [19, 20]. Main metrics used in indoor localization are global accuracy and mean positioning error [19, 21].

In [22], the authors compare the performance of Wi-Fi and BLE4.0 beacons and conclude that BLE4.0 beacons outperform Wi-Fi by 27% in terms of mean positioning error. In [19], the authors show that the positioning error of Wi-Fi is around 5–10 m and 1–2 m using BLE4.0, that is, an overall 50% improvement. Nevertheless, in [23] good results are presented for Wi-Fi fingerprint, obtaining an average error of 2.5 meters. In [6], the authors make use of RANSAC with the aim of improving the quality of the information to be used to feed a particle filter. Their main goal has been to enhance the mean positioning error and accuracy of Wi-Fi-based indoor localization mechanisms. In [24], the authors have shown that the use of particles filters or Kalman filter algorithms may not always be a good choice for BLE4.0 beacon, where the use of different classification algorithms seems to provide better results. From the above, it is clear

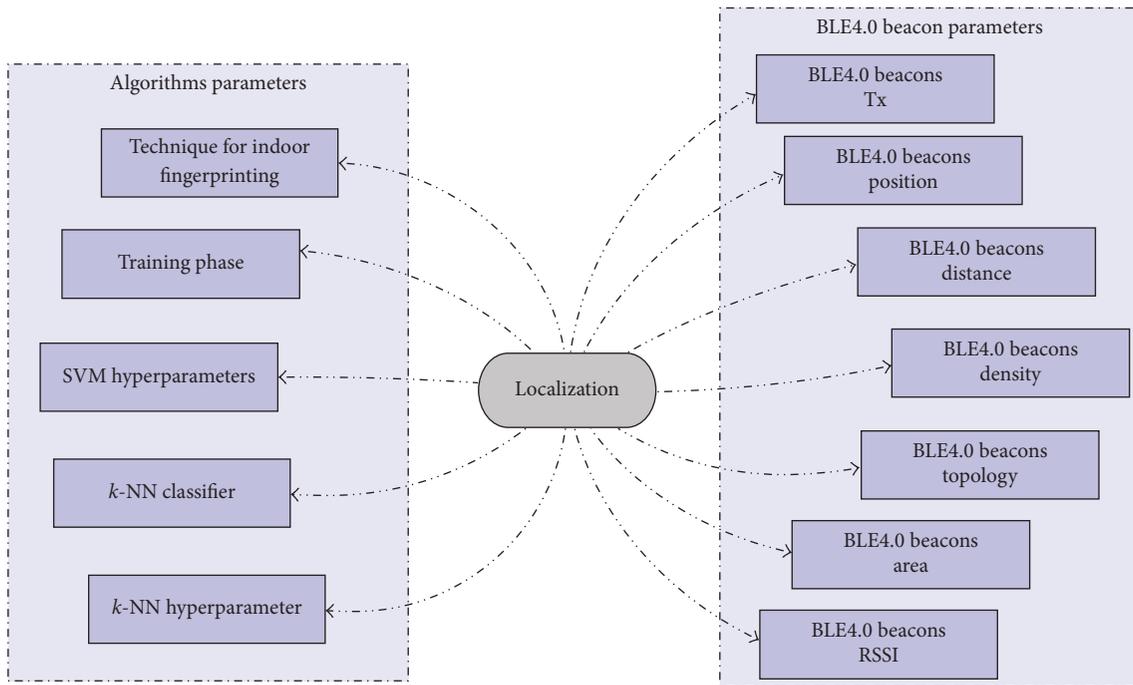


FIGURE 2: Overall parameters proposal.

that the development of accurate and robust wireless indoor localization mechanisms is still a long way to go. Besides the technological development, mainly radio systems and antennas, the use of filtering and/or classification algorithms is still one of the main research topics [25].

**2.2. BLE4.0-Based Localization Mechanisms.** Nowadays, it is widely recognized that multipath fading is one of the main challenges faced on the development of robust and accurate BLE4.0-based indoor localization mechanisms [12]. In order to overcome this challenge, the research community is actively exploring on defining the best system configuration, for example, density of BLE4.0 beacons and relative placement [26], and on identifying the most suitable data processing methodologies, that is, filtering and classification algorithms.

Some works have explored the use of regression model, separate channel fingerprints supplemented by Extended Kalman filters (EKF) [27] or particle filters [28]. In the later work, the mean positioning error of less than 4 m has been reported. Both works have shown that the physical area plays a major role on the results, not only the materials but also the dimensions.

In order to reduce the mean positioning error, different classification algorithms have been studied as new localization techniques based on fingerprinting. However, one of the main challenges is to properly tune the various parameters of the classification algorithms, since they play a major role in the achievable accuracy and mean positioning error [29, 30]. In [29], the authors have compared three different classification algorithms, Neural Networks, SVM, and  $k$ -NN.

Their results have shown that  $k$ -NN reports the best mean positioning error, approximately 4 m. In [30] better results are presented by using a combination of BLE4.0 beacon and Wi-Fi technologies and the same classification algorithms. A more in-depth analysis on the parameters of different classification algorithms is presented in [31], where the best results have been obtained using a weighted distance (WD) for  $k$ -NN. In [32], Peng et al. have obtained similar results using  $k$ -NN, testing different values for “ $k$ .”

In summary, all the abovementioned works present localization results studying different parameters, ranging from the dimensions of the area under study to the hyperparameters of the classification algorithms. In this paper, we discuss in depth the impact of two sets of parameters: (i) system configuration: the deployment and setting of the BLE4.0 beacons, namely, density and transmission power, and (ii) algorithms: the parameters governing the different classification algorithms. In this context, Figure 2 shows the overall system and algorithmic parameters studied in this paper.

### 3. BLE4.0 Signal Characterization

Recent studies have shown that BLE4.0 beacon signals are highly sensitive to interference and fast fading. Similar to Wi-Fi, BLE4.0 operates in the 2.4 GHz band divided into 40 channels, each 2 MHz wide. In order to avoid interference between BLE4.0 and Wi-Fi devices, BLE4.0 mainly uses channels 37 (2402 MHz), 38 (2426 MHz), and 39 (2480 MHz) [21]. BLE4.0 devices transmit on these channels cyclically, and they only make use of other channels when paired with other devices.

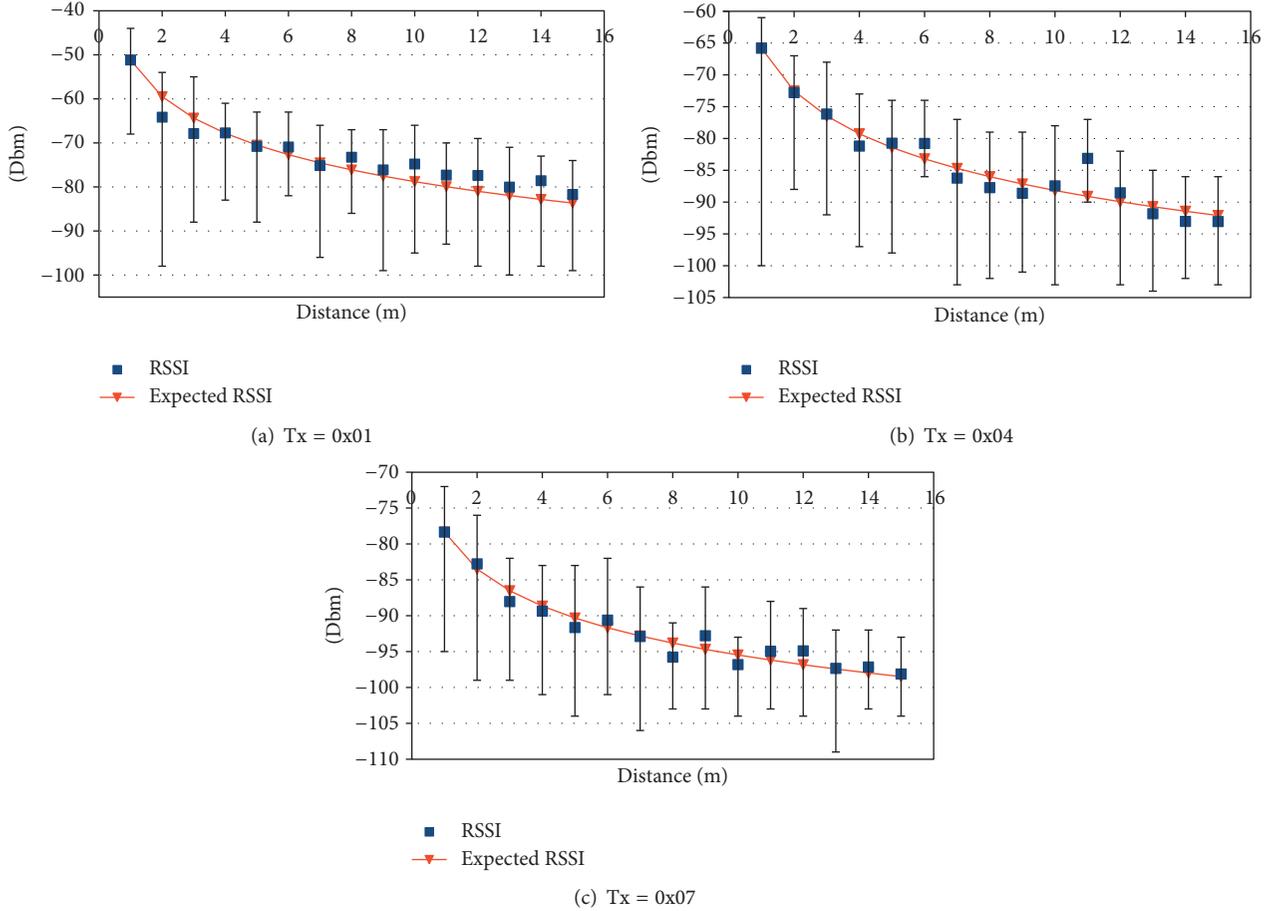


FIGURE 3: RSSI-distance correlation for three different transmission power (Tx) levels.

As for the multipath effect, it requires the development of tools and methodologies enabling the setup planning of robust and accurate BLE4.0-based indoor localization systems. In this section, we experimentally study the channel propagation of BLE4.0 signal. Our main goal is setting a baseline experimental prototype allowing us to identify the key system parameters. We then motivate the use of supervised learning algorithms as a viable methodology to enhance the accuracy and robustness of BLE4.0-based indoor localization mechanisms.

**3.1. Radiopropagation Model.** A large number of recent efforts on developing RSSI-based localization mechanisms have made use of a RSSI-distance mapping function [9] depicted in the following equation:

$$\text{RSSI} = U_L - 10 \cdot n \cdot \log\left(\frac{d}{d_0}\right), \quad (1)$$

where  $U_L$  denotes the RSSI in dB at  $d_0$  distance (typically 1 meter),  $n$  is the path loss coefficient factor, and  $d$  is the distance in meters between two wireless devices: a transmitter and a receiver.

All the parameters in (1) can be experimentally measured except  $n$ , which needs to be estimated. Classical approaches

determine the  $n$  value that minimizes the estimation error in a set of ground truth preliminary measurements [24]. This is usually done using the lowest squared error as metric.

**3.2. BLE4.0 Beacon RSSI Distance Modelling.** In order to verify the suitability of applying this approach using BLE4.0 beacons, we conducted a preliminary experimental test. We initially placed the smartphone device at 1m from the BLE4.0 beacon and, progressively, moved it away from the BLE4.0 beacon in steps of 1m up to the maximum distance of 15m. At each location, we sampled the RSSI level for a period of one minute. We made use of an Android 5.1 smartphone, from now on referred to as the receiver. We conducted three sets of independent measurements by varying the transmission power (Tx) level of the BLE4.0 beacon, namely 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, and 0x07, which, respectively, correspond to 4 dBm, 0 dBm, -4 dBm, -8 dBm, -12 dBm, -16 dBm, and -20 dBm [33]. All measurements were conducted under Line-of-Sight (LoS) conditions.

Figures 3(a), 3(b), and 3(c) show the results for Tx = 0x01, Tx = 0x04, and Tx = 0x07 transmission power levels, respectively. As seen in the figures, the RSSI decreases as a function the distance between the BLE4.0 beacons and the target receiver. We also include in the figures the results of adjusting the samples to the model given by (1).

TABLE 1: Mean squared error and standard deviation obtained for a distance of up to 15 m for all transmission power (Tx) levels.

Tx	Mean squared error (m)	Standard deviation (m)
0x01	7.81	0.68
0x02	5.99	0.60
0x03	6.66	0.70
0x04	4.31	0.40
0x05	5.75	0.59
0x06	5.55	0.57
0x07	1.53	0.13

TABLE 2: Mean squared error and standard deviation obtained for a distance of up to 8 m for all transmission power (Tx) levels.

Tx	Mean squared error (m)	Standard deviation (m)
0x01	6.71	0.58
0x02	5.25	0.52
0x03	6.01	0.65
0x04	2.49	0.23
0x05	4.10	0.40
0x06	3.49	0.39
0x07	1.44	0.14

Table 1 provides the mean squared error and standard deviation for all Tx tested. The results show the infeasibility in looking for a direct relation using the aforementioned RSSI-distance model in combination with a triangulation technique: a given RSSI value may correspond to different distance estimates [6]. Nevertheless, the results obtained establish the basis to explore alternative solutions, as well as the guidelines on setting the BLE4.0 beacons. In fact, a closer look to the results depicted in Figure 3(c) show that the RSSI-distance model may hold up to an approximate distance between 7 and 8 meters. In all cases, we notice that the RSSI drops substantially between one and three meters, but it then exhibits smaller variations in the range between three and six meters. In the case when the transmission power is set to Tx = 0x04 and Tx = 0x07, we notice a higher decrease on the RSSI in the 6 to 8 m interval than in the case when Tx = 0x01. Beyond the distance of 8 m, the RSSI levels show severe discrepancy with the RSSI/distance model.

Table 2 provides the mean squared error and standard deviation for all transmission power levels for a maximum distance of 8 m. According to these results, it is clear that the use of the lowest transmission power, Tx = 0x07, offers the best solution. This is an important result, since one of our main aims is to limit the power consumption as a means to span the lifetime of the BLE4.0 beacons. The results also show that limiting the size of the experimental area will play an important role in the localization process. For instance, in the case of Tx = 0x07, a mean RSSI value of -96 dBm may correspond to a distance of 8 m when limiting the maximum distance to 8 m, while -95 dBm, a higher RSSI, may correspond to 11 m or 12 m when we consider a maximum distance of 15 m. These results provide the basis on determining the initial deployment and power setting of the

BLE4.0 beacons. In fact, this analysis provides us the basis to configure the setup to obtain the data required to guide the supervised learning algorithms.

#### 4. Bluetooth Signal Attenuation

In this section we analyze in depth the RSSI fingerprint throughout the experimental area and its behaviour at different times of the day. Our main aim is to get an insight on the factors that may drastically affect the localization process [23, 30]. For this experiment we choose a medium transmission power level, that is, Tx = 0x04. We defined an experimental area fragmented into 15 zones, of 1 m<sup>2</sup> each, separated by a guard zone of 0.5 m<sup>2</sup> to better differentiate the RSSI of joint sectors (see Figure 4 for details). The experimental setup consists of a total area of 9.6 m by 6.3 m, where the minimal distance between a BLE4.0 beacon and the receiver will be 1.5 m.

In Figure 5, four different views of the physical area represented in Figure 4 (our laboratory) are shown, which are the perspective images taken from four different BLE4.0 beacons positions. As seen in the picture, BLE4.0 beacons “Be10” and “Be11” have been placed by the window side while BLE4.0 beacons “Be07,” “Be08,” and “Be09” have been placed by the drywall side.

*4.1. RSSI Fingerprint.* During our first survey, we monitored the RSSI of each BLE4.0 beacon at each one of the fifteen sectors of our experimental area.

Figures 6(a)–6(e) depict the RSSI average values of each of the five BLE4.0 beacons over the experimental area. Note that the RSSI reported by the BLE4.0 beacons placed close to the windows, namely, BLE4.0 beacons “Be10” and “Be11,” are characterized by a lower signal strength; see Figures 6(d) and 6(e). In the case of BLE4.0 beacon “Be11,” the signal vanishes quickly starting at neighbouring sectors.

Hence, from these figures we can extract the following conclusions:

- (i) We can easily identify the location of each BLE4.0 beacon from the RSSI fingerprint.
- (ii) The RSSI level of the beacons placed by the drywall side is higher than the one reported by the beacons located by the window side.

These results show the need to evaluate the attenuation of the BLE4.0 beacons under different conditions.

*4.2. Intraday RSSI Surveys.* In order to illustrate the challenges faced on developing RSSI-based indoor localization mechanisms, we carried out three survey campaigns. Similar to our previous survey campaign, we monitored the RSSI levels of the various BLE4.0 beacons throughout the experimental area. The campaigns were carried out at three different times throughout a day: morning, midday, and afternoon. We will refer to the sample traces as Take 1, Take 2, and Take 3, respectively. In the following, we will discuss our main findings on the analysis of the data obtained for BLE4.0 beacon “Be09.” Our choice has been based on the fact that

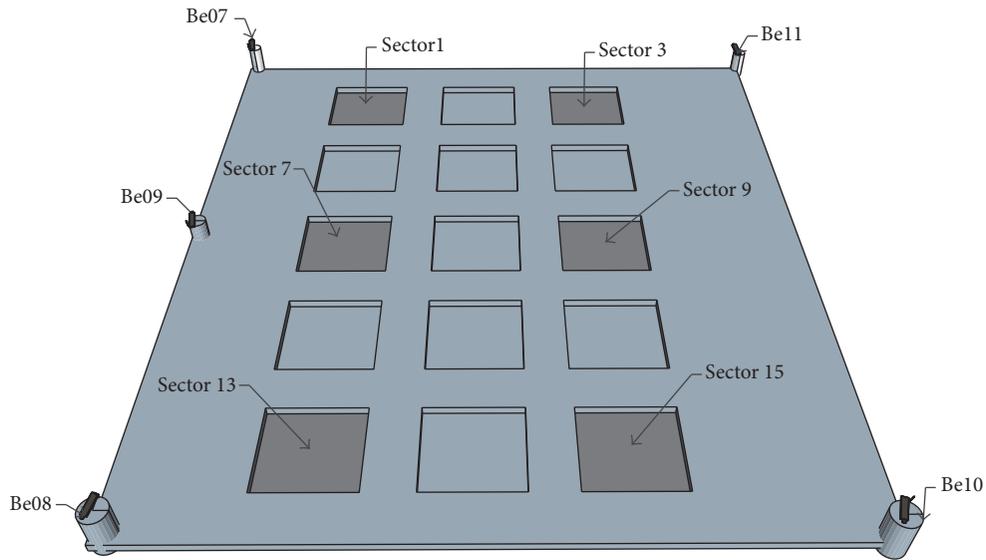


FIGURE 4: Experimental area for Bluetooth signal attenuation experiments.



(a) From BLE4.0 beacon "Be07"



(b) From BLE4.0 beacon "Be08"



(c) From BLE4.0 beacon "Be10"



(d) From BLE4.0 beacon "Be11"

FIGURE 5: Pictures, each one of each of the four corners of the laboratory.

BLE4.0 beacon "Be09" was placed at the midpoint of the drywall side. This should allow us to compare the RSSI levels at the two opposite sides of the experimental area over the same distance.

Figure 7 shows the RSSI traces for Sectors 1, 3, 9, and 15. Recall that BLE4.0 beacon "Be09" is located at the right side of Sector 7. We have also included the mean RSSI corresponding

to each one of the traces. Table 3 summarizes the main statistics of all three traces. From the analysis of the traces, we can make the following observations:

- (i) The RSSI varies substantially throughout the time. The values reported for Sector 13, located close to the corridor, exhibit the major differences between the highest and lowest RSSI value; see Table 3. This clearly

TABLE 3: Statistics for three different RSSI values (dBm) traces of BLE4.0 beacon “Be09” using Tx = 0x07.

Sector	Take 1		Take 2		Take 3	
	Mean/variance	Min/Max	Mean/variance	Min/Max	Mean/variance	Min/Max
1	-84.1/6.9	-106.0/-72.0	-84.5/7.1	-109.0/-72.0	-82.9/6.4	-105.0/-73.0
3	-90.7/6.9	-108.0/-76.0	-90.9/5.6	-108.0/-80.0	-88.6/6.8	-105.0/-74.0
7	-83.0/11.9	-110.0/-66.0	-85.0/9.2	-105.0/-67.0	-82.7/8.7	-107.0/-69.0
9	-86.8/7.3	-105.0/-73.0	-88.5/7.4	-110.0/-75.0	-82.0/9.4	-109.0/-68.0
13	-85.1/9.8	-110.0/-64.0	-88.2/10.1	-106.0/-68.0	-82.1/9.1	-110.0/-68.0
15	-84.0/7.3	-109.0/-70.0	-83.5/7.4	-101.0/-68.0	-80.4/7.4	-105.0/-70.0

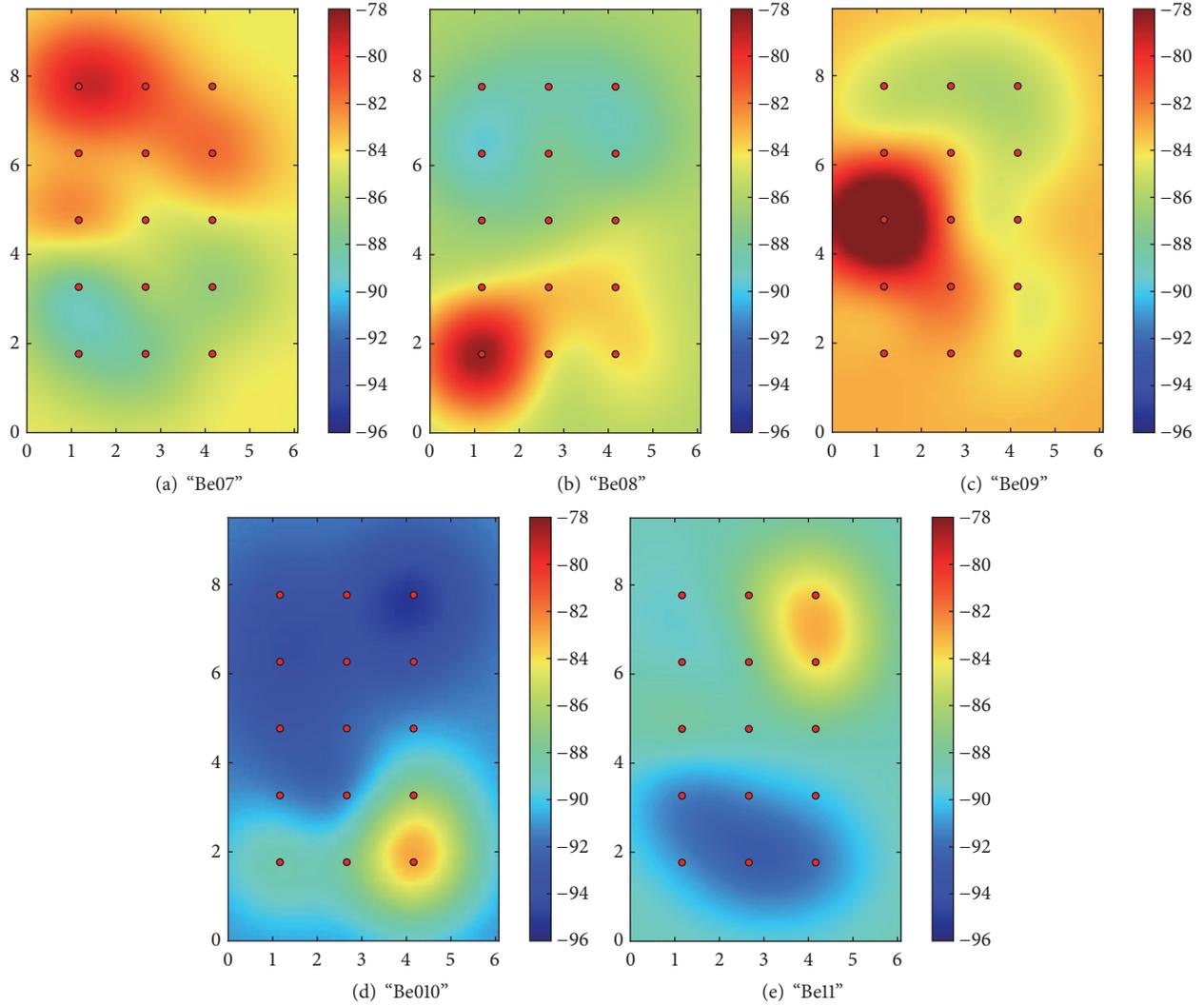


FIGURE 6: RSSI fingerprint for the BLE4.0 beacons.

shows the need of taking into account the floor plan when developing an indoor localization mechanism.

- (ii) The mean RSSI levels of sectors located at the same distance from Be09 substantially vary from one to another. For instance, the RSSI levels of Take 3 of Sectors 3 and 15 located both at the same distance from BLE4.0 beacon “Be09” exhibit a difference as

high as 8 dBm. The gap between mean of the three traces for these two sectors consistently report a high value.

- (iii) The RSSI varies substantially from one survey campaign to another in the sectors located by the windows. We notice that the RSSI level of Sector 1, the one located at the corner of two drywalls, exhibits a

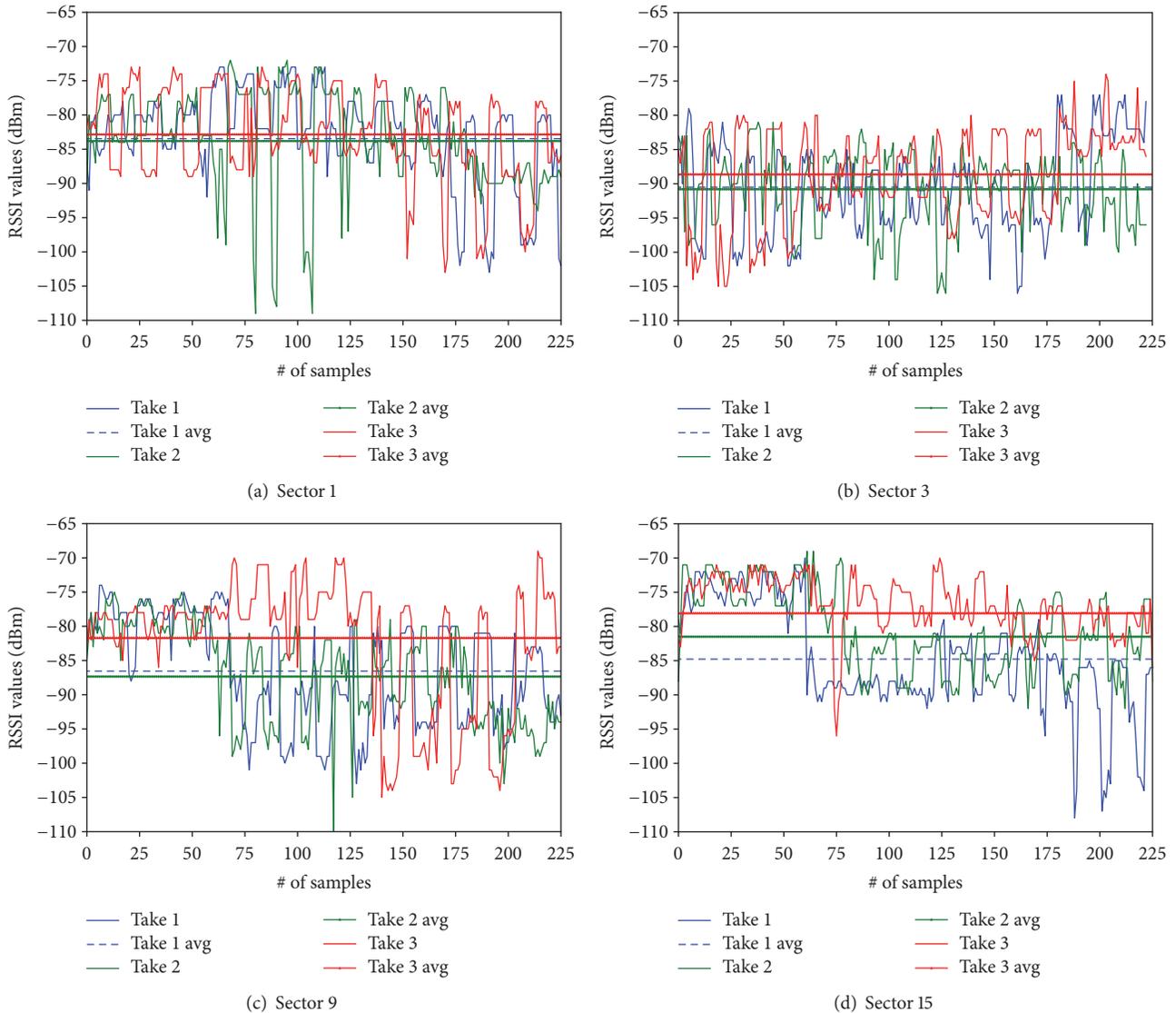


FIGURE 7: RSSI intraday traces for BLE4.0 beacon “Be09.”

more constant value. From these results, we confirm that counting with the floor plan of the experimental area is a must to be able to properly analyze the results.

According to the previous analysis and the experimentation carried out in Section 3, we can remark the following characteristics for the RSSI using BLE4.0 beacons:

- (i) Some of the levels of the RSSI for the sectors close to the window side are more than 10% lower than the ones reported for the sector located close to the drywall. This behaviour of the RSSI affects the classification process since the RSSI varies substantially.
- (ii) The RSSI from BLE4.0 beacons is very sensitive and depending on structural characteristics of the surrounding walls.
- (iii) The levels of the RSSI reported for sector located at the same distance from the reference BLE4.0 beacon,

“Be09” in our case, may substantially vary from one sector to another.

Our results show the need of exploring alternative data processing mechanisms towards the development of a RSSI-based localization solution. In order to be able to focus on the characterization of the signal in an indoor environment taking into account only the floor plan, we have restricted the access to the lab premises during our experiments.

## 5. Experimental Apparatus and Algorithms

In this section, we introduce the specifications and technical details of our experimental setting. Firstly, we describe the experimental tools developed in our research. Next, the two classification algorithms used in our experiments are explained with their configurations and metrics analyzed in the third and fourth part, respectively. Finally, we described

the physical layout of the testbed we have used to carry out all the indoor localization experiments.

*5.1. Experimental Tools.* From the previous sections analysis, we argue that the following holds:

- (i) Supervised learning algorithms are worth exploring. We therefore suggest evaluating the use of Support Vector Machine (SVM) and  $k$ -Nearest Neighbour ( $k$ -NN) algorithms.
- (ii) The actual distance between the BLE4.0 beacon transmitter and the target plays a central role in the estimated RSSI. We should therefore consider multiple experimental setups by varying the number and distance between the reference BLE4.0 beacons and the target.
- (iii) Line-of-Sight seems to be an essential requirement in order to get a first insight into the Bluetooth capabilities towards the development of indoor location fingerprints.
- (iv) *Data preprocessing.* Given that our interest is to evaluate the system configuration, beacons density, and power, we have decided not to apply any filtering or outliers detection technique. We therefore use the raw data collected during our surveys.
- (v) The transmission power (Tx) level should also be carefully considered to ensure the long run of the BLE4.0 beacons. We use Tx = 0x07 throughout our first set of experiments and Tx = 0x04 and Tx = 0x07 during the second set of experiments. These two transmission power levels offer the best characteristics for our study: lower power consumption and an almost monotonic decrease of the mean RSSI level as a function of the distance in the 0–8 m interval.

*5.2. Supervised Learning Algorithms.* In this work, we propose making use of supervised learning algorithms (SLAs) to estimate the position of the receiver. SLAs consist of two phases, a training phase, where input data should have been previously annotated with their corresponding category. This phase generates a classification model, which is subsequently used to infer the category of provided test data during the classification phase. That is to say, when applied to localization, SLA is used to generate the RSSI fingerprint from which the location can be obtained.

In this work, we explore the use of two popular SLAs, namely, the  $k$ -Nearest Neighbour ( $k$ -NN) [21, 29–31] and the Support Vector Machine (SVM) [28, 29] algorithms. A brief description of these two algorithms is included in the following:

- (i)  $k$ -NN: given a test instance, this algorithm selects the  $k$ -Nearest Neighbours, based on a predefined distance metric of the training set. In our case, we use the Euclidean distance since our predictor variables (features) share the same type, that is, the RSSI values, properly fitting the indoor localization problem [34]. Although  $k$ -NN uses the most common neighbour

of the  $k$  located categories (which is the mode of the category), some variations are used (e.g., weighted distance) to avoid removing relevant information. In this paper, we have set the hyperparameter to  $k$  with values 1, 3, and 5. We have verified that further increasing  $k$  does not improve our results. We use both mentioned versions of the algorithm: the weighted distance (WD) and mode (MD).

- (ii) SVM: given the training data, a hyperplane is defined to optimally discriminate between different categories. If a linear classifier is used, SVM constructs a line that performs an optimal discrimination. For the nonlinear classifier, kernel functions are used, which maximize the margin between categories. In this paper, we have explored the use of linear classifier and polynomial kernel with two different grades, namely, 2 and 3. Finally, we present only the best results which were obtained with a polynomial kernel with a quadratic function [34].

*5.3. Experimental Setups.* From our preliminary experimental analysis, it is clear that we should keep in mind the following aspect. If the locations of the BLE4.0 beacons change, we must carry out a new off-line/on-line sample collection campaign. The measurement campaigns mainly consisted of the following three steps.

(i) *Off-Line.* RSSI measurements collection phase: during this phase, we use the receiver for collecting a set of RSSI samples at predetermined locations spread over the experimental field covered by  $N$  BLE4.0 beacons.

(ii) *Data Storage.* The data is organized into a  $N$ th dimension vector and labelled with the coordinates of the receiver position.

(iii) *Classification.* We evaluate the performance of the two SLAs using the RSSI measurements as our source data. The evaluation will be measured in terms of the accuracy of the estimated location of the target.

*5.4. Classification Metrics.* Prior to the training phase, RSSI measurements are obtained by placing the receiver at different locations. These captures are then stored in a database during an off-line phase including the  $\langle x, y \rangle$  coordinates and RSSI level for each sample. Afterwards, the RSSI receiver measures are captured again in an on-line phase. These latter instances are then compared with the model derived in order to predict the location of the receiver, that is, generate the RSSI-based location fingerprint.

We evaluate the localization performance of the two classification algorithms in terms of the following metrics:

- (i) Global accuracy: it is the algorithm's precision in the classification phase. The value is calculated in percentage (%) between the exact positioning operations and the total number of positioning operations over the whole experimental area.

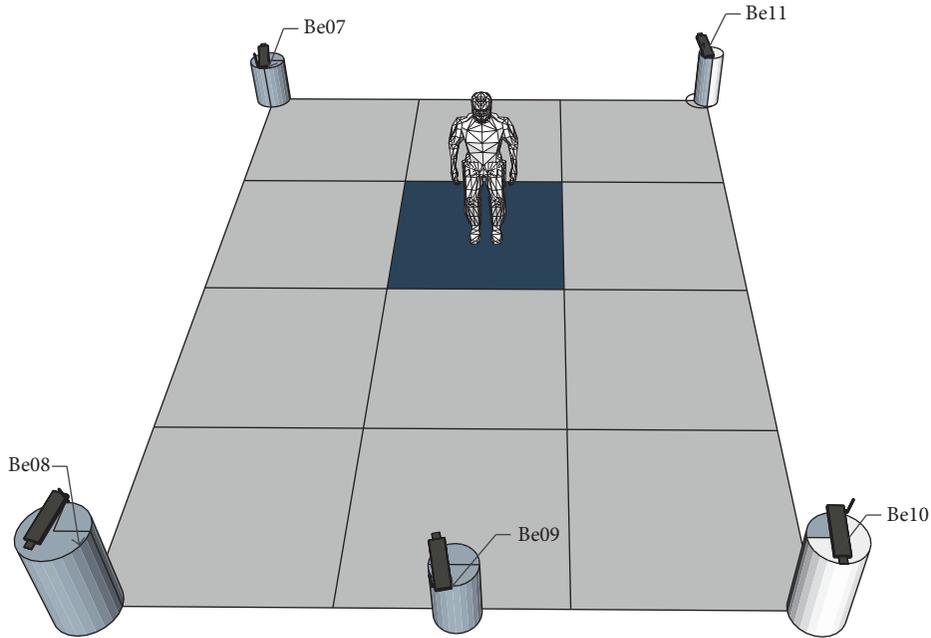


FIGURE 8: Configuration of Experimental Area 1.

- (ii) Local accuracy: it is the individual algorithm's precision in the classification phase for each sector of the experimental area. The value is calculated in percentage (%) too.
- (iii) Mean positioning error: it is the average error for the whole experimental area. This error is calculated in meters ( $m$ ) taking into account the total dimensions of each area. From now it is named as "mean error."

**5.5. Survey Campaign Characteristics.** Our experiments were conducted in a lab of our research institute. We placed four BLE4.0 beacons at each one of the four corners in a rectangular area, and we considered two experimental areas with different dimensions. A fifth BLE4.0 beacon was placed in the middle of one of the longest edges of the room.

For the experimental areas used in this paper, we carried out a survey campaign as follows:

- (i) We fixed the Tx of all BLE4.0 beacons to the same level.
- (ii) We placed the receiver at the center of each one of the sectors, and we measured the RSSI of each one of the five BLE4.0 beacons during  $x$  minutes depending on the experimental area.
- (iii) The survey was carried out through a time period of five days. The lab occupancy was limited to two people, the same that were in charge of collecting the data during the afternoon hours.

Once the initial parameters are established, in the next sections we proceed to analyze other parameters for indoor localization in the two different environments taking into account the physical area represented in Figure 5.

## 6. Performance Evaluation Results

This section analyzes the results for the classification algorithms in two different areas with different physical characteristics. This analysis has been performed taking into account the three classification metrics: (i) global accuracy; (ii) local accuracy; and (iii) mean error.

**6.1. Experimental Area 1.** In this first setup, we explore the distribution and number of BLE4.0 beacons in the experimental field. The total size of the experimental field used for this first experiment is set to an area of  $4\text{ m} \times 3\text{ m}$  subdivided into twelve sectors of  $1\text{ m}^2$ , as shown in Figure 8. Five BLE4.0 beacons, denoted by "Be07," "Be08," "Be09," "Be10," and "Be11," were placed around the area. With the main goal of identifying blind spots in the experimental field and the number of required BLE4.0 beacons, we carried out six independent trials: in the first configuration, we used five BLE4.0 beacons while, in the following five trials, we removed one BLE4.0 beacon at a time. As already mentioned, by limiting the maximum distance between the BLE4.0 beacon and the target to lower than 8 m, we avoid huge discrepancies on the distance-estimation model.

RSSI samples were collected in each of the 12 sectors during approximately five to six minutes. We evaluate and store the arithmetic mean of all the collected samples. No samples were discarded during this phase.

For each trial, the data training set consisted of 231 vectors and a validation set of 99 vectors, randomly selected for each experiment. The results show the classification metrics of the algorithm executed 50 times.

**6.1.1. Case 1:  $k$ -NN.** We use three different values of  $k$ , namely  $k = 1$ ,  $k = 3$ , and  $k = 5$ . This should allow us to identify the

TABLE 4:  $k$ -NN, global accuracy (%) for different BLE4.0 beacon setups for Tx = 0x07. Best result is shown in bold.

Configuration	$k = 1$	$k = 3$	$k = 5$
Be07, Be08, Be09, Be10, Be11	12.96	12.82	12.32
Be08, Be09, Be10, Be11	10.06	10.12	9.79
Be07, Be09, Be10, Be11	11.67	10.54	10.92
Be07, Be08, Be10, Be11	13.83	11.81	11.75
Be07, Be08, Be09, Be11	<b>14.20</b>	12.82	12.78
Be07, Be08, Be09, Be10	13.17	12.28	12.64

best value fitting our requirements. Furthermore, the analysis used global and local accuracy and mean error with two main criteria, namely, the MD of the  $k$  values, and the WD between the  $k$  values obtained.

*Global Accuracy.* Table 4 shows the global accuracy according to the different BLE4.0 beacons setups for Experimental Area 1 using the MD of the  $k$  values. We can notice the best global accuracy is obtained for  $k = 1$ , being the best configuration without BLE4.0 beacon “Be10.” Moreover, the worst setup occurs when the BLE4.0 beacon “Be07” is eliminated, being for  $k = 5$  the worst results. As seen from the results as the  $k$  value is increased, the global accuracy decreases. In fact, this case represents the worst case; it clearly shows that attempting to estimate the position of the target using neighbouring values without taking into account their relevance has a negative impact on the results.

*Mean Error.* Table 5 shows the mean error for different setups. As seen from the table, increasing the number of neighbours  $k$  has a negative impact on the mean error when using the MD modality. On the contrary, the mean error is reduced by approximately 20% in most cases when increasing  $k$  from 1 to 5 in the WD modality. In this case, the setup with BLE4.0 beacons at the corners (i.e., without BLE4.0 beacon “Be09”) gives us better results for  $k = 5$ . The worst results are obtained when we remove one of the two BLE4.0 beacons, “Be07” or “Be08,” placed by the drywall side of the experimental area.

Figure 9(a) shows the positioning error heatmap for the best global accuracy and Figures 9(b) and 9(c) the best positioning error using MD and WD, respectively.

From the results shown in tables and heatmaps, we can conclude the following:

- (i) A closer look at the heatmaps reveals very good results, an estimation positioning error as low as 0.4 m.
- (ii) Table 5 shows that the lowest mean errors are obtained using only four BLE4.0 beacons placed at the corners.
- (iii) Table 5 shows that increasing  $k$  from 1 to 5 has a positive impact when using the WD modality of the  $k$ -NN algorithm but a negative impact when the MD modality is preferred. This shows the importance of weighting the information according to its relevance.

(iv) Figure 9(c) reveals that the use of the WD modality of the  $k$ -NN considerably improves the accuracy at the center of the experimental area: fusing the WD of the BLE4.0 beacons proves effective.

(v) The heatmaps reveal a higher mean error in the sectors close to the BLE4.0 beacons, being considerably lower in the case of “Be10.” This latter BLE4.0 beacon has exhibited the lowest RSSI level among all BLE4.0 beacons; see Figure 6(d). This may translate into the estimation of a slight change on the distance as the signal decreases.

*Local Accuracy.* In this section, we evaluate the local accuracy for the best global accuracy and the lowest mean error cases; see Figures 10(a), 10(b), and 10(c), respectively. We are mainly interested in defining the guidelines to configure the localization setup according to the user needs.

Figure 10(c), for  $k = 5$ , shows that the sectors close to “Be07” and “Be11” estimate a mean error of around 2.25 m (refer to Figure 9(c)), with an accuracy of 40% while the accuracy at the center experimental area, corresponding to the lowest mean error, is approximately 6%. We also notice that the lowest accuracy is reported in the sector close to BLE4.0 beacon “Be10.” Figure 10(a) shows that the accuracy over the whole experimental area with respect to the accuracy of the other two figures can be improved by discarding BLE4.0 beacon “Be10.” However, we notice that the mean error in sectors close to BLE4.0 beacons “Be07” and “Be11” gets severely affected. Figure 10(b) shows similar results to the case when BLE4.0 beacon “Be10” is removed. These results clearly show that the placement of four BLE4.0 beacons at the corners of the experimental area provides a more uniform localization accuracy over the whole area. However, from Figure 10(c) it is also clear that it is important to consider the relevance of the information provided by the BLE4.0 beacons, that is, the difference on the initial RSSI levels of each BLE4.0 beacon.

*6.1.2. Case 2: SVM.* For the SVM classifier, we use three different kernels: linear, polynomial of degree 2 ( $d = 2$ ), and polynomial of degree 3 ( $d = 3$ ). As in the  $k$ -NN case, the analysis has been developed for the same experimental area and the same number of training and validation samples.

*Global Accuracy.* Global accuracy values for SVM are presented in Table 6. In all system configurations, but the one not making use of “Be07,” the table shows that a polynomial kernel of degree  $d = 3$  provides better results, that is, linear and  $d = 2$ . When comparing one to one the results for each BLE4.0 beacon system configuration, we find that SVM (with  $d = 2$ ) reports a global accuracy approximately between 1% and 4%, depending on the BLE4.0 beacons setup, lower than the one obtained using  $k$ -NN with  $k = 1$ .

*Mean Error.* Table 7 shows the mean error for different BLE4.0 beacon setups. As can be observed all mean error values are very similar, but the lowest mean error is provided for the SVM with a polynomial kernel with  $d = 2$ . Furthermore, we note that all values are higher than the ones reported by the  $k$ -NN algorithm using WD modality.

TABLE 5:  $k$ -NN, mean error (m), using statistical mode (MD) and weighted distance (WD), for different BLE4.0 beacon setups for Tx = 0x07. Best results are shown in bold.

Configuration	MD (m)			WD (m)		
	$k = 1$	$k = 3$	$k = 5$	$k = 1$	$k = 3$	$k = 5$
Be07, Be08, Be09, Be10, Be11	1.52	1.74	1.69	1.52	1.34	1.29
Be08, Be09, Be10, Be11	1.64	1.73	1.75	1.64	1.40	1.35
Be07, Be09, Be10, Be11	1.56	1.71	1.68	1.56	1.36	1.30
Be07, Be08, Be10, Be11	1.52	1.66	1.61	1.52	1.33	<b>1.27</b>
Be07, Be08, Be09, Be11	1.54	1.64	1.63	1.54	1.32	1.28
Be07, Be08, Be09, Be10	<b>1.51</b>	1.72	1.68	1.51	1.32	1.28

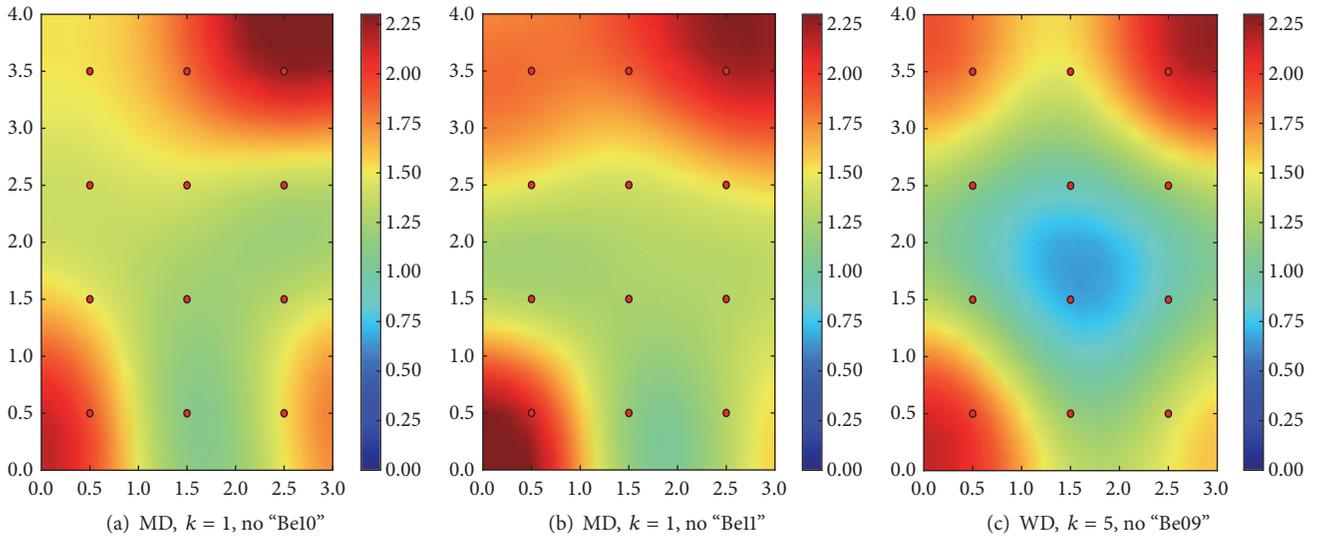


FIGURE 9:  $k$ -NN, positioning error heatmaps, using statistical mode (MD) and weighted distance (WD), for different BLE4.0 beacon setups for Tx = 0x07.

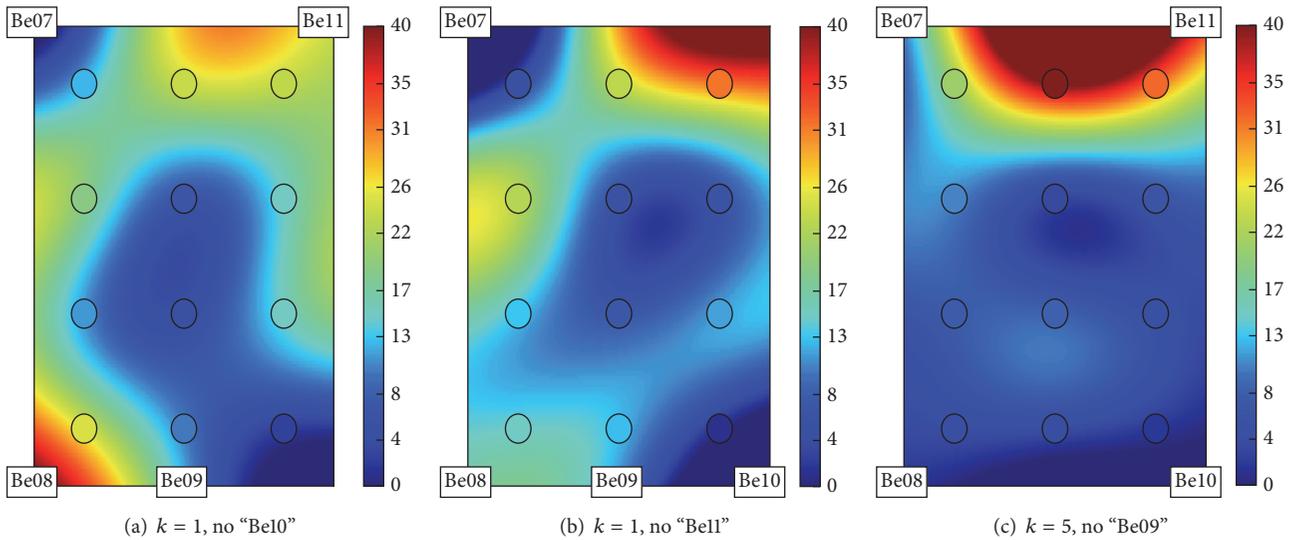


FIGURE 10:  $k$ -NN, local accuracy heatmaps for different BLE4.0 beacon setups for Tx = 0x07.

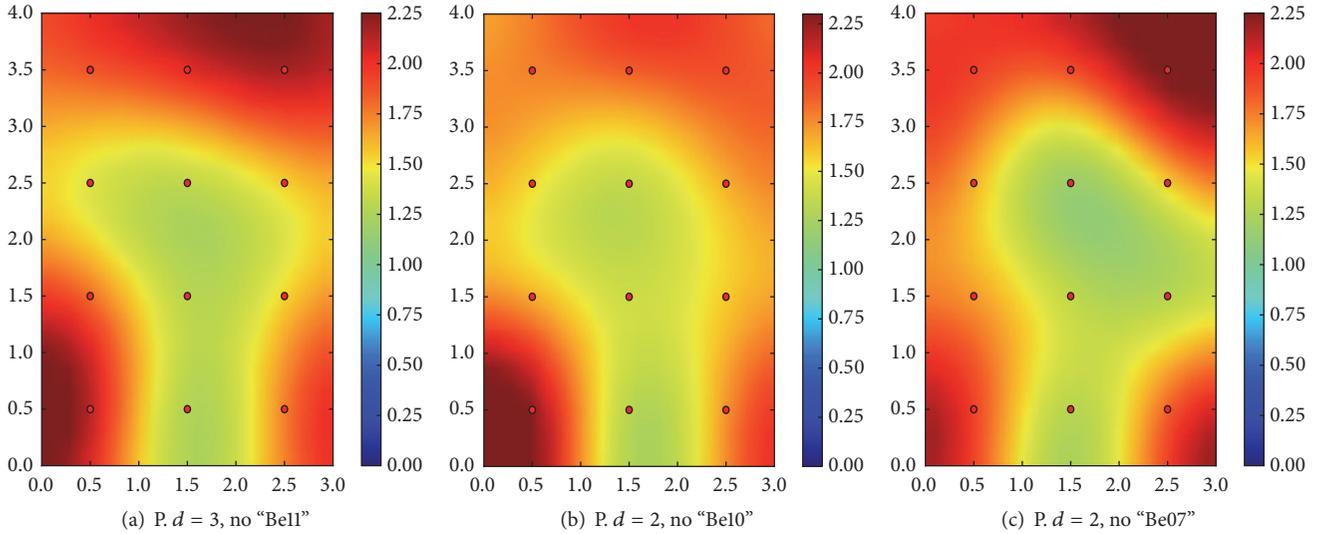


FIGURE 11: SVM, positioning error heatmaps for different BLE4.0 beacon setups for Tx = 0x07.

TABLE 6: SVM, global accuracy (%) for different BLE4.0 beacon setups for Tx = 0x07. Best result is shown in bold.

Configuration	Linear	Pol. $d = 2$	Pol. $d = 3$
Be07, Be08, Be09, Be10, Be11	10.00	9.29	11.21
Be08, Be09, Be10, Be11	8.38	10.60	10.70
Be07, Be09, Be10, Be11	10.80	9.39	9.49
Be07, Be08, Be10, Be11	9.29	9.79	10.51
Be07, Be08, Be09, Be11	9.70	10.50	10.71
Be07, Be08, Be09, Be10	9.70	10.60	<b>12.32</b>

TABLE 7: SVM, mean error (m) for different BLE4.0 beacon setups for Tx = 0x07. Best results are shown in bold.

Configuration	Linear	Pol. $d = 2$	Pol. $d = 3$
Be07, Be08, Be09, Be10, Be11	1.65	1.64	1.65
Be08, Be09, Be10, Be11	1.64	<b>1.60</b>	1.65
Be07, Be09, Be10, Be11	1.63	1.64	1.62
Be07, Be08, Be10, Be11	1.65	1.62	1.62
Be07, Be08, Be09, Be11	1.63	<b>1.60</b>	1.61
Be07, Be08, Be09, Be10	1.64	1.66	1.64

Figure 11(a) shows the positioning error heatmaps for the best global accuracy and Figures 11(b) and 11(c) for the lowest mean error. Similar to the results reported by the  $k$ -NN algorithm, sectors close to the BLE4.0 beacons exhibit higher mean error. However, in contrast to the mean error heatmaps for  $k$ -NN (see Figure 9), the heatmaps for SVM are more uniform throughout the central sector of the experimental area. Moreover, the sector around BLE4.0 beacon “Be10” exhibits worse results than the other ones obtained using the  $k$ -NN algorithm. This clearly shows the benefits of taking into account the RSSI levels reported by the various BLE4.0 beacons. These RSSI levels do not exclusively depend on

the distance but also on the structural characteristics of the environment.

*Local Accuracy.* Figure 12 shows the local accuracy corresponding to the highest global accuracy and the lowest mean errors using SVM. Figure 12(a) shows similar behaviour to the results depicted in Figure 10(b): the sectors close to BLE4.0 beacons “Be07,” “Be08,” and “Be11” show better local accuracy results. However, the local accuracy for the sectors close to the BLE4.0 beacons is considerably lower than the one obtained when using the  $k$ -NN algorithm. This effect also causes a lower global accuracy. Figures 12(b) and 12(c) show similar behaviours, that is, lower local accuracies with respect to the ones reported by the  $k$ -NN algorithm.

From the results obtained with both algorithms ( $K$ -NN and SVM), we can conclude that, in order to assess the capabilities of BLE4.0-based wireless indoor mechanisms, it is essential to count with all the three metrics: mean error, global accuracy, and local accuracy. Up to date, most studies limit their evaluation to report on the mean error and global accuracy. We should argue that, by providing the local accuracy, together with the system parameters, transmission power, and actual placement of the BLE4.0 beacons, the system designer should be able to identify the shortcomings to overcome.

In the following, we will carry out a second set of experiments. Our main aim is twofold:

- (1) We aim to explore the system localization parameters. In this case, we will consider a slightly larger experimental area and the use of a medium transmission power level, Tx = 0x04.
- (2) We aim to use larger training and validation datasets with respect to the one used in the previous study.

*6.2. Experimental Area 2.* In this second experimental setup, we further explore the performance of the  $k$ -NN and SVM

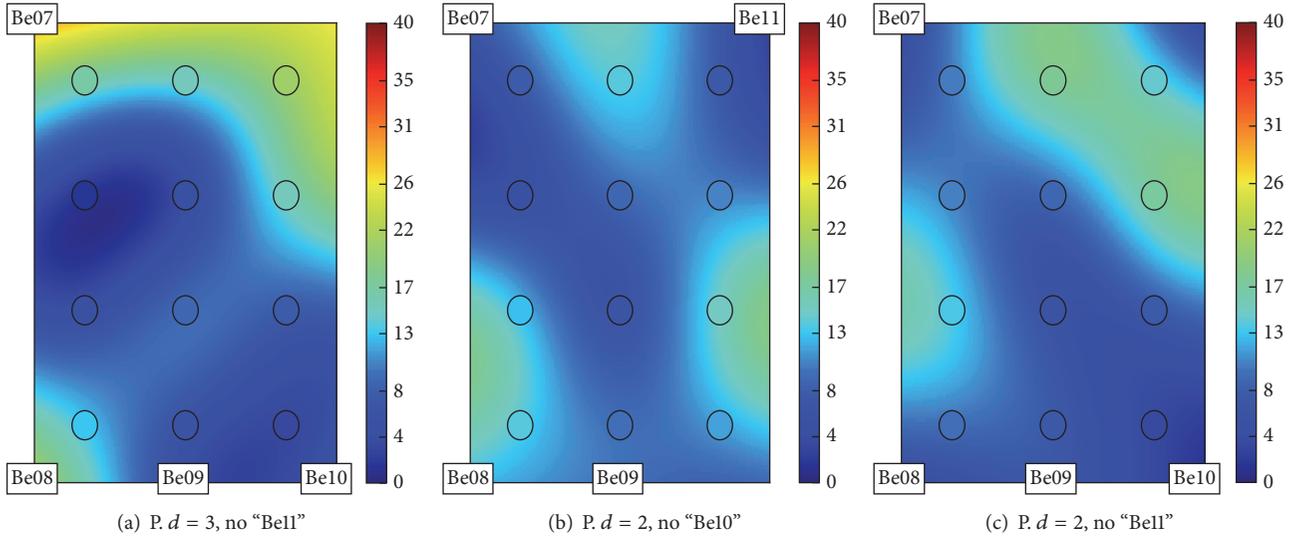


FIGURE 12: SVM, local accuracy heatmaps for different BLE4.0 beacon setups for  $T_x = 0x07$ .

TABLE 8: Average RSSI and training and validation data for two transmission powers ( $T_x$ ) level.

$T_x$	RSSI 1 m (dBm)	# training data	# validation data
0x04	-57	608	261
0x07	-75	757	290

algorithms using two different transmission power settings, namely,  $T_x = 0x04$  and  $T_x = 0x07$ . We define an experimental area fragmented into 15 zones of  $1\text{m}^2$  each separated by a guard zone of  $0.5\text{m}^2$  to better differentiate the RSSI of joint sectors. The experimental setup consists of a total area of 9.6 m by 6.3 m, where the minimal distance between a BLE4.0 beacon and the receiver is 1.5 m; see Figure 13.

Similar to the previous experimental trials, we sampled the RSSI during two minutes at the center of each of the fifteen zones. Table 8 shows the RSSI and the size of the training and validation dataset used for two transmission power levels under study.

**6.2.1. Case 1:  $k$ -NN.** Similar to the previous study, we proceeded to analyze  $k$ -NN with the same classification metrics for  $T_x = 0x04$  and  $T_x = 0x07$ .

**Global Accuracy.** Table 9 shows different BLE4.0 beacon setups used in our environment, where the best configuration is obtained eliminating the BLE4.0 beacon “Be09” for  $T_x = 0x04$ . Furthermore, for both transmission power levels we can see that the BLE4.0 beacons placed at the corners are essential in this experimental area. Moreover, a higher value of  $k$  improves the global accuracy considerably.

Therefore, the use of a larger area with guard zones enables an improvement on the global accuracy: the classification algorithm is able to better differentiate the RSSI between the different sectors. Table 4 shows an improvement

as high as 7% with respect to the results obtained for the Experimental Area I setup. Also, as expected, an intermediate transmission power level and a higher value of  $k$  improve the global accuracy. Furthermore, our results also show that the BLE4.0 beacons should be placed at the corners of the experimental area setup.

**Mean Positioning Error.** Table 10 depicts the mean error for all system configurations for the  $k$ -NN algorithm. As can be observed, when MD is used, increasing the value of  $k$  does not always improve the mean error as in the previous case. When WD is used, similar to the previous experiment, the mean error is considerably reduced. Also, in this latter case, the configuration with the four BLE4.0 beacons located at the corners and the one using five BLE4.0 beacons, offers very similar results. From this table, we also notice that better results are obtained when a higher transmission power level is used, that is, for  $T_x = 0x04$ . Then, better mean error results are obtained at the expense of using a higher transmission power level.

Figure 14 depicts different error heatmaps. Figures 14(a) and 14(b) show the heatmaps for the best results when setting  $T_x = 0x04$  using MD and WD, respectively. The fact of using WD proves effective in reducing the mean error across all the experimental area. On the contrary, the heatmaps produced when using the MD modality show that the classification algorithm is unable to take into account the difference on the RSSI levels of the different BLE4.0 beacons. The sectors close to BLE4.0 beacon “Be07” are characterized by the largest mean error. In other words, since the RSSI value of BLE4.0 beacon “Be07” is higher than the ones characterizing the other BLE4.0 beacons, the estimated distance is longer than expected. This effect is worsened by the inclusion of BLE4.0 beacon “Be09”; see Figure 14(c). Also, when MD is used, the classification splits the area into two main sectors; see Figures 14(a) and 14(c). Finally, the use of higher transmission power,  $T_x = 0x07$ , results in a more uniform positioning

TABLE 9:  $k$ -NN, global accuracy (%) for different BLE4.0 beacon setups for Tx = 0x04 and Tx = 0x07. Best results are shown in bold.

Tx	Configuration	$k = 1$	$k = 3$	$k = 5$
0x04	Be07, Be08, Be09, Be10, Be11	17.38	19.45	20.02
	Be08, Be09, Be10, Be11	14.15	17.26	18.64
	Be07, Be09, Be10, Be11	15.54	17.84	18.76
	Be07, Be08, Be10, Be11	16.92	18.07	<b>21.29</b>
	Be07, Be08, Be09, Be11	16.57	16.46	17.84
	Be07, Be08, Be09, Be10	15.42	14.96	17.72
0x07	Be07, Be08, Be09, Be10, Be11	15.21	15.44	<b>16.11</b>
	Be08, Be09, Be10, Be11	13.87	14.09	15.32
	Be07, Be09, Be10, Be11	14.32	12.64	13.87
	Be07, Be08, Be10, Be11	13.98	14.54	15.77
	Be07, Be08, Be09, Be11	15.21	15.21	13.76
	Be07, Be08, Be09, Be10	13.09	11.52	12.98

TABLE 10:  $k$ -NN, mean error (m), using statistical mode (MD) and weighted distance (WD), for different BLE4.0 beacon setups for Tx = 0x04 and Tx = 0x07. Best results are shown in bold.

Tx	Configuration	MD (m)			WD (m)		
		$k = 1$	$k = 3$	$k = 5$	$k = 1$	$k = 3$	$k = 5$
0x04	Be07, Be08, Be09, Be10, Be11	2.25	2.17	<b>2.12</b>	2.25	1.91	<b>1.82</b>
	Be08, Be09, Be10, Be11	2.35	2.32	2.33	2.35	2.04	1.99
	Be07, Be09, Be10, Be11	2.25	2.19	2.17	2.25	1.87	1.84
	Be07, Be08, Be10, Be11	2.30	2.26	2.16	2.30	1.92	1.84
	Be07, Be08, Be09, Be11	2.32	2.34	2.30	2.32	2.00	1.93
	Be07, Be08, Be09, Be10	2.29	2.38	2.23	2.29	2.01	1.91
0x07	Be07, Be08, Be09, Be10, Be11	2.30	2.32	<b>2.29</b>	2.26	2.02	1.96
	Be08, Be09, Be10, Be11	2.35	2.36	2.29	2.35	2.03	1.95
	Be07, Be09, Be10, Be11	2.41	2.44	2.44	2.41	2.12	2.07
	Be07, Be08, Be10, Be11	2.31	2.39	2.37	2.31	2.00	<b>1.90</b>
	Be07, Be08, Be09, Be11	2.39	2.43	2.44	2.39	2.16	2.01
	Be07, Be08, Be09, Be10	2.41	2.52	2.44	2.41	2.12	2.04

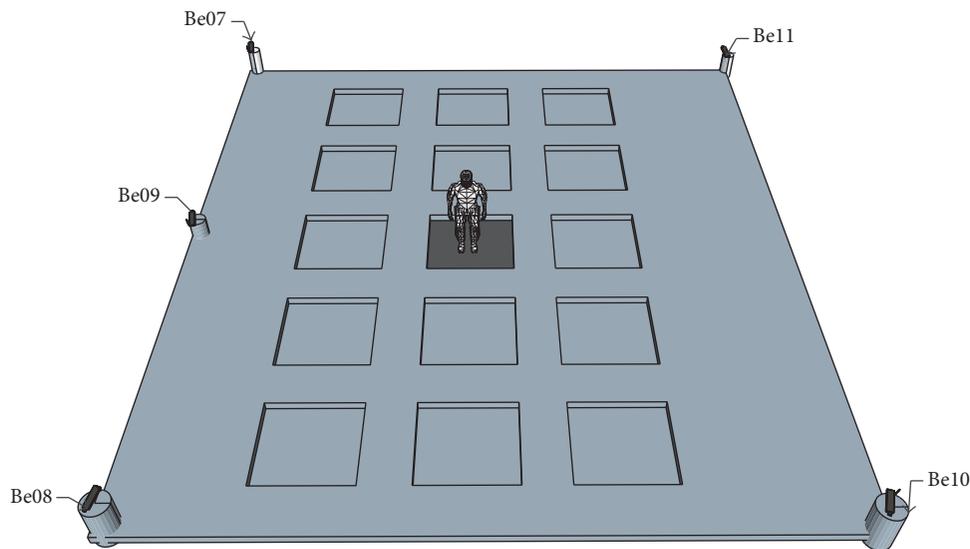


FIGURE 13: Configuration of Experimental Area 2.

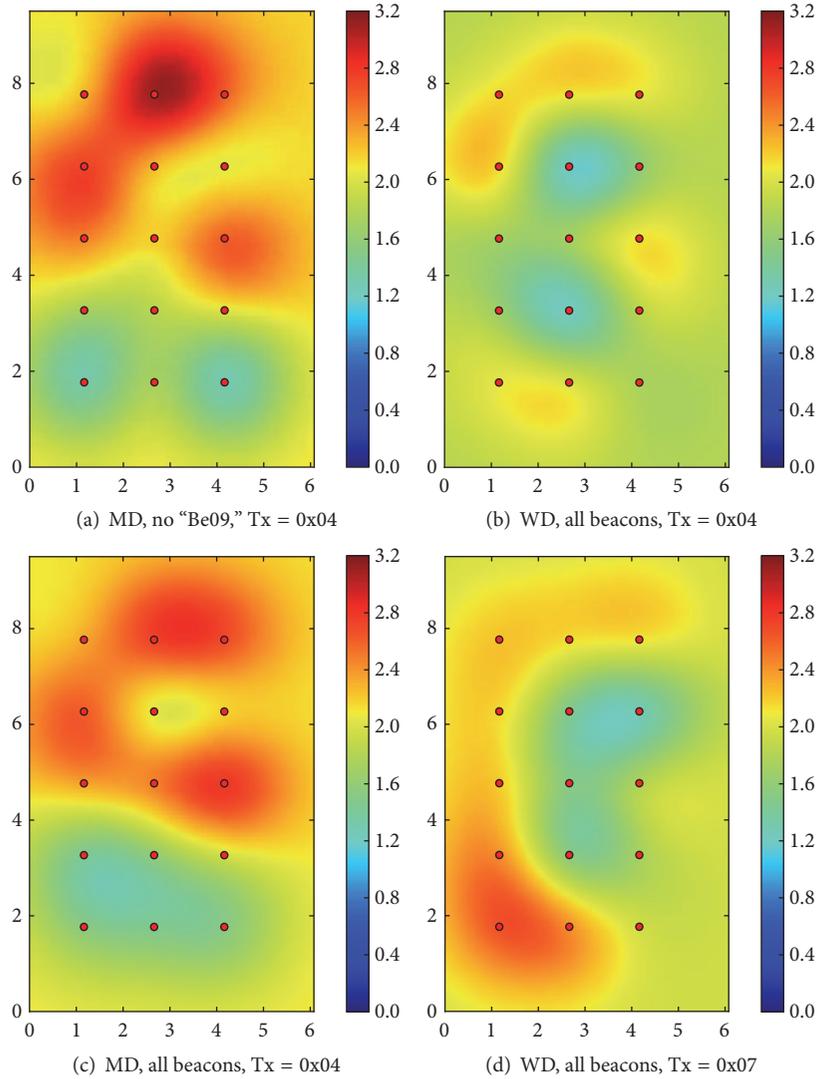


FIGURE 14:  $k$ -NN (with  $k = 5$ ), positioning error heatmaps, using statistical mode (MD) and weighted distance (WD), for different BLE4.0 beacon setups for Tx = 0x04 and Tx = 0x07.

error throughout the whole area; see Figure 14(b). In fact, this system configuration provides the best results in terms of the mean positioning error, as shown in Table 10.

*Local Accuracy.* As already mentioned before in this section, the BLE4.0 beacons located at the corners are essential, as can be seen in Figure 15, which represent the behaviour of the local accuracy throughout the experimental area. Specifically, Figures 15(a) and 15(b) are related to the best global accuracy and mean error for Tx = 0x04, respectively.

In this case, we notice that a more uniform local accuracy results in a lower mean positioning error; see Figures 15(b) and 14(b). More specifically, comparing Figures 15(c) with 15(d), we can see that the use of BLE4.0 beacon “Be09” provides better local accuracy in remote sectors. The results of local accuracy for Tx = 0x07 provided similar results.

6.2.2. Case 2: SVM. Similarly as with Experimental Area 1, we have proceeded to analyze the SVM experimental results

with the same classification metrics for Tx = 0x04 and Tx = 0x07.

*Global Accuracy.* Table 11 shows the global accuracy for Tx = 0x04 and Tx = 0x07. We can see that the best results, for both transmission power levels, are obtained when the configuration of all BLE4.0 beacons is used.

Moreover, using a linear kernel for Tx = 0x04 provides significantly better results than other SVM configurations. For Tx = 0x07 the best result is obtained using a polynomial ( $d = 2$ ) kernel, but far from the results obtained with Tx = 0x04. Comparing results with Experimental Area 1 (see Table 6), we can see how SVM improve the accuracy a 10% when using a bigger experimental area.

*Mean Positioning Error.* Table 12 shows the mean error values. Again, the best results for Tx = 0x04 are obtained when all BLE4.0 beacons are used and with the linear kernel function, but for Tx = 0x07 the best performance is obtained when

TABLE 11: SVM, global accuracy (%) for different BLE4.0 beacon setups for Tx = 0x04 and Tx = 0x07. Best results are shown in bold.

Tx	Configuration	linear	P. $d = 2$	P. $d = 3$
0x04	Be07, Be08, Be09, Be10, Be11	<b>23.48</b>	21.17	20.94
	Be08, Be09, Be10, Be11	19.68	21.29	19.56
	Be07, Be09, Be10, Be11	21.17	18.99	15.65
	Be07, Be08, Be10, Be11	22.55	22.09	21.40
	Be07, Be08, Be09, Be11	20.71	18.53	19.33
	Be07, Be08, Be09, Be10	20.14	21.40	20.02
0x07	Be07, Be08, Be09, Be10, Be11	15.32	<b>17.89</b>	15.55
	Be08, Be09, Be10, Be11	16.00	14.43	14.77
	Be07, Be09, Be10, Be11	13.42	13.98	13.31
	Be07, Be08, Be10, Be11	16.00	17.33	17.56
	Be07, Be08, Be09, Be11	11.63	14.76	15.32
	Be07, Be08, Be09, Be10	13.20	14.43	15.55

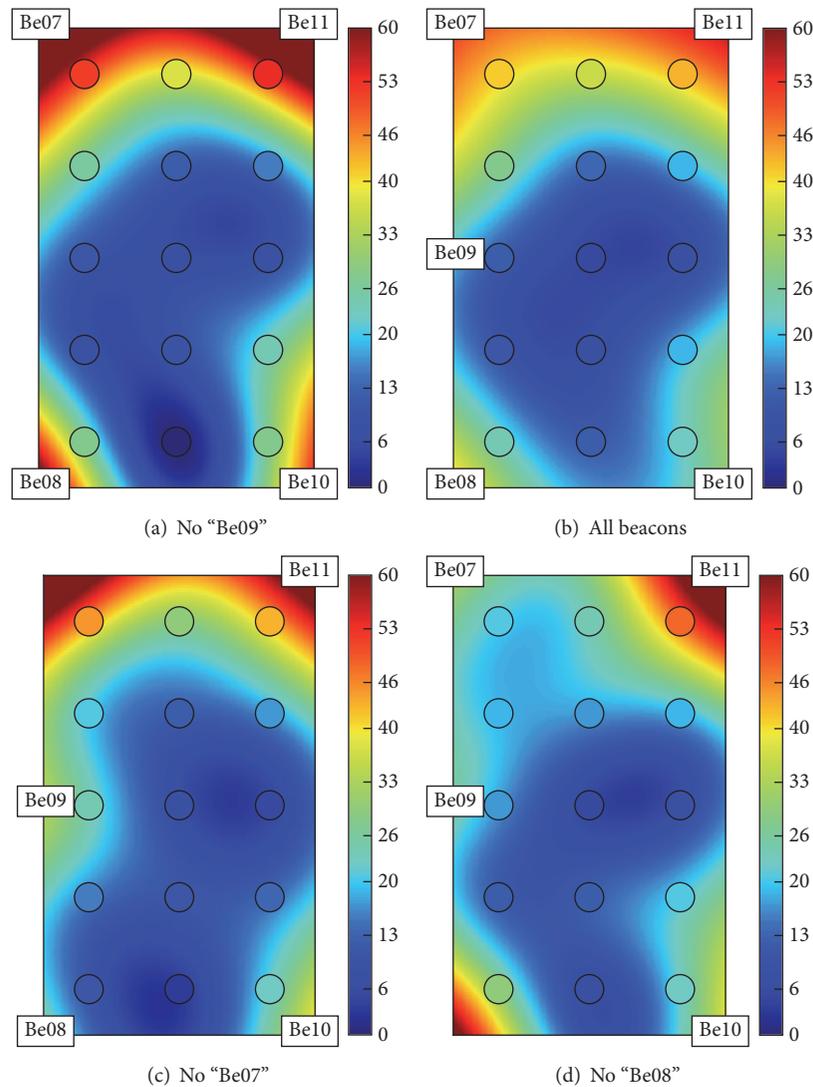
FIGURE 15:  $k$ -NN (with  $k = 5$ ), local accuracy heatmaps for different BLE4.0 beacon setups for Tx = 0x04.

TABLE 12: SVM, mean error (m) for different BLE4.0 beacon setups for Tx = 0x04 and Tx = 0x07. Best results are shown in bold.

Tx	Configuration	linear	P. $d = 2$	P. $d = 3$
0x04	Be07, Be08, Be09, Be10, Be11	<b>1.93</b>	2.01	1.99
	Be08, Be09, Be10, Be11	2.14	2.10	2.10
	Be07, Be09, Be10, Be11	1.96	2.07	2.21
	Be07, Be08, Be10, Be11	2.06	2.05	2.14
	Be07, Be08, Be09, Be11	2.12	2.12	2.12
	Be07, Be08, Be09, Be10	2.09	2.04	2.09
0x07	Be07, Be08, Be09, Be10, Be11	2.39	2.08	2.28
	Be08, Be09, Be10, Be11	2.38	2.21	2.19
	Be07, Be09, Be10, Be11	2.58	2.37	2.29
	Be07, Be08, Be10, Be11	2.33	<b>1.98</b>	2.09
	Be07, Be08, Be09, Be11	2.47	2.36	2.36
	Be07, Be08, Be09, Be10	2.64	2.26	2.36

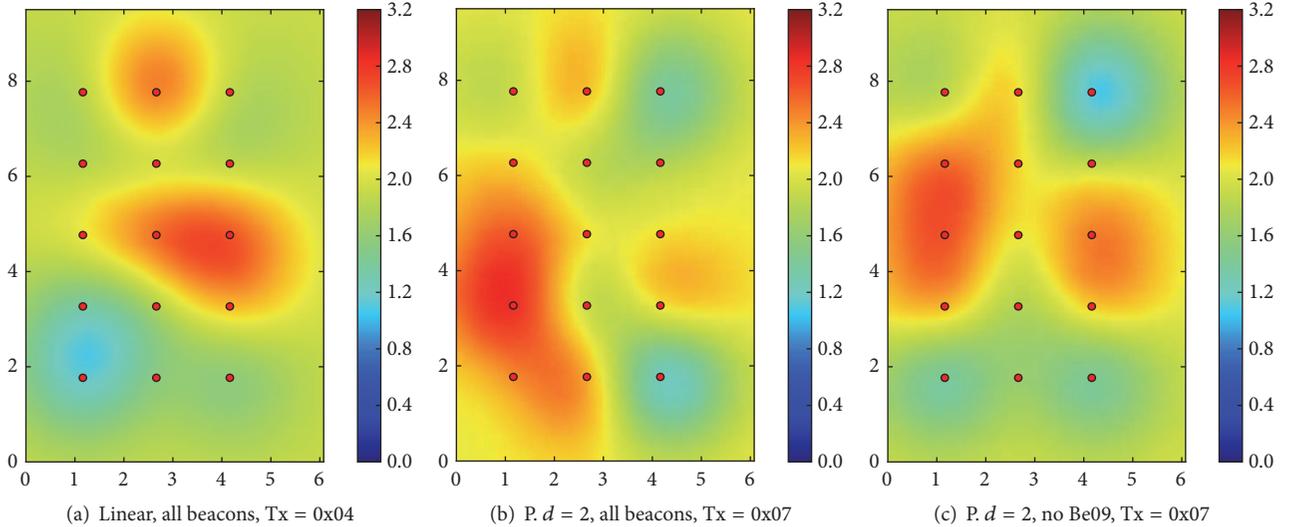


FIGURE 16: SVM, positioning error heatmaps for different BLE4.0 beacon setups for Tx = 0x04 and Tx = 0x07.

the BLE4.0 beacons are located at the corners (configuration without BLE4.0 beacon “Be09”) of the experimental area and using a polynomial ( $d = 2$ ) kernel function.

Figure 16(a) depicts the positioning error heatmaps for the best global accuracy for Tx = 0x04, which is also the same with lowest mean error. Figures 16(b) and 16(c) depict the best global accuracy and lowest mean error for Tx = 0x07, respectively. Similar results are obtained as in previous experiments, where we can observe that a good global accuracy does not provide a lower mean error (in general), but a balanced mean error throughout the area provides better results. In this Experimental Area 2, comparing Figure 14 with Figure 16, we also observe that  $k$ -NN have a more uniform mean error than SVM, since the central sectors normally have lower mean error. Finally, comparing SVM for Experimental Areas 1 and 2 (see Figures 11 and 16) we observe that the mean error is improved in bigger areas, specially in sectors close to the BLE4.0 beacons, with the mean error being more uniform throughout the whole area.

*Local Accuracy.* Figure 17 shows the local accuracy behaviour for different setups and transmission power levels.

We obtained similar results as in previous sections, where in general a balanced local accuracy provides lower mean error (comparing Figures 17(a) and 16(a)). Typically, sectors placed in the corners have higher local accuracy, and BLE4.0 beacon “Be09” usually improves the local accuracy throughout the area as we can see in  $k$ -NN experiments, comparing Figure 15 with Figure 17.

## 7. Lessons Learned

In this section, we summarize the main guidelines on setting the system configuration and algorithm parameters enabling the setting of a more accurate and robust localization mechanism. In our discussion, we will present our main findings by following the parameters related to the classification algorithms and the localization system setup. In order to guide our discussion, we will follow Figure 2, where we have

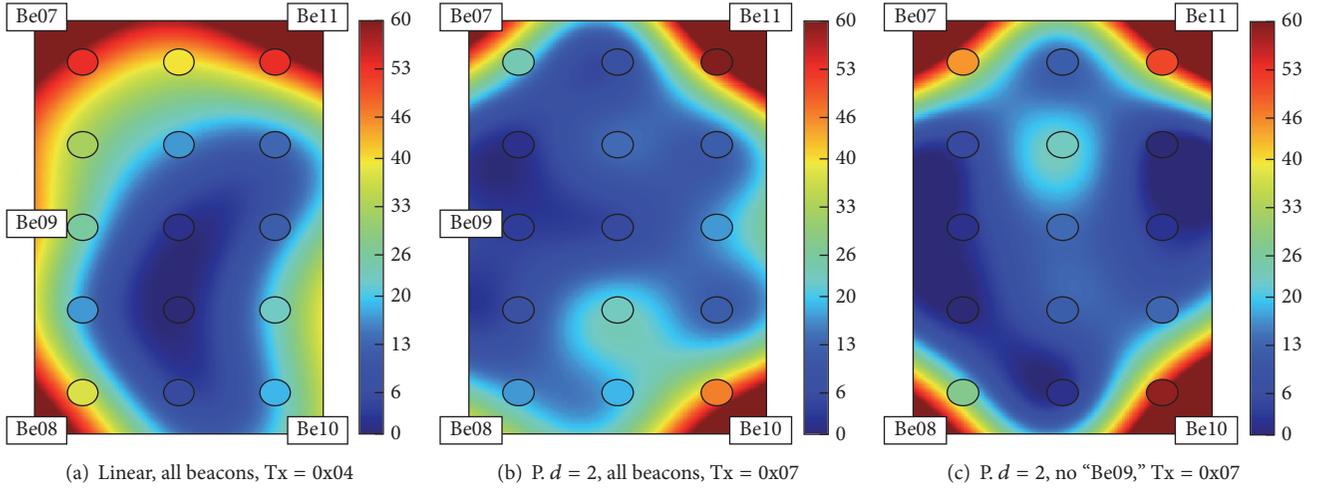


FIGURE 17: SVM, local accuracy (%) heatmaps for different BLE4.0 beacon setups for  $T_x = 0x04$  and  $T_x = 0x07$ .

listed the main parameters to be set. Furthermore, we will refer to the setting of our experimental system as a means to illustrate the applicability of our guidelines. Since the results obtained using the second experimental setup were clearly superior to those obtained in the first setup, we will derive the main guidelines from the lessons learned through our experimental trials. Our main aim is to provide guidelines allowing us to identify the main system and algorithm parameters to be tuned on the development of a robust and accurate BLE4.0-based indoor localization mechanism.

Table 13 summarizes the best system and algorithms setups derived from our study. As already mentioned, we focus on our second experimental setting. In fact, the size and organization of the experimental area were among the first parameters to be set; see Figure 2. From our preliminary study on the channel characterization, Section 3, we were able to identify the signal propagation allowing us to better distinguish the various sectors. As for the actual organization of the experimental area, our results have shown that the use of a guard zone proves to be effective in improving the classification process.

As for all the other parameters related to the system setting, the following guidelines can be derived.

(i) *BLE4.0 Beacon Transmission Power.* The setting of this parameter has to be derived taking into account the information that it may provide in order to enable the classification of the process. In this case, it should provide enough information to enable distinguishing the various sectors of interest. In our particular study, we found out that the use of a medium power level showed slightly better results in terms of the mean positioning error for the case of the  $k$ -NN (WD) and the SVM classification algorithm setups. In the case of the  $k$ -NN (MD) algorithm, the results exhibited a higher discrepancy. For this latter setup, we notice that both system configurations include all five BLE4.0 beacons. It is therefore clear that the information of BLE4.0 beacon "Be09" helps to compensate the discrepancies on the RSSI levels reported by

the BLE4.0 beacons located close to the windows and those located close to the drywall. As for the accuracy reported for the two transmission power setups, we notice that the use of a medium transmission power level considerably improves the accuracy of the localization mechanisms; see Tables 9 and 11.

(ii) *BLE4.0 Beacons Position and Topology.* From our preliminary study, Section 3, we have found out the importance of identifying the materials composing the various walls. In a more complex setup where, for instance, big metal cabinets may be present, the designers should take care of evaluating the RSSI levels close to and around such objects. The information obtained from such preliminary study should condition the actual topology of the system. In our case, we have found out that the levels of the RSSI detected may considerably vary depending on whether the BLE4.0 beacon has been placed close to a drywall or window. As seen in Table 13, the system configurations for the  $k$ -NN (WD) and SVM algorithms when using transmission power level  $T_x = 0x07$  do not include BLE4.0 beacon "Be09." Since in this case the RSSI levels reported by the BLE4.0 beacons close to the window and those to the drywall do not greatly differ, there is no requirement for BLE4.0 beacon "Be09." However, in the case when  $T_x = 0x04$  is preferred, the inclusion of BLE4.0 beacon "Be09" provides some extra information and therefore compensates for the discrepancies on the RSSI levels reported by all the other BLE4.0 beacons.

(iii) *BLE4.0 Beacons Density and Spacing.* From our results, it is clear that the number of required BLE4.0 beacons to cover a given area will depend on the size of the area to cover, the transmission power, and the RSSI levels reported by the BLE4.0 beacons. As our results show, the discrepancies on the RSSI reported by the BLE4.0 beacons due to the structural characteristics of the surrounding walls will require the use of additional BLE4.0 beacons. In our case, the use of four BLE4.0 beacons placed at the four corners provided the best results when using the lowest transmission power; see Table 13. The inclusion of BLE4.0 beacon "Be09" under these latter

TABLE 13: Mean positioning error (m) for the best configurations and setups for  $k$ -NN, using statistical mode (MD) and weighted distance (WD) and SVM.

Algorithm	Power Tx = 0x04	Power Tx = 0x07
$k$ -NN (MD), $k = 5$	[07, 08, 09, 10, 11] $\Rightarrow$ 2.12	[07, 08, 09, 10, 11] $\Rightarrow$ 2.29
$k$ -NN (WD), $k = 5$	[07, 08, 09, 10, 11] $\Rightarrow$ 1.82	[07, 08, 10, 11] $\Rightarrow$ 1.90
SVM, $d = 2$	[07, 08, 09, 10, 11] $\Rightarrow$ 1.93	[07, 08, 10, 11] $\Rightarrow$ 1.98

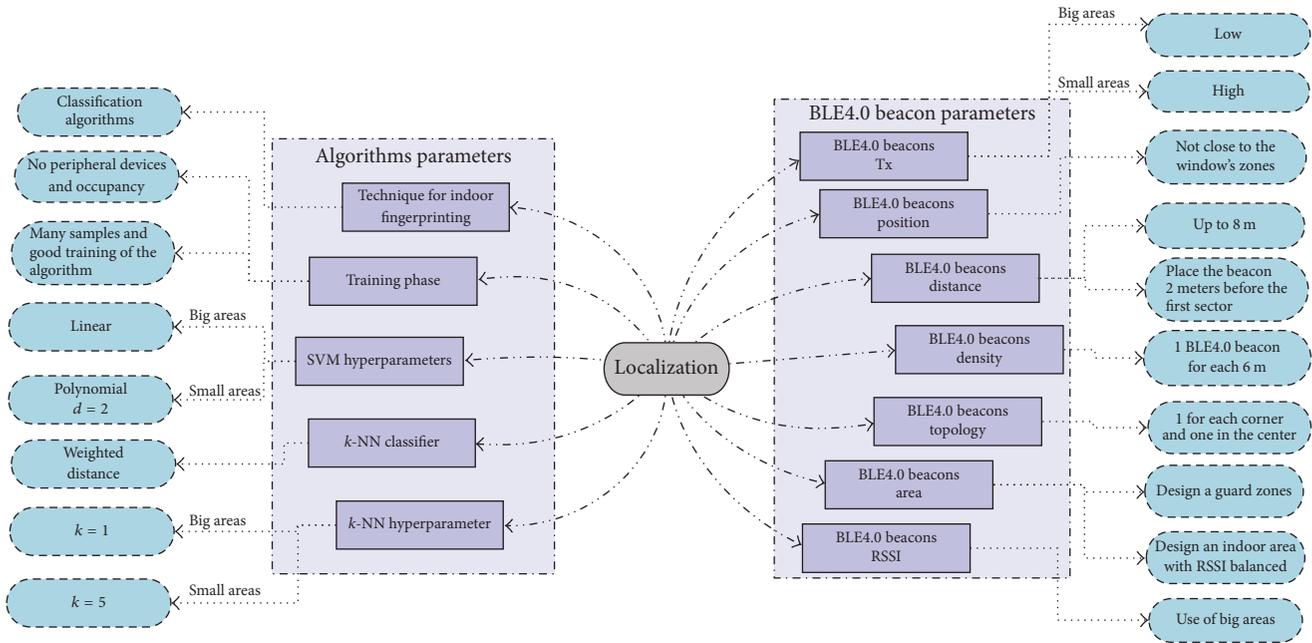


FIGURE 18: Overall recommended values for the parameters.

conditions exhibited slightly worse results; see Tables 10 and 12.

Regarding the operation and setting of the algorithm parameters, the following guidelines can be derived:

- (i) Classification algorithms: a classification process needs to be able to differentiate the RSSI levels of the BLE4.0 beacons at different sectors. Therefore, a steeper fall on the RSSI should provide the best results. Furthermore, ambiguities should be removed in order to reduce errors on the classification process. In our particular setup, we had to limit the distance to 8 m and use medium and low transmission power levels.
- (ii)  $k$ -NN classifier: the use of values of  $k$  higher than or equal to five provides good results given the limitations on the mean positioning error and accuracy reported by BLE-based localization mechanisms. In our setup, we have observed that the use of  $k = 5$  may compensate the discrepancies on the reported RSSI levels when using low transmission power levels. In fact, in the case when using Tx = 0x07, the best configuration for  $k$ -NN (WD) does not make use of BLE4.0 beacon “Be09.” However, the overall best results were obtained for the system configuration

$k$ -NN (WD), with  $k = 5$  and using all the five BLE4.0 beacons. This clearly shows that the use of a transmission power level enabling differentiating the various sectors together with the use of a compensating BLE4.0 beacon “Be09” proves effective.

- (iii) WD criteria work better than MD criteria due to the fact that the first one use all the  $k$  neighbours in order to polish the final result with a weighted average distance.
- (iv) SVM algorithm: similar to the results obtained for the  $k$ -NN algorithm, the choice of the transmission power plays a major role in the setting of the algorithm parameters. In the case when the lower transmission power is used, the best results are obtained for a system configuration not making use of the BLE4.0 beacon “Be09.” Furthermore, the use of a higher transmission power level simplifies the configuration of the SVM. In this latter case, a linear classifier is used.

Finally, Figure 18 summarizes the recommendations for tuning up the indoor localization mechanism. The recommendations include the setting of the parameters of the classification algorithms and configuration of the BLE4.0 beacons.

## 8. Conclusions and Future Plans

In this paper, we have explored the use of two supervised learning algorithms towards the development of BLE4.0 beacon-based location mechanisms. From our study, we have identified that the use of  $k$ -NN and SVM algorithms may prove effective in developing an indoor location fingerprinting mechanism. Furthermore, our results have provided us with some useful insight on the key parameters of both, the physical infrastructure and the supervised learning algorithm. The Tx level and the number and placement of BLE4.0 beacons are the main physical parameters to be looked at, while the number of neighbours to be used plays a major role in the performance of the  $k$ -NN algorithm.

Moreover, with the purpose of improving indoor localization parameters, it is necessary to count with a floor plan defined, for example, guard zones, for the data acquisition phase, in order to differentiate the RSSI in contiguous sectors.

With respect to the environment, we can say that it must be configured so that we have a fall on the RSSI. This is due to the fact that sectors that are close to the transmitters do not provide good results. Other important aspects to improve the indoor localization mechanisms are the topology, the Tx levels, and, above all, the classification algorithms hyperparameters.

Our immediate research activities will focus on the impact of room occupancy using multiple additional sensors and the impact of using different and much more Machine Learning algorithms for the location estimation.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

## Acknowledgments

This work has been partially funded by the “Programa Nacional de Innovación para la Competitividad y Productividad, Innóvate-Perú” of the Peruvian government, under Grant no. FINCyT 363-PNIPC-PIAP-2014, and by the Spanish Ministry of Economy and Competitiveness under Grant nos. TIN2015-66972-C5-2-R and TIN2015-65686-C5-3-R.

## References

- [1] A. S. Paul and E. A. Wan, “RSSI-Based indoor localization and tracking using sigma-point kalman smoothers,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 5, pp. 860–873, 2009.
- [2] J. Yim, C. Park, J. Joo, and S. Jeong, “Extended Kalman Filter for wireless LAN based indoor positioning,” *Decision Support Systems*, vol. 45, no. 4, pp. 960–971, 2008.
- [3] S. Shuo, S. Hao, and S. Yang, “Design of an experimental indoor position system based on RSSI,” in *Proceedings of the 2nd International Conference on Information Science and Engineering, ICISE2010*, pp. 1989–1992, chn, December 2010.
- [4] S. Feldmann, K. Kyamakya, A. Zapater, and Z. Lue, “An indoor Bluetooth-based positioning system: concept, implementation and experimental evaluation,” in *Proceedings of the International Conference on Wireless Networks (ICWN '03)*, pp. 109–113, June 2003.
- [5] R. Chavez-Romero, A. Cardenas, M. Maya, A. Sanchez, and D. Piovesan, “Camera space particle filter for the robust and precise indoor localization of a wheelchair,” *Journal of Sensors*, vol. 2016, Article ID 8729895, 11 pages, 2016.
- [6] J. Martínez-Gómez, M. M. Del Horno, M. Castillo-Cara, V. M. B. Luján, L. O. Barbosa, and I. Garcia-Varea, “Spatial statistical analysis for the design of indoor particle-filter-based localization mechanisms,” *International Journal of Distributed Sensor Networks*, vol. 12, no. 8, 2016.
- [7] M. S. Bargh and R. de Groote, “Indoor localization based on response rate of bluetooth inquiries,” in *Proceedings of the First ACM International Workshop*, pp. 49–54, San Francisco, Calif, USA, September 2008.
- [8] R. Faragher and R. Harle, “An analysis of the accuracy of bluetooth low energy for indoor positioning applications,” in *Proceedings of the 27th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ '14)*, pp. 201–210, Tampa, Fla, USA, September 2014.
- [9] T. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edition, 2001.
- [10] P. Davidson and R. Piche, “A survey of selected indoor positioning methods for smartphones,” *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 1347–1370, 2017.
- [11] L. Pei, R. Chen, J. Liu, T. Tenhunen, H. Kuusniemi, and Y. Chen, “Inquiry-based bluetooth indoor positioning via RSSI probability distributions,” in *Proceedings of the 2nd International Conference on Advances in Satellite and Space Communications (SPACOMM '10)*, pp. 151–156, Athens, Greece, June 2010.
- [12] L. Pei, M. Zhang, D. Zou, R. Chen, and Y. Chen, “A survey of crowd sensing opportunistic signals for indoor localization,” *Mobile Information Systems*, vol. 2016, Article ID 4041291, 16 pages, 2016.
- [13] Y. Guo, Y. Sun, H. Luo, and N. Guizani, “Accurate indoor localization based on crowd sensing,” in *Proceedings of the 2016 International Wireless Communications and Mobile Computing Conference (IWCMC)*, vol. 16, pp. 2852–2868, Paphos, Cyprus, 2016.
- [14] M. S. Aman, H. Jiang, C. Quint, K. Yelamarthi, and A. Abdelgawad, “Reliability evaluation of iBeacon for micro-localization,” in *Proceedings of the 7th IEEE Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, UEMCON 2016*, New York, NY, USA, October 2016.
- [15] M. Castillo-Cara, E. Huaranga-Junco, G. Mondragón-Ruiz, A. Salazar, L. O. Barbosa, and E. A. Antúnez, “Ray: smart indoor/outdoor routes for the blind using bluetooth 4.0 BLE,” in *Proceedings of the 7th International Conference on Ambient Systems, Networks and Technologies, ANT 2016 and the 6th International Conference on Sustainable Energy Information Technology, SEIT 2016*, pp. 690–694, Madrid, Spain, May 2016.
- [16] S. Pagano, S. Peirani, and M. Valle, “Indoor ranging and localisation algorithm based on received signal strength indicator using statistic parameters for wireless sensor networks,” *IET Wireless Sensor Systems*, vol. 5, no. 5, pp. 243–249, 2015.
- [17] M. Sugano, T. Kawazoe, Y. Ohta, and M. Murata, “Indoor localization system using RSSI measurement of wireless sensor network based on ZigBee standard,” in *Proceedings of the 6th IASTED International Multi-Conference on Wireless and Optical*

- Communications: Wireless Sensor Networks, WSN 2006*, pp. 503–508, Alberta, Canada, July 2006.
- [18] S.-Y. Lau, T.-H. Lin, T.-Y. Huang, I.-H. Ng, and P. Huang, “A measurement study of Zigbee-based indoor localization systems under RF interference,” in *Proceedings of the 4th ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization, WiNTECH’09, Co-located with the Mobile Computing and Networking Conference, MobiCom’09*, pp. 35–42, Beijing, China, September 2009.
- [19] A. Corbacho Salas, *Indoor Positioning System based on Bluetooth Low Energy*, Universitat Politècnica de Catalunya, Barcelona, Spain, 2014.
- [20] Z. Farid, R. Nordin, and M. Ismail, “Recent advances in wireless indoor localization techniques and system,” *Journal of Computer Networks and Communications*, vol. 2013, Article ID 185138, 12 pages, 2013.
- [21] R. Faragher and R. Harle, “Location fingerprinting with bluetooth low energy beacons,” *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 11, pp. 2418–2428, 2015.
- [22] X. Zhao, Z. Xiao, A. Markham, N. Trigoni, and Y. Ren, “Does BTLE measure up against wifi? A comparison of indoor location performance,” in *Proceedings of the 20th European Wireless Conference, EW 2014*, pp. 263–268, Barcelona, Spain, May 2014.
- [23] L. N. Chen, B. H. Li, K. Zhao, C. Rizos, and Z. Q. Zheng, “An improved algorithm to generate a Wi-Fi fingerprint database for indoor positioning,” *Sensors*, vol. 13, no. 8, pp. 11085–11096, 2013.
- [24] T. I. Chowdhury, M. M. Rahman, S.-A. Parvez et al., “A multi-step approach for RSSi-based distance estimation using smartphones,” in *Proceedings of the 2015 International Conference on Networking Systems and Security, NSysS 2015*, Dhaka, Bangladesh, January 2015.
- [25] K. Bouchard, R. Ramezani, and A. Naeim, “Features based proximity localization with Bluetooth emitters,” in *Proceedings of the 7th IEEE Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, UEMCON 2016*, New York, NY, USA, October 2016.
- [26] X.-Y. Lin, T.-W. Ho, C.-C. Fang, Z.-S. Yen, B.-J. Yang, and F. Lai, “A mobile indoor positioning system based on iBeacon technology,” in *Proceedings of the 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC ’15)*, pp. 4970–4973, IEEE, Milan, Italy, August 2015.
- [27] Y. Zhao, F. Yin, F. Gunnarsson, M. Amirijoo, E. Ozkan, and F. Gustafsson, “Particle filtering for positioning based on proximity reports,” in *Proceedings of the 18th International Conference on Information Fusion, Fusion 2015*, pp. 1046–1052, Washington, DC, USA, July 2015.
- [28] Y. Zhuang, J. Yang, Y. Li, L. Qi, and N. El-Sheimy, “Smartphone-based indoor localization with bluetooth low energy beacons,” *Sensors*, vol. 16, no. 5, article 596, 2016.
- [29] L. Zhang, X. Liu, J. Song, C. Gurrin, and Z. Zhu, “A comprehensive study of bluetooth fingerprinting-based algorithms for localization,” in *Proceedings of the 27th International Conference on Advanced Information Networking and Applications Workshops (WAINA ’13)*, pp. 300–305, Barcelona, Spain, March 2013.
- [30] P. Kriz, F. Maly, and T. Kozel, “Improving indoor localization using bluetooth low energy beacons,” *Mobile Information Systems*, vol. 2016, Article ID 2083094, 11 pages, 2016.
- [31] Q. Wang, Y. Feng, X. Zhang, Y. Sun, and X. Lu, “IWKNN: An Effective Bluetooth Positioning Method Based on Isomap and WKNN,” *Mobile Information Systems*, vol. 2016, Article ID 8765874, 2016.
- [32] Y. Peng, W. Fan, X. Dong, and X. Zhang, “An iterative weighted knn (iw-knn) based indoor localization method in bluetooth low energy (ble) environment,” in *Proceedings of the Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCOM/IoP/SmartWorld), 2016 Intl IEEE Conferences*, pp. 794–800, Toulouse, France, 2016.
- [33] “Jaalee, Beacon IB0004-N Plus,” <https://www.jaalee.com/>, 2017.
- [34] J. Brownlee, “Non linear algorithms,” in *Master Machine Learning Algorithms*, M. L. Mastery, Ed., pp. 100–120, 2016.

## Research Article

# Linear Kalman Filter for Attitude Estimation from Angular Rate and a Single Vector Measurement

Shangqiu Shan,<sup>1</sup> Zhongxi Hou,<sup>1</sup> and Jin Wu<sup>2</sup>

<sup>1</sup>College of Aerospace Science and Engineering, National University of Defense Technology, Changsha 410073, China

<sup>2</sup>School of Automation, University of Electronic Science and Technology of China, Chengdu, China

Correspondence should be addressed to Jin Wu; jin\_wu\_uestc@hotmail.com

Received 7 May 2017; Accepted 20 July 2017; Published 18 October 2017

Academic Editor: Mohannad Al-Durgham

Copyright © 2017 Shangqiu Shan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, a new Kalman filtering scheme is designed in order to give the optimal attitude estimation with gyroscopic data and a single vector observation. The quaternion kinematic equation is adopted as the state model while the quaternion of the attitude determination from a strapdown sensor is treated as the measurement. Derivations of the attitude solution from a single vector observation along with its variance analysis are presented. The proposed filter is named as the Single Vector Observation Linear Kalman filter (SVO-LKF). Flexible design of the filter facilitates fast execution speed with respect to other filters with linearization. Simulations and experiments are conducted in the presence of large external acceleration and magnetic distortion. The results show that, compared with representative filtering methods and attitude observers, the SVO-LKF owns the best estimation accuracy and it consumes much less time in the fusion process.

## 1. Introduction

Attitude estimation is a key methodology in many applications involving satellite control, unmanned aerial vehicles, inertial human motion tracking, and so forth [1–4]. It requires accurate and efficient estimation from various sensor measurements. In fact, attitude estimation can be done by using different filtering methods, for example, the Kalman filter (KF) and complementary filter (CF) [5]. As these filtering methods adopt various senses of evaluating errors, the performances are quite different. Among all, the KF follows the minimum mean squared error (MMSE) principle that can be proved to be optimal; hence it has been widely used [6, 7].

In the past decade, many filters have been designed to give attitude estimation based on specific sensors. These sensors mainly consist of accelerometer, magnetometer, and inclinometer [8]. For instance, Sabatini proposed an extended Kalman filter (EKF) design by means of inertial and magnetic sensing [9]. Based on the same sensor combination, Li and Wang proposed an effective Adaptive Kalman Filter (AKF) in order to make the filter more accurate under harsh

environments [10]. Using estimation of external acceleration, Suh designed an indirect Kalman filter (IKF, [11]). With algebraic calculations, Valenti et al. proposed a much more simplified quaternion Kalman filter (AQUA q-KF) that fuses the gyroscope, accelerometer, and magnetometer together [12]. Not only can KF methods give the optimal attitude estimates, but they also can make the estimated attitude more smooth. Yet it provides us with the statistic information of the estimated attitude parameters [3].

Apart from KFs, the CF methods are studied widely as well since the KF methods can hardly be used on platforms with relatively low hardware configurations [13, 14]. As a matter of fact, when the Kalman gain matrix is set to a constant matrix, the KF turns out to be a fixed-gain complementary filter, also known as the limiting Kalman filter [15]. Using the gradient descent optimization, Madgwick et al. proposed a fixed-gain CF [16]. Although CF is much faster than KF, it owns different convergence speed for different motions and thus lowers the overall attitude estimation accuracy. In this case, related adaptive schemes are introduced by Tian et al. (Adaptive-Gain Orientation Filter, AGOF) [17], Marantos et al. (Adaptive Wahba's Complementary Filter,

a-WCF) [18], and Vasconcelos et al. [19], which provide adaptive performances. Nonlinear CF is also developed by Mahony et al. using special orthogonal groups [20].

Above methods are mainly focused on specific sensor combinations and they need specific attitude determination. The multi-sensor-observation attitude determination can be solved with solutions to Wahba's problem [21], which, leads to many efficient methods like QUaternion ESTimator (QUEST) [22], Fast Optimal Attitude Matrix (FOAM) [23], Singular Value Decomposition (SVD) [24], and so forth [25, 26]. However, they require numerous costly matrix operations like adjoint, determinant, and eigendecomposition that make the batch processing slow. Wahba's problem requires two or more vector observations because when there is only one vector observation, there would be two optimal associated eigenvectors [27]. In fact Valenti et al. have designed a new way for calculating attitude from a single accelerometer or magnetometer. However, the obtained quaternion is proved to be noncontinuous making it difficult to be fused with a gyroscope [12]. Besides, methods like Madgwick's CF, AQUA q-KF, a-WCF, and AGOF may face a dilemma when one of the sensors fails. This is because the measurement information of these filters uses joint attitude determination or optimization which requires all the sensor data to be effective. Even one sensor's failure may cause the attitude estimation to be incorrect or singular. To prevent the attitude angles from catastrophes, this paper aims to perform attitude estimation from angular rates and a single vector observation with better accuracy and less time consumption. The main contributions are listed below:

- (1) The stable attitude determination solution from a single sensor observation is obtained with some findings in our previous work [28].
- (2) A novel LKF scheme that is the Single Vector Observation Linear Kalman filter (SVO-LKF) is designed and investigated to fuse the angular rates and a single vector observation together.
- (3) Variance analysis of the state model and measurement model is performed which shows some interesting properties. An enhanced variance determination is designed to ensure the stability of the SVO-LKF.
- (4) Several experiments are carried out to verify the effectiveness of the proposed SVO-LKF. Some harsh conditions are added. Comparisons with representative methods are demonstrated with experiment results and related discussion.

This paper is briefly structured as follows: Section 2 contains the attitude determination using a single sensor observation. Section 3 includes the novel design of the SVO-LKF. Simulations and experiments are given in Section 4 in order to show comparisons of various aspects of the proposed SVO-LKF between representative methods. Section 5 contains the concluding remarks.

## 2. Single Sensor Attitude Determination

This section deals with the attitude determination from a single sensor observation. Given a sensor observation  $\mathbf{D}^b = (D_x^b, D_y^b, D_z^b)^\top$  in the body frame, the relationship with its corresponding vector observation in the reference frame can be written as

$$\mathbf{D}^b = \mathbf{C}\mathbf{D}^r, \quad (1)$$

where  $\mathbf{D}^r = (D_x^r, D_y^r, D_z^r)^\top$  denotes the reference vector and  $\mathbf{C}$  is the direction cosine matrix (DCM). In [29], we developed a method using the decomposition of the DCM, which can be given with quaternions

$$\mathbf{C} = (\mathbf{P}_1\mathbf{q}, \mathbf{P}_2\mathbf{q}, \mathbf{P}_3\mathbf{q}) \quad (2)$$

with the parameters of

$$\begin{aligned} \mathbf{P}_1 &= \begin{pmatrix} q_0 & q_1 & -q_2 & -q_3 \\ -q_3 & q_2 & q_1 & -q_0 \\ q_2 & q_3 & q_0 & q_1 \end{pmatrix}, \\ \mathbf{P}_2 &= \begin{pmatrix} q_3 & q_2 & q_1 & q_0 \\ q_0 & -q_1 & q_2 & -q_3 \\ -q_1 & -q_0 & q_3 & q_2 \end{pmatrix}, \\ \mathbf{P}_3 &= \begin{pmatrix} -q_2 & q_3 & -q_0 & q_1 \\ q_1 & q_0 & q_3 & q_2 \\ q_0 & -q_1 & -q_2 & q_3 \end{pmatrix}. \end{aligned} \quad (3)$$

In this section, the theory is extended to arbitrary sensor with exactly the similar approach.

Inserting (2) into (1) gives

$$\mathbf{D}^b = (D_x^r\mathbf{P}_1 + D_y^r\mathbf{P}_2 + D_z^r\mathbf{P}_3)\mathbf{q}. \quad (4)$$

It has been proved that

$$\begin{aligned} \mathbf{P}_1^\top &= \mathbf{P}_1^\dagger, \\ \mathbf{P}_2^\top &= \mathbf{P}_2^\dagger, \\ \mathbf{P}_3^\top &= \mathbf{P}_3^\dagger, \end{aligned} \quad (5)$$

where  $\dagger$  stands for the Moore-Penrose pseudo inverse. In fact, another property has been shown in the appendix of our recent contribution [29]:

$$\mathbf{P}_1\mathbf{P}_2^\top + \mathbf{P}_2\mathbf{P}_1^\top = \mathbf{P}_1\mathbf{P}_3^\top + \mathbf{P}_3\mathbf{P}_1^\top = \mathbf{P}_2\mathbf{P}_3^\top + \mathbf{P}_3\mathbf{P}_2^\top = \mathbf{0}_{3 \times 3}. \quad (6)$$

Also, in [30], we also show that, for a single vector observation, the following equation always holds:

$$\mathbf{q} = \mathbf{W}\mathbf{q}, \quad (7)$$

where  $\mathbf{W}$  can be given by the following:

$$\mathbf{W} = \begin{pmatrix} D_x^b D_x^r + D_y^b D_y^r + D_z^b D_z^r & -D_z^b D_y^r + D_y^b D_z^r & D_z^b D_x^r - D_x^b D_z^r & -D_y^b D_x^r + D_x^b D_y^r \\ -D_z^b D_y^r + D_y^b D_z^r & D_x^b D_x^r - D_y^b D_y^r - D_z^b D_z^r & D_y^b D_x^r + D_x^b D_y^r & D_z^b D_x^r + D_x^b D_z^r \\ D_z^b D_x^r - D_x^b D_z^r & D_y^b D_x^r + D_x^b D_y^r & -D_x^b D_x^r + D_y^b D_y^r - D_z^b D_z^r & D_z^b D_y^r + D_y^b D_z^r \\ -D_y^b D_x^r + D_x^b D_y^r & D_z^b D_x^r + D_x^b D_z^r & D_z^b D_y^r + D_y^b D_z^r & -D_x^b D_x^r - D_y^b D_y^r + D_z^b D_z^r \end{pmatrix}. \quad (8)$$

Note that, in [28], a similar equation is derived showing a new continuous solution to accelerometer attitude determination. Then, following such commitment, we can also obtain the solution from arbitrary single vector observation by

$$\mathbf{q} = \frac{1}{2} (\mathbf{W} + \mathbf{I}) \mathbf{q}_{\text{random}}, \quad (9)$$

where  $\mathbf{q}_{\text{random}}$  is a randomly chosen unit quaternion, if and only if  $\mathbf{W}^2 = \mathbf{I}$ . As a matter of fact, the square of  $\mathbf{W}$  can

actually be computed by [30]

$$\mathbf{W}^2 = \left[ (D_x^b)^2 + (D_y^b)^2 + (D_z^b)^2 \right] \cdot \left[ (D_x^r)^2 + (D_y^r)^2 + (D_z^r)^2 \right] \mathbf{I} = \mathbf{I}. \quad (10)$$

Hence, (9) can be used as an approach for measuring quaternion from a single vector observation

$$\mathbf{J}(\mathbf{q}_{y_k}, \mathbf{D}^r) = \begin{pmatrix} D_x^r q_0 + D_y^r q_3 - D_z^r q_2 & D_y^r q_0 - D_x^r q_3 + D_z^r q_1 & D_x^r q_2 - D_y^r q_1 + D_z^r q_0 \\ D_x^r q_1 + D_y^r q_2 + D_z^r q_3 & D_x^r q_2 - D_y^r q_1 + D_z^r q_0 & D_x^r q_3 - D_y^r q_0 - D_z^r q_1 \\ D_y^r q_1 - D_x^r q_2 - D_z^r q_0 & D_x^r q_1 + D_y^r q_2 + D_z^r q_3 & D_x^r q_0 + D_y^r q_3 - D_z^r q_2 \\ D_y^r q_0 - D_x^r q_3 + D_z^r q_1 & D_z^r q_2 - D_y^r q_3 - D_x^r q_0 & D_x^r q_1 + D_y^r q_2 + D_z^r q_3 \end{pmatrix}, \quad (11)$$

$$\mathbf{J}(\mathbf{q}_{y_k}, \hat{\mathbf{q}}_{k-1}) = \frac{1}{2} (\mathbf{W} + \mathbf{I}).$$

### 3. Design of the Novel KF Scheme

**3.1. Kalman Filter Basis.** A discrete linear system without external control is constructed as follows [31]:

$$\begin{aligned} \mathbf{x}_k &= \Phi_{k,k-1} \mathbf{x}_{k-1} + \xi_k, \\ \mathbf{y}_k &= \mathbf{H}_k \mathbf{x}_k + \varepsilon_k, \end{aligned} \quad (12)$$

where  $\mathbf{x}$ ,  $\mathbf{y}$  denote the state vector and measurement vector, respectively.  $k$  denotes the time epoch.  $\Phi_{k,k-1}$  defines the transition matrix from epoch  $k-1$  to epoch  $k$ .  $\mathbf{H}_k$  denotes the measurement matrix at epoch  $k$ .  $\xi$  and  $\varepsilon$  stand for the white Gaussian process noise and measurement noise respectively, such that

$$\begin{aligned} \xi_k &\approx N(\mathbf{0}, \Sigma_{\xi_k}), \\ \varepsilon_k &\approx N(\mathbf{0}, \Sigma_{\varepsilon_k}), \end{aligned} \quad (13)$$

where  $\Sigma$  denotes the variance or covariance matrix. Using the following classical Kalman filtering process, we can recursively achieve the estimation [32]

$$\begin{aligned} \mathbf{x}_k^- &= \Phi_{k,k-1} \hat{\mathbf{x}}_{k-1}, \\ \Sigma_{\mathbf{x}_k^-} &= \Phi_{k,k-1} \Sigma_{\mathbf{x}_{k-1}^-} \Phi_{k,k-1}^\top + \Sigma_{\xi_k}, \\ \hat{\mathbf{x}}_k &= \mathbf{x}_k^- + \mathbf{G}_k (\mathbf{y}_k - \mathbf{H}_k \mathbf{x}_k^-), \end{aligned}$$

$$\begin{aligned} \Sigma_{\hat{\mathbf{x}}_k} &= (\mathbf{I} - \mathbf{G}_k \mathbf{H}_k) \Sigma_{\mathbf{x}_k^-}, \\ \mathbf{G}_k &= \Sigma_{\mathbf{x}_k^-} \mathbf{H}_k^\top (\mathbf{H}_k \Sigma_{\mathbf{x}_k^-} \mathbf{H}_k^\top + \Sigma_{\varepsilon_k})^{-1}, \end{aligned} \quad (14)$$

where  $\mathbf{G}_k$  denotes the Kalman gain matrix.

**3.2. Filter Design.** The attitude quaternion is chosen as the state vector. The 1st-order quaternion kinematic equation is transformed into the following state model [33]:

$$\mathbf{q}_k = \left[ \mathbf{I} + \frac{T_k}{2} [\Omega \times] \right] \mathbf{q}_{k-1} + \xi_k, \quad (15)$$

where

$$[\Omega \times] = \begin{pmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{pmatrix} \quad (16)$$

is formed by the angular velocity  $\omega = (\omega_x, \omega_y, \omega_z)^\top$  and  $T_k$  is the sampling time for  $k$ th epoch. The noise item  $\xi_k$  can be given by

$$\xi_k = -\frac{T_k}{2} \Xi(\mathbf{q}_{k-1}) \mathbf{v}_{g,k}, \quad (17)$$

```

Initialize:
 $k = 0, \hat{\mathbf{q}}_0 = (1, 0, 0, 0)^\top,$ 
 $\Sigma_{\mathbf{v}_g} = \text{diag}(\sigma_{g,x}^2, \sigma_{g,y}^2, \sigma_{g,z}^2),$ 
 $\Sigma_{\mathbf{D}^b} = \text{diag}(\sigma_{D_x^b}^2, \sigma_{D_y^b}^2, \sigma_{D_z^b}^2),$ 
 $\Sigma_{\hat{\mathbf{q}}_0} = \mathbf{I}_{4 \times 4}, \mathbf{D}^r = (D_x^r, D_y^r, D_z^r)^\top.$ 
while no stop commands received do
  (1) Input:  $\omega = (\omega_x, \omega_y, \omega_z)^\top, \mathbf{D}^b = (D_x^b, D_y^b, D_z^b)^\top, T_k$ 
  (2)  $k = k + 1, \mathbf{D}^b = \frac{\mathbf{D}^b}{\|\mathbf{D}^b\|}$ 
  (3) Prediction:  $\mathbf{q}_k^- = [\mathbf{I}_{4 \times 4} + (T_k/2) [\Omega \times]] \hat{\mathbf{q}}_{k-1}$ 
  (4) Propagation:  $\Sigma_{\mathbf{q}_k^-} = [\mathbf{I}_{4 \times 4} + (T_k/2) [\Omega \times]] \hat{\mathbf{q}}_{k-1} [\mathbf{I}_{4 \times 4} + (T_k/2) [\Omega \times]]^\top + \Sigma_{\varepsilon_k}$ 
  (5) Kalman Gain:  $\mathbf{G}_k = \Sigma_{\mathbf{q}_k^-} (\Sigma_{\mathbf{q}_k^-} + \Sigma_{\varepsilon_k})^{-1}$ 
  (6) Estimation:  $\hat{\mathbf{q}}_k = \mathbf{q}_k^- + \mathbf{K}_k (\mathbf{q}_{y_k} - \mathbf{q}_k^-)$ 
  (7) Update of Covariance:  $\Sigma_{\hat{\mathbf{q}}_k} = (\mathbf{I}_{4 \times 4} - \mathbf{K}_k) \Sigma_{\mathbf{q}_k^-}$ 
  (8) Normalization:  $\hat{\mathbf{q}}_k = \hat{\mathbf{q}}_k / \|\hat{\mathbf{q}}_k\|$ 
end while

```

ALGORITHM 1: Proposed SVO LKF.

where

$$\Xi(\mathbf{q}) = \begin{pmatrix} q_1 & q_2 & q_3 \\ -q_0 & -q_3 & -q_2 \\ q_2 & -q_0 & -q_1 \\ -q_2 & q_1 & -q_0 \end{pmatrix} \quad (18)$$

and  $\mathbf{v}_{gk}$  stands for the gyroscopic noise with the auto-covariance of  $\Sigma_{\mathbf{v}_{gk}} = \text{diag}(\sigma_{g,x}^2, \sigma_{g,y}^2, \sigma_{g,z}^2)$ . Hence the auto-covariance of  $\xi_k$  can be calculated by [12]

$$\Sigma_{\xi_k} = \frac{T_k^2}{4} \Xi(\mathbf{q}_{k-1}) \Sigma_{\mathbf{v}_{gk}} \Xi^\top(\mathbf{q}_{k-1}). \quad (19)$$

The quaternion from a strapdown sensor can be used as the measurement vector:

$$\mathbf{q}_{y_k} = \mathbf{q}_k + \varepsilon_k. \quad (20)$$

In this paper, the measurement quaternion is obtained via (9)'s mutant; that is, it can be obtained with

$$\mathbf{q}_{y_k} = \frac{\mathbf{W} + \mathbf{I}}{2} \hat{\mathbf{q}}_{k-1}, \quad (21)$$

where  $\hat{\mathbf{q}}_{k-1}$  stands for the estimated quaternion at epoch  $k - 1$ . Obviously, the variance of the proposed observation model relies on the variance of the vector observation and the variance of the estimated quaternion at last time epoch. Seen from the above equation, there are also nonlinearities inside it since there are multiplication items between  $\mathbf{W}$  and  $\mathbf{q}$ . In this case, the only way to derive the observation variance is to linearize the above equation. Here we define  $\mathbf{J}(\mathbf{A}, \mathbf{B})$

as the Jacobian matrix of  $\mathbf{A}$ , with respect to  $\mathbf{B}$ . Given the variances of vector observation and last estimated quaternion  $\Sigma_{\mathbf{D}^r, k}, \Sigma_{\hat{\mathbf{q}}_{k-1}}$ , respectively, we can compute (11). The ranks of the two matrices are analytically computed as 3 and 2. That is to say, although the observation variance has correlation with last estimated quaternion, the final computed variance has much more relationship with the vector observation. Hence, in the current presented algorithm, the correlation between the process and observation models can be ignored. In this way the observation variance is approximated as

$$\Sigma_{\varepsilon_k} = \mathbf{J}(\mathbf{q}_{y_k}, \mathbf{D}^r) \Sigma_{\mathbf{D}^r, k} \mathbf{J}^\top(\mathbf{q}_{y_k}, \mathbf{D}^r). \quad (22)$$

The overall calculation procedure of the proposed scheme is shown in Algorithm 1.

As  $\mathbf{J}(\mathbf{q}_{y_k}, \mathbf{D}^r)$  has the rank of 3, the measurement quaternion can only obtain an observability of maximum two Euler angles. Such finding copes with the analytic results from that in [30], as well. However, note that, in the end of each filtering update, the quaternion would be normalized, adding to an insurance that the quaternion should not be divergent. The same circumstance occurs in the variance of quaternion kinematic equation and is resolved by quaternion normalization, too.

In engineering practice, during filtering process, the covariance of the state may become negative definite, which would lead to filtering divergence. The reason is that when the algorithm is applied on embedded platform, the word-length of the float-point numbers may cause numerical loss [34]. Hence, the square root Kalman filter (SRKF, [35]) is

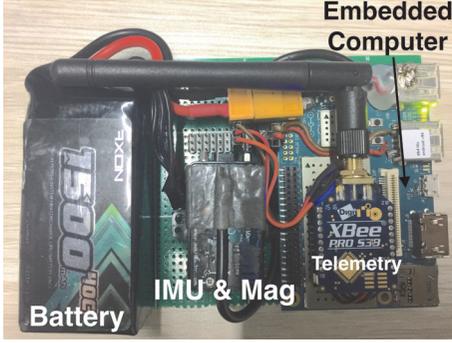


FIGURE 1: Designed experimental platform for the test and verification of the proposed filter.

adopted to compensate for such problem. Using the Cholesky factorization, SRKF's update equations are given as follows:

$$\begin{aligned}
 \mathbf{S}_{0,0} &= \sqrt{\text{Var}(\mathbf{x}_0)}, \\
 \mathbf{x}_k^- &= \Phi_{k,k-1} \hat{\mathbf{x}}_{k-1}, \\
 \mathbf{S}_{k,k-1} &= (\Phi_{k,k-1} \mathbf{S}_{k-1,k-1} \sqrt{\Sigma_{\xi_k}}) (\Phi_{k,k-1} \mathbf{S}_{k-1,k-1} \sqrt{\Sigma_{\xi_k}})^\top \\
 \mathbf{F}_k &= (\mathbf{H}_k \mathbf{S}_{k,k-1} \mathbf{S}_{k,k-1}^\top \mathbf{H}_k^\top + \Sigma_{\varepsilon_k})^c, \\
 \mathbf{S}_{k,k} &= \mathbf{S}_{k,k-1} \begin{bmatrix} \mathbf{I} - \mathbf{S}_{k,k-1}^\top \mathbf{H}_k^\top (\mathbf{F}_k^\top)^{-1} \\ \{\mathbf{F}_k + (\Sigma_{\varepsilon_k})^c\}^{-1} \mathbf{H}_k \mathbf{S}_{k,k-1} \end{bmatrix}, \\
 \mathbf{G}_k &= \mathbf{S}_{k,k-1} \mathbf{S}_{k,k-1}^\top \mathbf{H}_k^\top (\mathbf{F}_k^\top)^{-1} \mathbf{F}_k^{-1}, \\
 \hat{\mathbf{x}}_k &= \mathbf{x}_k^- + \mathbf{G}_k (\mathbf{y}_k - \mathbf{H}_k \mathbf{x}_k^-),
 \end{aligned} \tag{23}$$

where  $\text{Var}\{\cdot\}$  stands for the variance and  $\{\mathbf{A}\}^c$  denotes matrix  $\mathbf{A}$ 's Cholesky factorization. For common tasks on 64-bit PC, it is not necessary to use such formulations.

#### 4. Experiments, Simulations, and Results

In this section, we are going to carry out several experiments to verify the performances of the proposed filter with respect to some representative methods. In order to achieve high precision data acquisition, a hybrid experimental platform is designed. As can be seen from Figure 1, the designed hardware consists of several parts including the power source, inertial measurement unit (IMU), magnetometer, telemetry, and an embedded computer. The power source is a Li-Po battery with the capacity of 1500 mah and voltage of 14.8 V. The IMU and magnetometer are integrated as the product 3DM-GX3-25 provided by Microstrain (c). It generates raw data of angular rate, acceleration, and magnetic sensing with a wide output rate from 50 Hz to 1000 Hz. Besides, this module also gives high precision Euler angles; that is, it can be used as a standard attitude and heading reference system (AHRS). The Xbee S3B Pro telemetry with the frequency of 900 MHz is

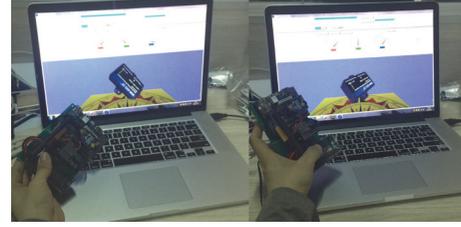


FIGURE 2: The upper monitor can effectively show the real-time attitude from SVO-LKF.

utilized for robust wireless data transmission. The embedded computer has the central processing unit (CPU) of an 4-core ARM-CortexA53-based Qualcomm 410c chip with the clock speed of 1.4 GHz. Such high-configuration platform ensures that different algorithms can be concurrently executed. Moreover, to make the visualization of the attitude estimation and data acquisition more intuitive, an upper monitor is designed (see Figure 2). The upper monitor can also log the raw data and different attitude estimation results, which are later analyzed using the MATLAB r2015b software.

Several representative methods are adopted to make comparisons with the proposed filter. Since the proposed filter is designed as 4-dimensional, another EKF with the same dimension of state variables is introduced [36]. Not only KF, but other complementary filters are compared as well, for example, the nonlinear complementary filter proposed by Mahony et al. [20] and the linear complementary filter proposed by Madgwick et al. [16]. Apart from these algorithms, to verify the performance of the proposed filter under harsh conditions, an adaptive KF named as the indirect Kalman filter (IKF) proposed by Suh is studied [11]. During the whole experiments, the attitude outputs from the 3DM-GX3-25 are served as the reference.

**4.1. Single Vector Observation from an Accelerometer.** In this subsection, the single vector observation is acquired from an accelerometer. First an experiment is conducted with normal motion. Data acquisition is carried out with the rate of 500 Hz. The raw data from the gyroscope and accelerometer is shown in Figure 3.

The parameters of the proposed filter are measured as  $\Sigma_{\mathbf{v}_g} = 0.0001 \times \text{diag}(1, 1, 1)$  and  $\Sigma_{\mathbf{D}^b} = 0.01 \times \text{diag}(1, 1, 1)$ . The reference vector of the accelerometer is set to  $\mathbf{D}^r = (0, 0, 1)^\top$ . With acquired raw data from sensors, the attitude estimation results from different sources are obtained using the designed embedded computer, which is depicted in Figure 4. Mahony's filter and Madgwick's filter gains are set to  $Kp = 0.5$  and  $\beta = 0.1$ , respectively.

As the accelerometer can only measure the roll and pitch angles, the yaw angle is not compensated for during the sensor fusion. This is indicated by the black arrow in Figure 4 showing the drift of the yaw angles. Thus the estimated yaw angles with various algorithms are basically the same. However, the roll and pitch angles are always compensated for by the accelerometer. In this figure, we can see that IKF is the worst intuitively. The attitude curves of the other algorithms

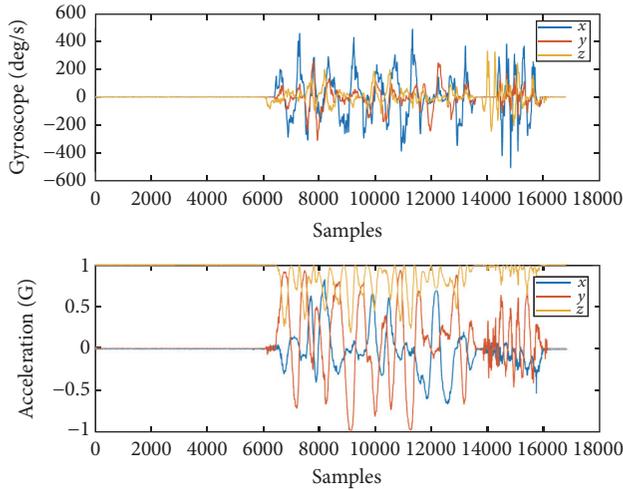


FIGURE 3: Raw data from the gyroscope and accelerometer during a normal motion.

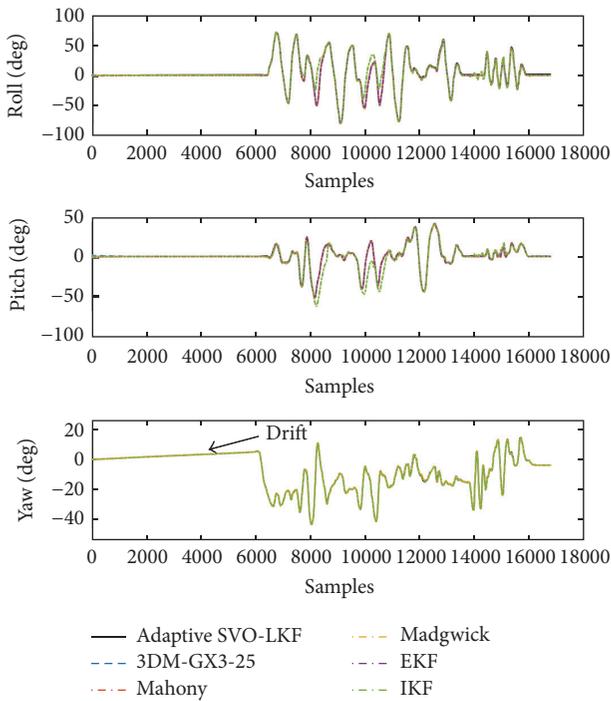


FIGURE 4: Estimated attitude angles from different sources. The black arrow indicates that the drift of the yaw angles exists.

are mixed together; that is, these algorithms have very similar estimation abilities of the roll and pitch angles. Compared with the reference angles from 3DM-GX3-25, the static root-mean-square errors (RMSEs) of various algorithms are listed in Table 1.

The statistics show that the proposed filter owns the best static mean errors of angles compared with other algorithms. However, a filter's performance should be tested under some harsh conditions, for example, in the presence of large external acceleration. In this case, the motion is generated

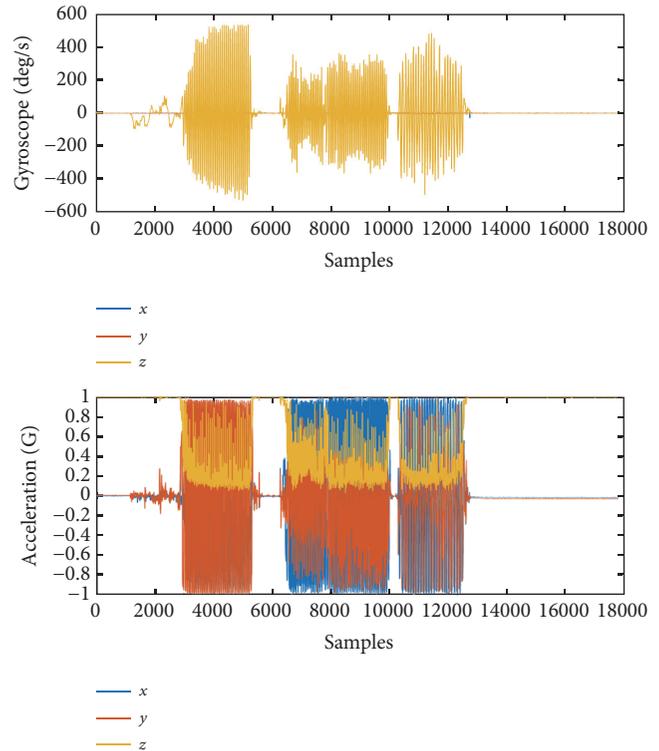


FIGURE 5: Raw data from the gyroscope and accelerometer in the presence of large external acceleration.

with hand on a table. The raw data collected are shown in Figure 5.

The parameters of the filter are the same with that of the previous experiment. Attitude estimation results are depicted in Figure 6. In fact since the motion is generated on a horizontal table, the real roll and pitch angles should be approximately 0. However, as the accelerometer is a very important signal source for AHRS, even the estimated angles from 3DM-GX3-25 contain large errors ranging from 0 to  $\pm 20$  degrees. Similarly, the behaviours of EKF, IKF, Mahony's filter, and Madgwick's filter are significantly impacted by external acceleration. The proposed SVO-LKF, however, remains relatively smooth compared with other filters. This is because the proposed filter is free of approximation of state and measurement models; that is, it is very accurate mathematically. Hence, attitude estimation from SVO-LKF is more accurate than EKF. Besides, the KF can dynamically calculate the Kalman gain matrix while, for the two adopted complementary filters, the gains are fixed. Consequently, the dynamic performances of these complementary filters are not comparable with the proposed SVO-LKF. The RMSEs of angles with respect to 0 and from different sources are listed in Table 2. The results show that the proposed filter owns the best attitude estimation performance under dynamic conditions.

**4.2. Time Consumption.** Time consumption significantly determines the characteristic of a certain algorithm when it is applied on different platforms. Here, we would like to

TABLE 1: Static RMSEs of roll and pitch angles.

Sources	Mean roll error (deg)	Mean pitch error (deg)
Proposed SVO-LKF	0.09776	0.16171
EKF	0.19971	0.19569
IKF	2.46761	8.56740
Mahony's filter	0.22335	0.22336
Madgwick's filter	0.30318	0.22512

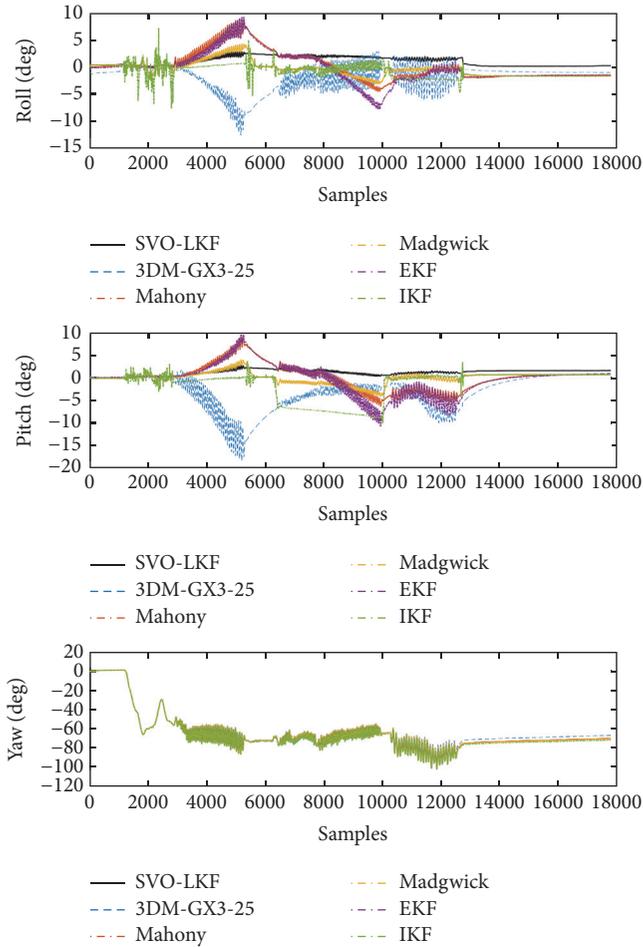


FIGURE 6: Attitude estimation results in the presence of large external acceleration.

study the time consumption performances of different filters. Using the samples from the experiment in the presence of large external acceleration, we obtain the time consumption information in Figure 7. We can find out that the proposed SVO-LKF is much more computationally cheap compared with other KF algorithms. This is because the filter is designed to be linear which uses the common KF procedures. However, EKF consumes more time on the calculation of linear approximation and noise propagation, which makes it slower than SVO-LKF. IKF is designed to be a 9-dimensional filter; hence the matrix operations are far more than that of the SVO-LKF. The internal reason that the proposed SVO-LKF is faster than previous algorithms is that the measurement model is

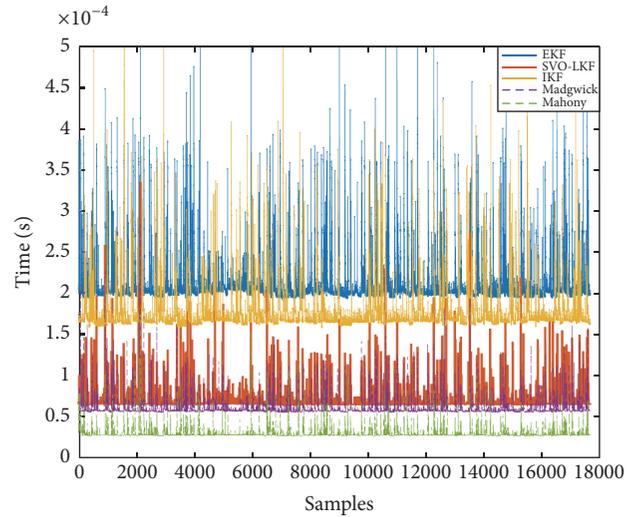


FIGURE 7: Time consumption of different sources adopted.

instantly obtained via linear matrix multiplications. For the EKF, the linearization consumes a lot during the filtering update. For the IKF, not only does it have 6-dimensional state vector (for just gyro and accelerometer), it also adopts the matrix factorization for detection of external acceleration as well. Hence the SVO-LKF is faster in this way; that is, it maintains a simplest formulation for KF based the quaternion only, which does not generate computation burden for other estimation.

From another aspect, we can see that SVO-LKF consumes more time than the two adopted complementary filters. We can also find out that the differences are not very large. Yet the SVO-LKF is proved to be the best among all these filters. Hence we think that although SVO-LKF is computationally expensive compared to complementary filters; it is well worth the advances of the accuracy. Besides, SVO-LKF can provide the stochastic information of the estimated quaternions, which is significant in some later applications like spacecraft control and quality monitoring. The mean time consumption of different filters is shown below (see Table 3).

The standard deviation of time consumption actually determines the execution stability of the attitude estimation system. If the software is implemented on embedded platforms with real-time operating system, for example, FreeRTOS and uCOS, such behaviour would significantly influence the overall scheduling of the system. We can see from the above table that the proposed SVO-LKF is the best among compared Kalman filters. It is also the comparable

TABLE 2: Dynamic RMSEs of roll and pitch angles.

Sources	Mean roll error (deg)	Mean pitch error (deg)
Proposed SVO-LKF	2.24812	1.86264
EKF	7.963827	11.010986
IKF	15.631756	8.465401
Mahony's filter	6.83063	6.83063
Madgwick's filter	2.415353	1.92556

TABLE 3: Statistics of time consumption from various algorithms.

Sources	Mean (s)	Standard deviation (s)
Proposed SVO-LKF	$7.160380763358772 \times 10^{-5}$	$1.707471484592145 \times 10^{-5}$
EKF	$2.209972969182915 \times 10^{-4}$	$4.794187657353048 \times 10^{-5}$
IKF	$1.815493757421541 \times 10^{-4}$	$3.406751301913417 \times 10^{-5}$
Mahony's filter	$3.093860316652542 \times 10^{-5}$	$8.491151067498806 \times 10^{-6}$
Madgwick's filter	$6.287929126378291 \times 10^{-5}$	$1.653065880139708 \times 10^{-5}$

with other complementary filters on this aspect as well, which ensures the robustness of SVO-LKF in applications.

## 5. Conclusion

Based on our previous work, this paper solved the attitude determination from a single vector observation. A novel linear Kalman filter (SVO-LKF) is designed to fuse the gyroscope's data and the single sensor observation together. The quaternion kinematic equation is used as the state model while the quaternion attitude determination from a single vector observation is treated as the measurement. Throughout analysis, we obtain the variance information of different models.

Several experiments are designed including the gyroscope-accelerometer and gyroscope-magnetometer combination. The experiments not only show the performances of the SVO-LKF along with other representative methods, but also give the details of the performances in the presence of large external acceleration. Related results are thoroughly discussed which shows that the proposed SVO-LKF is accurate in both static and dynamic conditions. Also when facing magnetic distortion, it basically remains uninfluenced compared to other filters. Finally, time consumption of various algorithms is given which shows that the SVO-LKF is the fastest one compared to other KF schemes. Although it is more time-costly than complementary filters, the advances of the accuracy are well worth the loss of time.

This paper shows that, using a single vector observation, we can make the attitude estimation much more accurate. Could it be a motivation for attitude estimation with even more vector observations? We think this would be another task for us to study in the future.

## Conflicts of Interest

The authors declare no conflicts of interest regarding the content of this paper.

## Acknowledgments

This work is supported by National Natural Science Foundation of China under the Grant no. 61450010. Professor Young Soo Suh provided the authors with the MATLAB code of the indirect Kalman filter. The authors genuinely thank them for their support.

## References

- [1] J. Zhu, Z. Zhou, Y. Li, C. Rizos, and X. Wang, "Further development of the attitude difference method for estimating deflections of the vertical in real time," *Measurement Science and Technology*, vol. 27, no. 7, Article ID 075004, 2016.
- [2] Z. Zhou, Y. Li, J. Zhang, and C. Rizos, "Integrated navigation system for a low-cost quadrotor aerial vehicle in the presence of rotor influences," *Journal of Surveying Engineering*, vol. 143, no. 1, 2017.
- [3] Z. Zhou, Y. Li, J. Liu, and G. Li, "Equality constrained robust measurement fusion for adaptive kalman-filter-based heterogeneous multi-sensor navigation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2146–2157, 2013.
- [4] X. Yun, J. Calusdian, E. R. Bachmann, and R. B. McGhee, "Estimation of human foot motion during normal walking using inertial and magnetic sensor measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 7, pp. 2059–2072, 2012.
- [5] W. T. Higgins Jr., "A comparison of complementary and Kalman filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 11, no. 3, pp. 321–325, 1975.
- [6] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Transactions of the ASME D*, vol. 83, pp. 95–108, 1961.
- [7] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [8] G. Ligorio and A. M. Sabatini, "A novel kalman filter for human motion tracking with an inertial-based dynamic inclinometer," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 8, pp. 2033–2043, 2015.

- [9] A. M. Sabatini, "Quaternion-based extended Kalman filter for determining orientation by inertial and magnetic sensing," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 7, pp. 1346–1356, 2006.
- [10] W. Li and J. Wang, "Effective adaptive Kalman filter for MEMS-IMU/magnetometers integrated attitude and heading reference systems," *Journal of Navigation*, vol. 66, no. 1, pp. 99–113, 2013.
- [11] Y. S. Suh, "Orientation estimation using a quaternion-based indirect Kalman filter with adaptive estimation of external acceleration," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 12, pp. 3296–3305, 2010.
- [12] R. G. Valenti, I. Dryanovski, and J. Xiao, "A linear Kalman filter for MARG orientation estimation using the algebraic quaternion algorithm," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 2, pp. 467–481, 2016.
- [13] M. Euston, P. Coote, R. Mahony, J. Kim, and T. Hamel, "A complementary filter for attitude estimation of a fixed-wing UAV," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '08)*, pp. 340–345, IEEE, Nice, France, September 2008.
- [14] J. Cockcroft, J. H. Muller, and C. Scheffer, "A complementary filter for tracking bicycle crank angles using inertial sensors, kinematic constraints, and vertical acceleration updates," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4218–4225, 2015.
- [15] C. K. Chui and G. Chen, *Kalman Filtering: with Real-Time Applications*, Springer Science & Business Media, Berlin, Germany, 2008.
- [16] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *Proceedings of the IEEE International Conference on Rehabilitation Robotics (ICORR '11)*, pp. 1–7, Zurich, Switzerland, July 2011.
- [17] Y. Tian, H. Wei, and J. Tan, "An adaptive-gain complementary filter for real-time human motion tracking with MARG sensors in free-living environments," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, no. 2, pp. 254–264, 2013.
- [18] P. Marantos, Y. Koveos, and K. J. Kyriakopoulos, "UAV state estimation using adaptive complementary filters," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 4, pp. 1214–1226, 2016.
- [19] J. F. Vasconcelos, B. Cardeira, C. Silvestre, P. Oliveira, and P. Batista, "Discrete-time complementary filters for attitude and position estimation: design, analysis and experimental validation," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 1, pp. 181–198, 2011.
- [20] R. Mahony, T. Hamel, and J.-M. Pfimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Transactions on Automatic Control*, vol. 53, no. 5, pp. 1203–1218, 2008.
- [21] G. Wahba, "A least squares estimate of satellite attitude," *SIAM Review*, vol. 7, no. 3, p. 409, 1965.
- [22] M. D. Shuster and S. D. Oh, "Three-axis attitude determination from vector observations," *Journal of Guidance and Control*, vol. 4, no. 1, pp. 70–77, 1981.
- [23] F. L. Markley, "Attitude determination using vector observations: a fast optimal matrix algorithm," *The Journal of the Astronautical Sciences*, vol. 41, no. 2, pp. 261–280, 1993.
- [24] Attitude determination and parameter estimation using vector observations and the Singular Value Decomposition, p. 245258, 1988.
- [25] Y. Yang and Z. Zhou, "An analytic solution to Wahbas problem," *Aerospace Science and Technology*, vol. 30, no. 1, pp. 46–49, 2013.
- [26] D. Mortari, "ESOQ: a closed-form solution to the Wahba problem," *The Journal of the Astronautical Sciences*, vol. 45, no. 2, pp. 195–204, 1997.
- [27] F. L. Markley and D. Mortari, "How to estimate attitude from vector observations," *Advances in the Astronautical Sciences*, vol. 103, no. 3, pp. 1979–1996, 2000.
- [28] J. Wu, Z. Zhou, J. Chen, H. Fourati, and R. Li, "Fast complementary filter for attitude estimation using low-cost MARG sensors," *IEEE Sensors Journal*, vol. 16, no. 18, pp. 6997–7007, 2016.
- [29] J. Wu, Z. Zhou, B. Gao, R. Li, Y. Cheng, and H. Fourati, "Fast linear quaternion attitude estimator using vector observations," *IEEE Transactions on Automation Science and Engineering*, no. 99, pp. 1–13, 2017.
- [30] J. Wu, Z. Zhou, R. Li, L. Yang, and H. Fourati, "Attitude determination using a single sensor observation: analytic quaternion solutions and property discussion," *IET Science, Measurement & Technology*, vol. 11, no. 6, article 731, 2017.
- [31] J. Farrell, *Aided Navigation: GPS with High Rate Sensors*, McGraw-Hill Education, New York, NY, USA, 2008.
- [32] A. Makni, H. Fourati, and A. Y. Kibangou, "Energy-aware adaptive attitude estimation under external acceleration for pedestrian navigation," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 3, pp. 1366–1375, 2016.
- [33] D. Titterton and J. Weston, *Strapdown Inertial Navigation Technology*, IET Digital Library, Stevenage, UK, 2nd edition, 2004.
- [34] F. H. Schlee, C. J. Standish, and N. F. TODA, "Divergence in the Kalman filter," *AIAA Journal*, vol. 5, no. 6, pp. 1114–1120, 1967.
- [35] N. A. Carlson, "Federated square root filter for decentralized parallel processes," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 3, pp. 517–525, 1990.
- [36] A. M. Sabatini, "Kalman-filter-based orientation determination using inertial/magnetic sensors: observability analysis and performance evaluation," *Sensors*, vol. 11, no. 10, pp. 9182–9206, 2011.

## Research Article

# Distributed Monocular SLAM for Indoor Map Building

**Ruwan Egodagamage and Mihran Tuceryan**

*Indiana University-Purdue University Indianapolis, Indianapolis, IN, USA*

Correspondence should be addressed to Ruwan Egodagamage; [rjegodag@iupui.edu](mailto:rjegodag@iupui.edu)

Received 28 April 2017; Revised 26 June 2017; Accepted 3 July 2017; Published 10 August 2017

Academic Editor: Jacky C. K. Chow

Copyright © 2017 Ruwan Egodagamage and Mihran Tuceryan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Utilization and generation of indoor maps are critical elements in accurate indoor tracking. Simultaneous Localization and Mapping (SLAM) is one of the main techniques for such map generation. In SLAM an agent generates a map of an unknown environment while estimating its location in it. Ubiquitous cameras lead to monocular visual SLAM, where a camera is the only sensing device for the SLAM process. In modern applications, multiple mobile agents may be involved in the generation of such maps, thus requiring a distributed computational framework. Each agent can generate its own local map, which can then be combined into a map covering a larger area. By doing so, they can cover a given environment faster than a single agent. Furthermore, they can interact with each other in the same environment, making this framework more practical, especially for collaborative applications such as augmented reality. One of the main challenges of distributed SLAM is identifying overlapping maps, especially when relative starting positions of agents are unknown. In this paper, we are proposing a system having multiple monocular agents, with unknown relative starting positions, which generates a semidense global map of the environment.

## 1. Introduction

Utilization of indoor maps is a critical component of accurate indoor tracking when existing infrastructures such as GPS do not work reliably. Therefore, generating such maps with high accuracy for unknown environments becomes critical in the infrastructure of indoor tracking. Furthermore, such maps may be generated partially by different agents moving in and out of an environment, and unifying these independently and partially generated maps into a highly accurate global map is crucial. The map generation and its utilization for localization can be done in many different modalities. Simultaneous Localization and Mapping (SLAM) is one of the main techniques for such map generation. In modern applications, multiple mobile agents may be involved in the generation of such maps, thus requiring a distributed computational framework.

SLAM is a problem that addresses generating a map of an environment and tracking an agent in the environment. These two tasks are interrelated, since an accurate map is necessary to localize the agent precisely, and only a correctly localized

agent can construct a good map. The SLAM problem is also known as the Tracking and Mapping (TAM) problem.

Cameras are becoming a popular choice for SLAM, since they are ubiquitous in smart devices. Furthermore, the smaller form factor and the lower cost of cameras also contribute to this choice. When we use a camera as the input device, the process is called visual SLAM. For visual SLAM, three main types of cameras are used: monocular, stereo, and RGBD. Unlike monocular cameras, stereo and RGBD cameras provide depth data in addition to image data to simplify the initialization and the pose estimation process.

Visual SLAM uses either the direct or the feature-based methods. Direct methods work on the intensity information of images without computing features and generally produce denser maps. Dense maps can be more attractive in certain applications, such as augmented reality, in which a user is interacting with the environment and virtual objects in the environment. It is desirable that this interaction be realistic and seamless. A dense map of the environment makes this interaction possible.

In distributed SLAM, multiple agents perform SLAM in an environment collaboratively. These agents (which are cameras for the purposes of this paper) can enter and exit the environment at any time. If there is a map of the environment, the agents can utilize it to localize themselves in it. If an agent moves in a part of the environment that is not mapped, it can start building the map and localize itself in it as part of the SLAM process. Each agent can do this independently, however, when they are operating in a common environment, it makes sense to use their locally built maps to complement each other. At the same time, they can complete and improve the global map, while helping each other in their respective tasks.

Additionally, using multiple agents to perform SLAM increases the robustness of SLAM process, which makes it more fault tolerant and less vulnerable to catastrophic failures. One of the main challenges in distributed SLAM is to compute map overlaps, especially when agents have no prior knowledge of their relative starting positions. Usually, agents also have limited bandwidth to communicate with each other.

In this paper, we introduce a distributed framework for monocular visual SLAM agents with no prior knowledge of their relative positions.

## 2. Related Work

In a seminal paper Smith et al. [1] introduced a solution to the SLAM problem using extended Kalman filter (EKF-SLAM). In their work, the extended Kalman filter is used to estimate the posterior distribution over agent pose and landmark positions incrementally. However, processing a covariance matrix is a significant challenge as it grows with the number of landmarks. The entire covariance matrix has to be updated even when the system observes one landmark. This severely limits the number of landmarks in EKF-SLAM, typically a few hundred. Furthermore, EKF-SLAM has Gaussian noise assumptions. FastSLAM by Montemerlo et al. [2, 3] addressed above limitations using a Monte Carlo Sampling (particle filter) based approach. Most importantly FastSLAM supported nonlinear process models and non-Gaussian pose distributions. In more recent work, FastSLAM by Cain and Leonessa [4] uses a compressed occupancy grid to reduce the data usage of each particle by 40%. Pei et al. [5] used distributed unscented particle filter to avoid reconfiguring the entire system during vehicle state estimation. Martinez-Cantin and Castellanos in [6] proposed an Unscented Kalman Filter based approach (UKF-SLAM) to support large scale environments.

Davison et al. [7] introduced MonoSLAM, a SLAM method of capturing the path of a freely moving camera (6 Degrees of Freedom) while generating a sparse map. This monocular visual SLAM method worked in a room-sized environment. The map consisted of image patches representing features. Their solution was a combination of EKF-SLAM for estimation and Particle Filtering (PF) for feature initialization. The entire system is initialized by positioning the camera in front of a marker.

Klein and Murray in [8] presented Parallel Tracking and Mapping (PTAM), one of the most significant solutions for

visual SLAM. This robust SLAM solution mainly focused on accurate and fast mapping in a similar environment to MonoSLAM. Its implementation decoupled mapping and localization into two threads. The front-end thread only performs pose estimation and feature tracking while the back-end thread performed mapping and everything else, such as feature initialization and removing unnecessary key frames. A set of sparse point features represented the map. The system is initialized by moving the camera roughly 10 centimeters perpendicular to the optical path. RANSAC [9] and 5-point algorithm [10] initialized the system. A global bundle adjustment (BA) [11] with Levenberg-Marquardt optimization [10] adjusted the pose of all key frames. Furthermore, a local BA changed the pose of a subset of key frames to allow a reasonable rate of exploration.

Although MonoSLAM and PTAM address the same problem, PTAM used BA in contrast to MonoSLAM's incremental approach. BA is heavily used and proven to work well for offline Structure from Motion (SfM). Even though BA is relatively computationally expensive, PTAM and other researchers recently adopted BA for many real-time monocular visual SLAM solutions. Strasdat et al.'s analysis in [12] showed that increasing the number of image features acquired per frame is more beneficial than incorporating information from increased number of closely placed camera frames. They argue that the former increases the accuracy of the motion estimation and a better map estimation for a given computational budget. Their analysis hence favors bundle adjustment techniques over incremental methods for accurate monocular visual SLAM. Moreover, BA helps to increase the number of features on the map, leading to denser maps.

Scale drift is one of the biggest challenges in monocular visual SLAM. Strasdat et al. [13] introduced a pose graph optimization technique that corrects the scale drift at loop closures. Their method handled large looped trajectories well.

The work by DTAM by Newcombe et al. [14] and LSD-SLAM by Engel et al. [15, 16] utilize image pixel intensities directly instead of computed features for SLAM. Their systems generate dense or semidense maps of the environment. Furthermore, these direct methods are more robust to motion blur of images.

During the SLAM process, an agent might revisit the same location in multiple instances. Error accumulation can lead this to go unnoticed. The solution for this problem is referred to as loop closing. This could be done using appearance based image-to-image, map-to-map, or map-to-image matching approaches. The survey from Williams et al. [17] concludes with positive remarks on map-to-image approaches.

*2.1. Distributed SLAM.* One of the challenges in generating a globally consistent map is identifying map overlaps of agents. It is relatively easier to determine map overlaps if all of the agent relative poses are known at all times. For example, Nettleton et al. [18] used global positioning sensors (GPS) to detect agent locations. When the agent position is known, it is only a matter of doing a proximity check between agent trajectories to detect their map overlaps. However, location

sensors like GPS are not always available, and they do not do well in indoors, nor in underwater vehicles.

The relative transformations between coordinate systems of agent maps can be computed if the starting position of each agent is known. Paull et al. [19] initialized all agents from known locations. Next, agents performed SLAM and estimated their new locations, while at the same time communicating their locations to each other. Given that the agents already knew the transformation between their maps, they were able to easily determine map overlaps, similar to the case in having location sensors.

When these agent relative locations are unknown, the distributed SLAM problem becomes more challenging. In some contributions, agents continued to build local maps until they saw each other. Howard et al. [20] proposes a method in which each agent would be able to detect other agents. Agents use these coincidental encounters to find their relative locations. Dieter et al. in [21] presents a method where each agent is actively seeking other agents in the environment to find relative locations between them. These methods either require special sensors to be seen by each other or to actively seek each other.

Some methods heavily depend on a central node. Zou and Tan in [22] allowed cameras to move independently in a dynamic environment. However, all of their cameras were initialized from the same scene and connected to the same computer. Although their cameras were distributed, all frames were processed at the same time in their SLAM process. This tightly couples agents, since agents do not possess any knowledge of the environment individually.

The multiagent system by Forster et al. in [23] used a centralized ground station for mapping, loop closure detection, and map merging. However, relying on a central agent is highly prone to failures, especially when the central node fails.

In [24], agents used a master-slave approach where the slave is always in the master agent's map to maintain a map overlap. Their method restricts the free movement of the slave agent.

Williams et al. in [25, 26] introduced a method to construct a global map from a multiagent system using a Constrained Local Submap Filter (CLSf). In their method, overlaps between the local and global maps are determined using a Maximal Common Subgraph (MCS) method. Kin and Newman in [27] use a visual similarity matrix to determine relative agent locations. A subsequence of visually similar images is detected from the images captured in each agent. Cunningham et al. in [28, 29] formulate the distributed SLAM problem using a graphical model. In their fully decentralized system, each agent maintained a consistent local map, augmented with information shared in a neighborhood of agents. In [30], authors proposed a fully distributed solution for the data association problem.

In the system proposed in [30], local feature matches are propagated through the low-bandwidth communication network. This method helps agents to find global correspondences with other agents with no direct connections. Work done in [18] discusses most informative features to transmit, to reduce bandwidth requirements.

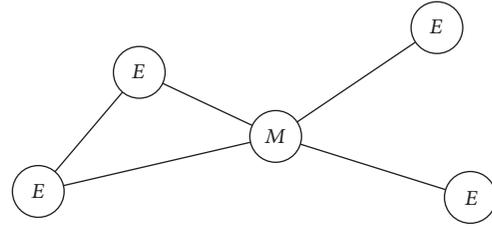


FIGURE 1: The network of nodes: the exploring nodes ( $E$ ) are connected to a monitoring node ( $M$ ). Some exploring nodes are connected with each other.

Our proposed framework performs distributed SLAM with no knowledge of the initial agent locations. Furthermore, the agents do not get their location directly from sensors like GPS. Instead, they estimate the location using a visual SLAM process. Moreover, the framework does not rely on a central agent, but rather a network of monitoring agents that look for map overlaps of SLAM agents.

We used the experimental framework for distributed SLAM we introduced in [31], during the development of this framework.

### 3. Materials and Methods

**3.1. Nodes of the Distributed Framework.** Our distributed SLAM framework consists of two types of nodes, the *exploring node* and the *monitoring node*. Each node is deployed in its own physical machine. At any given time, the framework has at least one monitoring node and an arbitrary number exploring nodes. Each node is identified using a global unique identifier.

A connection between nodes is made when nodes are required to communicate with each other over the network. As shown in Figure 1, the exploring nodes are connected to a monitoring node. Furthermore, two exploring nodes become connected when their maps overlap with each other.

**3.1.1. Exploring Node.** Each exploring node performs a semi-dense visual SLAM by using a camera as the only sensor, based on the work by [16]. Our choice is based on the fact that denser maps describe the environment in more detail, compared to the sparse, feature-based maps. Denser maps enable better interaction with the environment, especially in AR applications. This also means exploring nodes that communicate more data, compared to feature-based methods. Each exploring node maintains a set of key frames and a pose graph to represent the map. It periodically sends out its map information to the monitoring node, as well as to the other exploring nodes to which it is connected. Furthermore, it processes incoming commands from the monitoring node.

**3.1.2. Monitoring Node.** The monitoring node's responsibilities include map overlap detection between the exploring nodes and loop closure detection in each exploring node. It maintains a feature store in which all *salient* features are stored. Features from all incoming key frames are matched against the feature store to identify map overlaps and loop

```

(1) procedure GETUNIQUEID (key_frame_id, node_id)
(2)   id ← SHIFTLLEFT(node_id, 20)
(3)   id ← id + keyframe_id
(4)   return id                                ▷ A globally unique identifier
(5) end procedure

```

ALGORITHM 1: Unique identifier for a key frame.

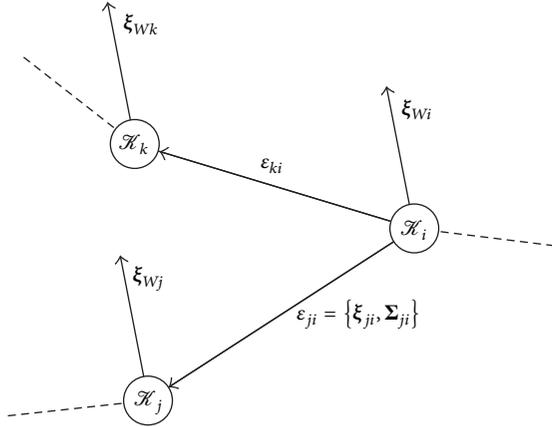


FIGURE 2: Pose graph with key frames and similarity transform constraints.

closures. The monitoring node uses a graph, which will be referred to as the *fusion graph* in this paper, to prioritize and issue commands to merge overlapping maps.

In advanced configurations of our proposed distributed framework, there could be multiple monitoring nodes. Ideally, each monitoring node connects to overlapping exploring nodes. In practice, monitoring nodes may move exploring nodes among themselves dynamically to minimize the number of overlapping exploring node clusters. Map overlap detection between two exploring nodes belonging to different monitoring nodes is accomplished by sharing features between monitoring nodes. In this paper, our experiments are limited to a single monitoring node configuration.

**3.2. Map of the Environment by Exploring Node.** As shown in Figure 2, the exploring node maintains a map of the environment using multiple key frames and a pose graph.

**3.2.1. Key Frames.** The  $i$ th key frame,  $\mathcal{K}_i$ , consists of an *absolute pose*  $\xi_{W_i} \in \mathbb{R}^7$ , an image  $I_i$ , a map containing  $z$  coordinate reciprocals corresponding to nonnegligible intensity gradient pixels  $D_i$  (an inverse depth map), an inverse depth variance map  $V_i$ , and a list of features  $F_i$ . The absolute pose is encoded with a translation, along with orientation and scale parameters using a quaternion. The elements of  $\xi_{ji}$  are the three components of the translation and the four components of the quaternion representing the rotation. The scale is represented using the magnitude of the quaternion. When  $\mathcal{K}_i$  is first introduced into the pose graph, the features of  $\mathcal{K}_i$  are

computed. In  $\mathcal{K}_i$ ,  $i$  corresponds to a 32-bit globally unique identifier computed using Algorithm 1. Figure 3 contains a visual representation of two different key frames.

We used SURF [32] features and SIFT [33] descriptors in our framework. Because we computed features only for the key frames, the added computational cost that resulted did not adversely affect the real-time performance.

**3.2.2. Pose Graph.** Pose graph edges  $\varepsilon_{ji}$  contain similarity transformations  $\xi_{ji}$  and  $\Sigma_{ji}$  constraints. Here  $\xi_{ji} \in \mathbb{R}^7$  and  $\Sigma_{ji}$  are relative pose transformations and corresponding covariance matrix between  $i$ th and  $j$ th the key frames, respectively.

Both absolute pose  $\xi_{W_i}$  and similarity transformation  $\xi_{ji}$  are encoded with a translation (three components) and orientation with scale using a quaternion (four components).

**3.2.3. Generating and Updating the Map.** The SLAM process simultaneously tracks the camera against the current key frame  $\mathcal{K}_i$  and improves its  $D_i$  and  $V_i$  based on its new observations. Once the camera deviates significantly from the  $\mathcal{K}_i$ , either a new key frame is created or, if available, an existing key frame is selected from the map. Next, if a new key frame is created, the previous key frame used for tracking is inserted into the pose graph. The pose graph is continuously optimized in the background. More information on the LSD-SLAM process is found in [15].

**3.2.4. Identifying Salient Features.** To determine the saliency of a feature, first the feature is filtered for its  $V_i(X_p)$ , where  $X_p$  is the location feature found. The  $p$ th feature in  $\mathcal{K}_i$  should satisfy

$$V_i(X_p) < T \times D_i(X_p)^2, \quad (1)$$

where  $T$  is a threshold computed empirically. Only salient features are kept as  $F_i$ . We experimented with different values for  $T$  to minimize the number of features exchanged, while still achieving sufficient map overlap detection. We found 0.001 to be a good value with satisfactory results.

**3.2.5. Sending Salient Features to Monitoring Node.** For every salient feature in  $F_i$ , the corresponding 3D location  $X_p$  and the descriptor  $d_p$  are computed. Next, the key frame identifier  $i$ , the salient features  $(X_p, d_p)_i$ , and the pose  $\xi_{W_i}$  are sent to the monitoring node.

The communication process between the nodes is explained in more detail in Section 3.6.2.

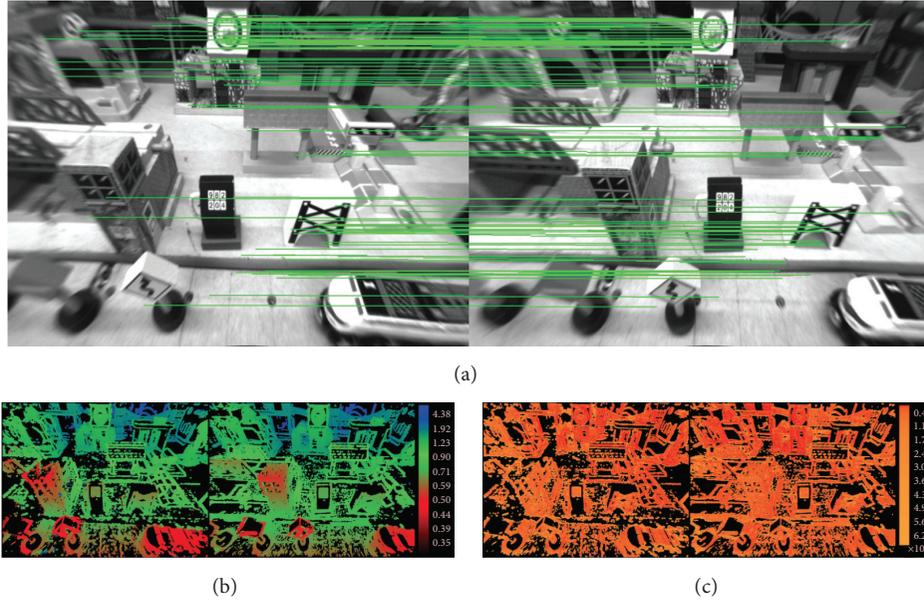


FIGURE 3: We show the matched features between key frames  $\mathcal{K}_i$  and  $\mathcal{K}_j$  superimposed on the images  $I_i$  and  $I_j$  (a). We also show the pseudo-color encoded  $D_i$  and  $D_j$  (b) and pseudo-color encoded  $V_i$  and  $V_j$  (c).

**3.3. Map Overlap Detection in Monitoring Node.** Exploring nodes of our distributed framework do not know their relative pose at the beginning. The monitoring node is responsible for detecting map overlaps and computing corresponding relative pose between nodes.

**3.3.1. Feature Store.** Each entry in the feature store contains a feature descriptor  $d_p$ , key frame identifier  $i$ , 3D feature location  $X_p$ , and key frame pose  $\xi_{Wi}$ . Every incoming feature descriptor is matched against the entries in the feature store, to identify common features between two key frames. We used the FLANN [34] method for feature matching. We used a search radius of 150, which was calculated empirically. We divide each key frame into a grid of 16 equal sized cells. If features are matched in at least 10 cells with another key frame  $\mathcal{K}_j$ , it is concluded that there is an overlap between key frames  $\mathcal{K}_i$  and  $\mathcal{K}_j$ . Our approach expects the indoor environment to contain sufficient textured regions of overlap to function properly. It fails if the overlapping region contains only scenes like texture-less walls.

If overlapping key frames belong to the same exploring node, it is considered that a loop closure is found. Otherwise, matching information contributes to the *fusion graph*.

**3.3.2. Fusion Graph.** All available exploring nodes are represented as vertices in the fusion graph as shown in Figure 4. Assume there is an overlap between key frames  $\mathcal{K}_r$  and  $\mathcal{K}_s$  and  $\mathcal{K}_r \in e_i$  and  $\mathcal{K}_s \in e_j$ , where  $e_i$  represent key frames in  $i$ th exploring node. Then, the fusion graph contains an edge between  $e_i$  and  $e_j$ . The number of features matched between  $e_i$  and  $e_j$  is represented using  $c_{ij}$  as shown in Figure 4. Note that the edge between  $e_i$  and  $e_j$  could represent matching features between many different key frame pairs. If the direction of

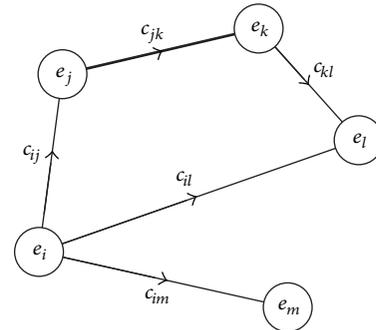


FIGURE 4: The fusion graph showing exploring nodes ( $e_i$ ) and the number of matching features ( $c_{ij}$ ) as the weight of each edge. In this example,  $c_{jk}$  is higher than other edges (indicated by the thicker edge), so  $e_j$  and  $e_k$  are merged first. Furthermore, the map merging process is initialized by sending  $e_j$ 's map to  $e_k$  following the direction of the edge.

the edge is  $e_j \rightarrow e_k$ , map merging process is initialized from  $e_j$ . As shown in Figure 5, first  $e_j$  sends the map to  $e_k$ . Then,  $e_k$  merges the map and sends its original map to  $e_j$ .  $e_j$  merges the received map and notifies the monitoring node about the completion of map merging process. Assume that the fusion graph edge having the largest  $c_{ij}$  satisfies

$$\max(c_{ij}) > m, \quad (2)$$

where  $m$  is an empirical threshold. Then the monitoring node concludes that a map overlap exists between exploring nodes  $e_i$  and  $e_j$ . Empirically, 120 shared features are found to be a good value for  $m$ . Next, the rigid transformation between  $e_i$

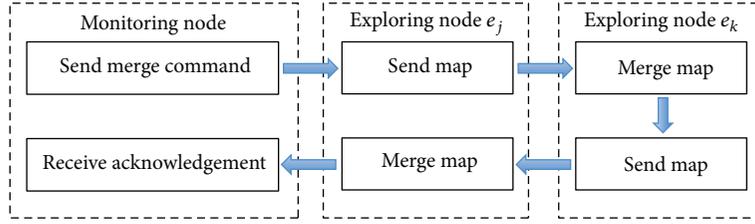


FIGURE 5: The map merging process of the fusion graph edge  $e_j \rightarrow e_k$ .

and  $e_j$ ,  $\xi_{ji}$ , is computed using a Singular Value Decomposition (SVD) based on the least squares method [35]. Similar to the absolute pose representation  $\xi_{ji}$  is encoded using 3 components of translation and 4 components of quaternion. The scale is initially assumed to be 1 and a proper value is estimated later, during the map merging followed by pose graph optimization in each exploring node.  $X_p$  of all relevant features between  $e_i$  and  $e_j$  are used for the computation. The RANSAC algorithm [9] is used to make the computation robust to outliers. Figure 3 shows a set of matched features between two key frames,  $\mathcal{K}_i$  and  $\mathcal{K}_j$ .

**3.3.3. Issuing Commands to Exploring Nodes.** A map merge command is issued to exploring nodes  $e_i$  and  $e_j$ . The command contains the relative pose  $\psi_{ji}$  between two nodes. The command also contains the key frame correspondences used to compute the relative pose between  $e_i$  and  $e_j$ . Similarly, a *loop closure* command is issued to an exploring node  $e_s$ , when both overlapping key frames  $\mathcal{K}_i$  and  $\mathcal{K}_j$  belong to  $e_s$ . The command contains the relative pose  $\xi_{ji}$  between key frames, which is also computed using the same least squares method [35].

The communication process is explained in more detail in Section 3.6.2.

**3.4. Merging Maps of Two Exploring Nodes.** First, as shown in Figure 1, a connection is created between two exploring nodes. Once the connection is made, each exploring node sends its map to its counterpart. Once the map is received, the key frame correspondences found in the *map merge* are directly transformed into new constraints between pose graphs of  $e_i$  and  $e_j$ . The similarity transformation of the constraint is computed using key frame pose  $\xi_{wk}$  and relative pose  $\xi_{ji}$  between exploring nodes.

Figure 6 shows how  $e_i$  and  $e_j$  were generating their own maps before merging. Figure 7 shows the resulting merged map for  $e_i$ . Once map merging is complete, each exploring node listens to its counterpart for new key frames and the pose graph, to incrementally update its map.

**3.5. Loop Closure.** In most instances, completing smaller loop closures increases the robustness of tracking. Completing large loop closures, however, has more impact in generating an accurate map. Direct semidense SLAM operations alone do not support large loop closures.

Upon receiving a loop closure command with  $\xi_{ji}$ , the exploring node checks whether  $\mathcal{K}_i$  and  $\mathcal{K}_j$  are consecutive

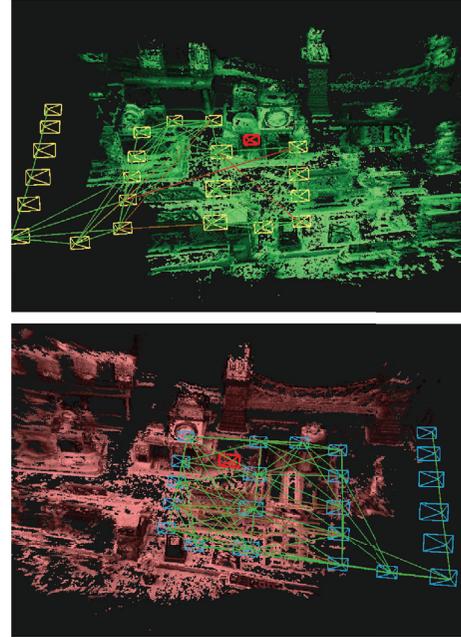


FIGURE 6: Map generation process of two exploring nodes. Each exploring node has its own coordinate system. Relative transformations between coordinate systems are initially not known.

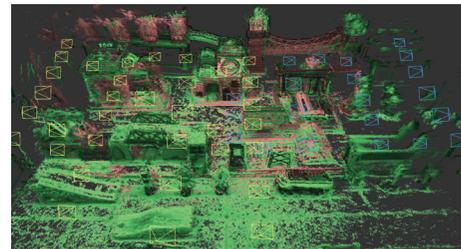


FIGURE 7: Resultant map of an exploration node after the map merging process. The exploring node's map and key frames are shown in green and yellow, respectively. The map and key frames received from the other node are shown in pink and blue, respectively. Constraints of the pose graph are not shown here to avoid too much clutter in the figure.

key frames in the pose graph. If that is the case, we discard the loop closure command since  $\mathcal{K}_j$  was constructed using  $\mathcal{K}_i$  and already has a better estimate for the edge  $e_{ji}$ . Otherwise,

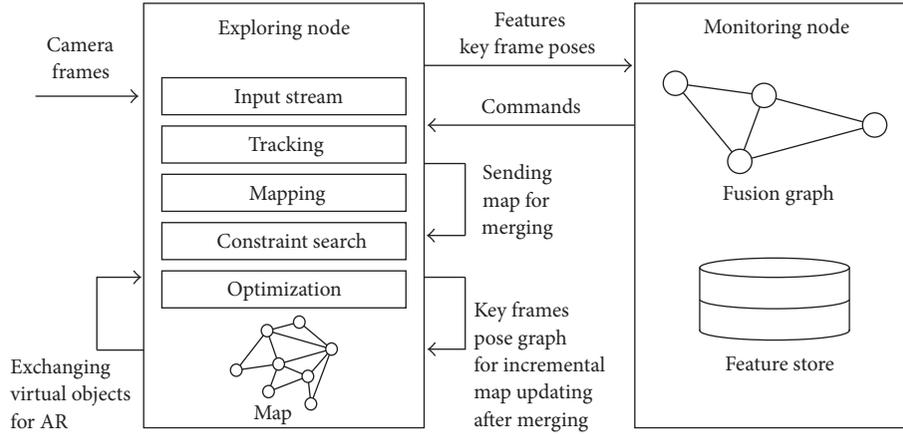


FIGURE 8: The distributed framework. In the figure, the arrows looping back to the exploring node rectangle represent communication between two exploring nodes.

it inserts the new edge and completes the loop closure process by performing another iteration of pose graph optimization.

**3.6. System Implementation.** We use the Robot Operating System (ROS) infrastructure to implement the distributed SLAM framework described in this paper [36]. A ROS node is responsible for performing computations. ROS also provides a message passing communication framework between nodes. Nodes in our framework are implemented as ROS nodes.

In its communication framework, ROS provides named communication buses called *topics*. Multiple nodes can publish messages to a topic while multiple subscribed nodes could receive them. ROS nodes can communicate with each other peer-to-peer via topics. In our framework, communication channels are implemented using ROS topics.

**3.6.1. Components of the Distributed Framework.** Figure 8 shows components of the distributed framework and the communications between nodes. The exploring node consists of five main modules: input stream, tracking, mapping, constraint search, and optimization modules. Each of these modules runs in its own thread and it contains the map.

The *input stream* module accepts all incoming messages including image frames, key frames, pose graph, map, and commands. All image frames are transferred to the tracking module. Key frames, pose graph, and map are transferred to the optimization module so that they can be merged into the map before an optimization iteration. Commands are processed in the input stream module itself.

The *tracking* module accepts the new frame from input stream module and tracks it against the current key frame. If the current key frame can no longer be used to track the current frame, a new key frame is created. The old key frame will be added to the map by the *mapping* module. The *constraint search* module is used to recover from tracking failures. The *optimization* module continuously optimizes the pose graph in the background.

The monitoring node maintains the feature store and the fusion graph as explained in Section 3.3.

**3.6.2. Communication between Nodes.** The distributed framework provides multiple communication channels for nodes. These communication channels are shown as arrows in Figure 8.

Upon creating new key frames, exploring nodes send salient features and the absolute pose of the key frame through the *features* channel. The monitoring node receives them and processes them to issue commands through the *commands* channel. The command could be either a *merge* command or a *loop closure* command.

When an exploring node receive a *merge* command, it creates multiple channels with the other exploring node. The *map* channel is used to exchange a map between each other. This channel ceases to exist once the map is transferred. And *key frames* and *pose graph* channels are created between these nodes.

For every new key frame, its information is written into the *key frames* channel. After every pose graph optimization, the pose graph information is written into *pose graph* channel. All exploring nodes that are using these channels incrementally update their maps after merging.

## 4. System Evaluation and Discussion

**4.1. Public Datasets.** To evaluate our framework, we need a monocular visual SLAM dataset, with multiple trajectories covering a single scene. We considered publicly available datasets, but they did not satisfy our requirements. For example, the dataset EuRoC [37] contains pure rotations, which did not work well with the monocular SLAM approach we used. The dataset Kitti [38] is mainly a stereo dataset. Even when we considered a single camera, the direct monocular SLAM process failed since the camera motion is along the optical axis. The TUM-Mono [39] dataset does not provide the ground truth for all frames and is primarily suitable for

TABLE 1: DIST-Mono dataset.

Dataset	Path	Initial camera rotation	Total travel (mm)
S01-A-0	Path A	0°	3706
S01-A-P20	Path A	20° CW	3706
S01-B-0	Path B	0°	3706
S01-B-N20	Path B	20° CCW	3706
S01-C-0	Path C	0°	3080



FIGURE 9: Experimental setup showing a camera mounted on a CNC machine allowing us to capture ground truth information. Camera mounted on a CNC machine.

evaluating single agent SLAM. That said, we created the DIST-Mono dataset to evaluate our framework. We also made it publicly available (<http://slam.cs.iupui.edu>).

**4.2. Experimental Setup.** Our experimental setup is designed to precisely define the ground truth of a camera motion. As shown in Figure 9 we mounted a Point Grey Firefly MV global shutter camera onto a Computer Numeric Controller (CNC) machine. We also prepared a 1 m × 1.5 m scene containing wooden objects. We then moved the camera along a path for about four minutes each time, while capturing its location ground truth periodically.

Our in-house built, 3-axis CNC machine is controlled using an open-source controller called TinyG. The controller converts the provided trajectory from the gcode file format into linear synchronized movements along  $x$ ,  $y$ , and  $z$  axes. The maximum travel volume of the machine is 1 m × 1 m × 0.3 m ( $x \times y \times z$ ). We also prepared a 1 m × 1.5 m scene containing wooden objects. The scene is uniformly lit by two 4-foot long LED tube lights.

To collect datasets, we moved the camera at a speed of 25 mm/s, along a path for about four minutes each time, while capturing its location ground truth periodically. We captured 640 × 480 resolution camera frames at 60 Hz and ground truth at 40 Hz. The CNC machine has 0.2 mm accuracy in all three axes. We developed an open-source ROS node (<http://github.com/japzi/rostinyg>) to capture the ground truth from the TinyG CNC controller.

During experimentation we played back the datasets at twice the speed it was recorded (2x), effectively making the camera movement to be 50 mm/s.

**4.3. The DIST-Mono Dataset.** The dataset consists of five subdatasets. We defined three camera motion paths: Path A, Path B, and Path C. All of these paths are on a plane

TABLE 2: Experiments and their absolute translation RMSE against ground truth.

Experiment	Datasets	RMSE (m)
Experiment 1	S01-A-0, S01-B-0	0.0136
Experiment 2	S01-A-0, S01-B-N20	0.0192
Experiment 3	S01-B-0, S01-C-0	0.0097
Experiment 4	S01-A-0, S01-C-0	0.0121

slanted above the scene as shown in Figure 10(a). These paths have roughly 10% overlap with each other and three different starting points per path. We generated two datasets using Path A by rotating the camera around its  $z$ -axis. In S01-A-0, the camera optical axis and scene  $Y$  axis are on a vertical plane. In S01-A-P20, we rotated the camera around its  $y$ -axis by 20°. This is illustrated in Figure 10(b). Similarly, we created datasets S01-B-0, S01-B-N20, and S01-C-0 as shown in Table 1.

**4.4. System Evaluation.** As an experiment, we deployed two exploring nodes and one monitoring node in three different machines. One exploring node processed the dataset SCENE-A-0 and the other the dataset SCENE-B-N20. After map merging, each exploring node exported its key frame poses in TUM dataset [40] pose format. Most importantly, these poses contain key frames from both exploring nodes. We then computed the absolute translation RMSE [40] against the ground truth. To support the nondeterministic nature of the distributed framework, we ran the experiment five times and the median result is recorded. Similarly we performed two more experiments with other combinations of datasets as shown in Table 2. Given the fact that monocular visual SLAM systems do not capture the scale, we manually calculated that in all experiments to minimize the RMSE error.

Figure 11 shows how estimated key frame positions are compared against ground truth in experiment 3. The red circles in the figure display the estimated key frame position, whereas the black circles display the ground truth of said key frame position. Red lines show the difference between the estimated and the ground truth positions of the key frame.

**4.5. AR Application: Adding and Viewing Virtual Objects.** We developed an AR application to test our framework. We added an AR window to each exploring node. The AR window allows users to add virtual objects (simple cube in our example) into its map. This allows us to demonstrate

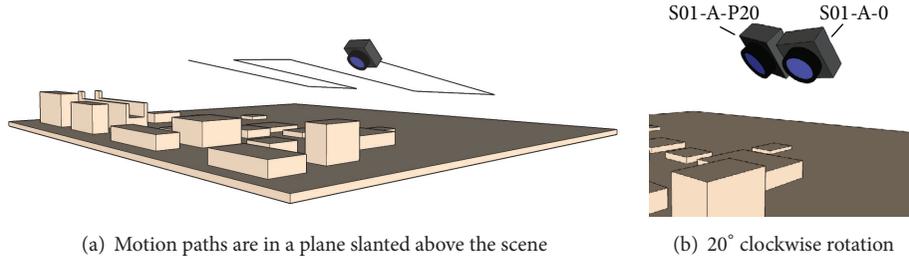


FIGURE 10: Camera motion and its initial rotation for datasets.

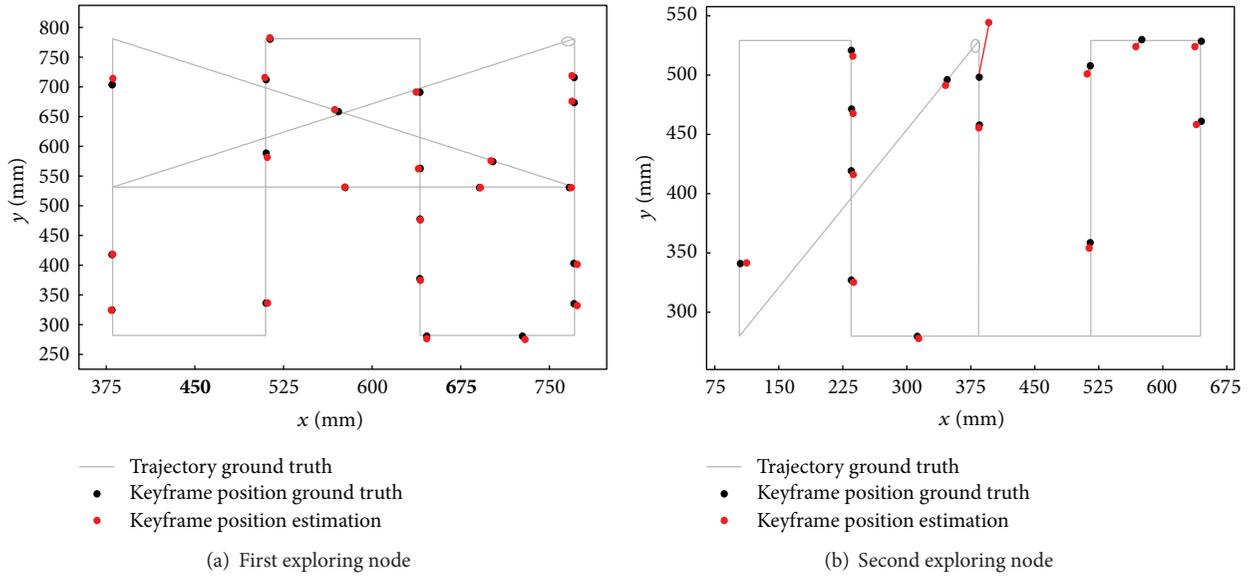


FIGURE 11: Key frame position estimation against ground truth.

the collaborative AR potential of the distributed SLAM framework, in which (i) each agent is able to view the augmented scene from its viewpoint and (ii) if it is in an unexplored part of the scene, generate its own local map and contribute it to the global map.

We also added a relevant channel to share the virtual object information between exploring nodes. Given that the relative transformation between nodes is known for connected exploring nodes, these cubes are placed correctly in the map. Figure 12 shows AR windows of two exploring nodes and two interactively added cubes.

**4.6. Discussion.** Our framework relies on image features for map overlap detection. As explained in Section 3.2.5, features  $(X_p, d_p)_i$  and the pose  $\xi_{W_i}$  of each  $\mathcal{X}_i$  are processed by monitoring nodes for this purpose. In addition to the current semidense monocular SLAM method, many key frame based SLAM methods using stereo or RGBD sensors could easily be adapted into exploring nodes, given the fact that those methods can produce required data with minimal effort. However, other sensors, such as LiDAR, do not provide the data appearance based features required in monitoring node.

Therefore, they require a major change in the monitoring node to function properly.

## 5. Conclusions

We introduced a distributed monocular SLAM framework that identifies map overlaps based on an appearance based method. Most importantly the framework computes relative transformations of local maps of SLAM nodes, even when their relative starting positions are unknown. We demonstrated empirically that the distributed framework we developed successfully generated the map using multiple agents and localized them in the environment with little amount of error. We achieved a pose location RMSE between 0.0097 m and 0.0136 m for experiments that were conducted in a  $1\text{ m} \times 1.5\text{ m}$  scene. Each node traveled about 4 m on average. We developed an empirical set up that generated the data set with associated ground truth to be used for extensive evaluation and validation of the distributed SLAM method. This data set has been made publicly available for other researchers to use. We also developed an augmented

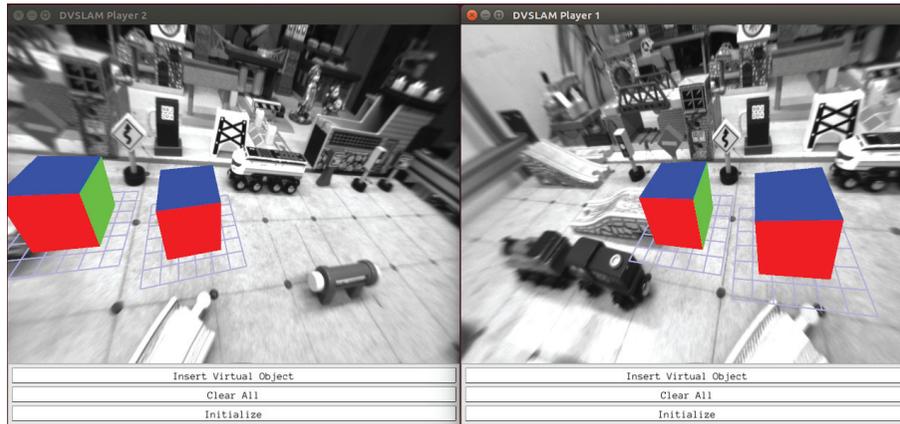


FIGURE 12: Same set of virtual objects is viewed from two different exploring nodes.

reality application to showcase how two nodes can use our framework to interact with the shared global map.

Our next steps would be to improve exploring node's SLAM process by incorporating features in pose graph optimization. That would help greatly in supporting public datasets as well. Furthermore, we will evaluate ORB descriptors instead of SIFT descriptors to improve performance and reduce the network bandwidth usage.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

## References

- [1] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous Robot Vehicles*, I. J. Cox and G. T. Wilfong, Eds., pp. 167–193, Springer-Verlag, New York, NY, USA, 1990.
- [2] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "Fastslam: A factored solution to the simultaneous localization and mapping problem," in *Proceedings of the the AAAI National Conference on Artificial Intelligence (AAAI '02)*, pp. 593–598, 2002.
- [3] M. Montemerlo, S. Thrun, D. Roller, and B. Wegbreit, "Fastslam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI '03)*, pp. 1151–1156, August 2003.
- [4] C. Cain and A. Leonessa, "FastSLAM Using Compressed Occupancy Grids," *Journal of Sensors*, vol. 2016, Article ID 3891865, 2016.
- [5] F.-J. Pei, H.-Y. Li, and Y.-H. Cheng, "An improved FastSLAM system based on distributed structure for autonomous robot navigation," *Journal of Sensors*, vol. 2014, Article ID 456289, 9 pages, 2014.
- [6] R. Martinez-Cantin and J. A. Castellanos, "Unscented SLAM for large-scale outdoor environments," in *Proceedings of the IEEE IRS/RSJ International Conference on Intelligent Robots and Systems (IROS '05)*, pp. 3427–3432, Edmonton, Canada, August 2005.
- [7] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: real-time single camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.
- [8] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *Proceedings of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '07)*, pp. 225–234, Nara, Japan, November 2007.
- [9] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the Association for Computing Machinery*, vol. 24, no. 6, pp. 381–395, 1981.
- [10] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, Cambridge University Press, 2nd edition, 2004.
- [11] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Undle Adjustment—A Modern Synthesis," in *Proceedings of the Vision algorithms: theory and practice*, vol. 2000 of *Lecture Notes in Computer Science*, pp. 298–372, Springer Berlin Heidelberg, Corfu, Greece.
- [12] H. Strasdat, J. M. M. Montiel, and A. J. Davison, "Real-time monocular SLAM: why filter?" in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 2657–2664, IEEE, May 2010.
- [13] H. Strasdat, J. M. M. Montiel, and A. Davison, "Scale drift-aware large scale monocular slam," in *Proceedings of the Robotics: Science and Systems*, Zaragoza, Spain, June 2010.
- [14] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proceedings of the 2011 IEEE International Conference on Computer Vision, ICCV 2011*, pp. 2320–2327, esp, November 2011.
- [15] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-Scale Direct monocular SLAM," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8690, no. 2, pp. 834–849, 2014.
- [16] J. Engel, J. Sturm, and D. Cremers, "Semi-dense visual odometry for a monocular camera," in *Proceedings of the 14th IEEE International Conference on Computer Vision (ICCV '13)*, pp. 1449–1456, Sydney, Australia, December 2013.
- [17] B. Williams, M. Cummins, J. Neira, P. Newman, I. Reid, and J. Tardós, "A comparison of loop closing techniques in monocular SLAM," *Robotics and Autonomous Systems*, vol. 57, no. 12, pp. 1188–1197, 2009.

- [18] E. Nettleton, S. Thrun, H. Durrant-Whyte, and S. Sukkarieh, "Decentralised slam with low-bandwidth communication for teams of vehicles," in *Proceedings of the Field and Service Robotics Conference (FSR '06)*, vol. 24, pp. 179–188, 2006.
- [19] L. Paull, G. Huang, M. Seto, and J. J. Leonard, "Communication-constrained multi-AUV cooperative SLAM," in *Proceedings of the 2015 IEEE International Conference on Robotics and Automation, ICRA 2015*, pp. 509–516, usa, May 2015.
- [20] A. Howard, L. E. Parker, and G. S. Sukhatme, "The SDR experience: Experiments with a large-scale heterogeneous mobile robot team," *Springer Tracts in Advanced Robotics*, vol. 21, pp. 121–130, 2006.
- [21] F. Dieter, K. Jonathan, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart, "Distributed multirobot exploration and mapping," *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1325–1338, 2006.
- [22] D. Zou and P. Tan, "CoSLAM: Collaborative visual SLAM in dynamic environments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 2, pp. 354–366, 2013.
- [23] C. Forster, S. Lynen, L. Kneip, and D. Scaramuzza, "Collaborative monocular SLAM with multiple Micro Aerial Vehicles," in *Proceedings of the 2013 26th IEEE/RSJ International Conference on Intelligent Robots and Systems: New Horizon, IROS 2013*, pp. 3963–3970, jpn, November 2013.
- [24] J. Solà, A. Monin, M. Devy, and T. Vidal-Calleja, "Fusing monocular information in multicamera SLAM," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 958–968, 2008.
- [25] S. Bernard Williams, *Efficient solutions to autonomous mapping and navigation problems [Ph.D. thesis]*, The University of Sydney, New South Wales, Australia, 2001.
- [26] S. B. Williams, G. Dissanayake, and H. Durrant-Whyte, "Towards multi-vehicle simultaneous localisation and mapping," in *Proceedings of the Robotics and Automation, IEEE International Conference (ICRA '02)*, pp. 2743–2748, usa, May 2002.
- [27] H. Kin and P. Newman, "Multiple map intersection detection using visual appearance," in *Proceedings of the International conference on computational intelligence, robotics and autonomous systems (CIRAS' 05)*, 2005.
- [28] A. Cunningham, M. Paluri, and F. Dellaert, "DDF-SAM: Fully distributed SLAM using constrained factor graphs," in *Proceedings of the 23rd IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010*, pp. 3025–3030, twn, October 2010.
- [29] A. Cunningham, V. Indelman, and F. Dellaert, "DDF-SAM 2.0: Consistent distributed smoothing and mapping," in *Proceedings of the 2013 IEEE International Conference on Robotics and Automation, ICRA 2013*, pp. 5220–5227, deu, May 2013.
- [30] E. Montijano, R. Aragues, and C. Sagüés, "Distributed data association in robotic networks with cameras and limited communications," *IEEE Transactions on Robotics*, vol. 29, no. 6, pp. 1408–1423, 2013.
- [31] R. Gamage and M. Tuceryan, "An experimental distributed framework for distributed Simultaneous Localization and Mapping," in *Proceedings of the 2016 IEEE International Conference on Electro Information Technology, EIT 2016*, pp. 665–667, usa, May 2016.
- [32] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.
- [33] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [34] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *Proceedings of the International Conference on Computer Vision Theory and Application (VISSAPP '09)*, pp. 331–340, Lisboa, Portugal, February 2009.
- [35] B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *Journal of the Optical Society of America A*, vol. 4, no. 4, pp. 629–642, 1987.
- [36] M. Quigley, K. Conley, B. Gerkey et al., "ROS: An Open-Source Robot Operating System," in *Proceedings of the ICRA workshop on open source software (ICRA '09)*, vol. 3, 5 pages.
- [37] M. Burri, J. Nikolic, P. Gohl et al., "The EuRoC micro aerial vehicle datasets," *International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.
- [38] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 3354–3361, Providence, RI, USA, June 2012.
- [39] J. Engel, V. Usenko, and D. Cremers, "A Photometrically Calibrated Benchmark For Monocular Visual Odometry," 2016, <https://arxiv.org/abs/1607.02555>.
- [40] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proceedings of the 25th IEEE/RSJ International Conference on Robotics and Intelligent Systems (IROS '12)*, pp. 573–580, October 2012.

## Research Article

# Integrated SINS/WSN Positioning System for Indoor Mobile Target Using Novel Asynchronous Data Fusion Method

Hai Yang,<sup>1</sup> Wei Li,<sup>2</sup> Chengming Luo,<sup>3</sup> Jinyao Zhang,<sup>2</sup> and Zhuoyin Si<sup>2</sup>

<sup>1</sup>*School of Mechatronic Engineering, Southwest Petroleum University, Chengdu, Sichuan 610500, China*

<sup>2</sup>*School of Mechatronic Engineering, China University of Mining and Technology, Xuzhou, Jiangsu 221116, China*

<sup>3</sup>*College of Internet of Things Engineering, Hohai University, Changzhou, Jiangsu 213022, China*

Correspondence should be addressed to Wei Li; [cmecumt512@yahoo.com](mailto:cmecumt512@yahoo.com)

Received 21 March 2017; Accepted 13 June 2017; Published 20 July 2017

Academic Editor: Mohannad Al-Durgham

Copyright © 2017 Hai Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

According to the asynchronous transmission of data for the SINS/WSN integrated positioning system, this paper proposes a novel asynchronous data fusion method using Unscented Kalman Filter for SINS/WSN integrated positioning system based on indoor mobile target. The state equation of the integrated system is built with the motion characteristic of mobile target. The pseudo measurement equation is built based on the time sequence of SINS/WSN measured value through detecting the measurement of WSN in every fusion period. Considering that the improved state-space model, comprised of the state equation and pseudo measurement equation, is the nonlinear equations, the Unscented Kalman Filter is applied to estimate the state value of the state-space model. Hence the asynchronous data fusion method for SINS/WSN integrated positioning system can be achieved. Simulation results and experimental tests show that the positioning system with proposed asynchronous data fusion algorithm performs feasibility and stability under circumstances of the asynchronous time, and it is superior to the traditional asynchronous data fusion and synchronous data fusion methods.

## 1. Introduction

Indoor localization of mobile target is playing an increasingly important role on home intelligence, factory controllability, and shopping malls automation [1]. Indoor positioning technology is a critical part in the location based services for indoor mobile target. High-precision positioning system can improve the automation and intelligence of mobile targets [2]. Strap-down Inertial Navigation System (SINS), entirely self-contained within the mobile target, neither sends signal to external nor depends on external signal [3]. The SINS can continuously supply the overall motion parameters and the short-term high performance for indoor positioning; however the SINS are known for their drift with time [4]. In order to achieve long-term stability, other technologies will be used to support the SINS. It is well known that the Global Positioning System (GPS) can supply the high-precision position and velocity of the mobile target [5]. According to the cumulative error of SINS, the GPS can be applied to

correct the SINS; then the integrated positioning system will be established based on SINS and GPS [6]. However, the signal of GPS is obstructed by the building; a non-GPS localization system need be aided by the SINS for indoor mobile target.

A Wireless Sensor Network (WSN) has enormous potential for the short-range indoor localization with intelligent and distributed network [7]. WSN is composed of some mobile nodes and a large number of anchor nodes through the self-organization and multihop methods [8]. In order to achieve continuous indoor positioning, some scholars put forward the INS/WSN integrated positioning system [9]. Hur and Ahn [10] propose a localization technique for mobile target using INS/WSN based on an intelligent filter with low complexity. Chen et al. [11] propose an INS/WSN integration system of mobile target with adaptive extended Kalman Filter. Because the WSN and SINS are two separate (self-contained) subsystems, the data alignment discrepancies between SINS and WSN would appear with

the clock difference and data transmission latency. The time synchronization between WSN and SINS becomes a matter of great public interest before the integrated positioning system is implemented [12]. The data alignment discrepancies could lead the suboptimal fusion algorithm and low-precision for the integrated system. This is due to the reason that the majority of the data fusion theories need to work in ideal condition without system time bias. However, the actual positioning system with multisensor cannot meet the condition of synchronous data fusion model [13]. In order to research this problem, Hu et al. [14] propose a batch asynchronous data fusion algorithm which can strictly synchronize asynchronous measurement in the time domain. Skog and Handel [15] and Yang and Shim [16] analyze the effects of time synchronization errors in a GPS-aided INS integrated positioning system and proposed a software-based time synchronization method using a data integration filter. Gao [17] proposes an asynchronous data fusion algorithm to solve problems about asynchronous fusion of multisensors and achieved good results for the SINS/GPS/CNS integrated navigation system. However, the data type of GPS is different from the WSN. The time synchronization method of GPS/INS integrated positioning system cannot be applied to the WSN/SINS positioning system. At the present, the proposed fusion method iteratively calculates the state equation for whole asynchronous data in every update period, which is described as the traditional asynchronous fusion algorithm. Furthermore, the traditional asynchronous fusion algorithms require large calculation burden and cost a lot of time, which may cause large position error occurring in a high-dynamic system.

Given all that, nowadays the research for time synchronization algorithms of integrated positioning system is mainly used for the GPS/INS. The traditional asynchronous data fusion algorithms have some adverse conditions which are the large calculation burden, being time-consuming, and large position error for high-dynamic system. To solve these problems, this paper proposed an asynchronous data fusion algorithm with Unscented Kalman Filter (UKF) to apply the SINS/WSN integrated positioning system. The state equation of state-space model is built with the motion characteristic of mobile target. The measurement equations of two subsystems are established based on the measurements of SINS and WSN, respectively. The time duration which is comprised of some sample periods of SINS is used for time unit of the fusion center. If the measurement of WSN existed in a time unit, the pseudo measurement equation of SINS/WSN will be built based on the asynchronous data fusion method. On the contrary, the measurement equation of pure SINS will be used. Hence the state-space model for asynchronous data fusion method is established. Owing to the nonlinear improved state-space model, the UKF is applied to estimate the state value. Then the novel asynchronous fusion with UKF for SINS/WSN integrated positioning system is achieved.

The rest of the paper is organized as follows. The proposed state-space model established with asynchronous data fusion theory is presented in Section 2. Section 3 describes Unscented Kalman Filter algorithm, while Section 4 evaluates the asynchronous data fusion method for SINS/WSN

integrated positioning system in simulation. In Section 5, we examine the performance of the proposed method. Finally, Section 6 concludes the paper.

## 2. State-Space Model for Positioning System

This section describes the process of establishment for the state-space model. This model is built with state equation and pseudo measurement equation of SINS/WSN integrated positioning system through the asynchronous data fusion method.

**2.1. State Equation.** The state vector of integrated system can be expressed as  $\mathbf{x} = [\mathbf{p}^n \ \boldsymbol{\varphi} \ \mathbf{v}^n \ \mathbf{a}^b]^T$ .  $\mathbf{p}^n$  and  $\mathbf{v}^n$  are the position and velocity of mobile target in the navigation frame ( $n$ -frame), respectively.  $\boldsymbol{\varphi}$  is the triaxis attitude angle of mobile target and  $\mathbf{a}^b$  is the acceleration of mobile target in the body frame ( $b$ -frame).

According to the movement characteristics of mobile target, the motion equations are illustrated as follows:

$$\begin{aligned} \mathbf{p}_k^n &= \mathbf{p}_{k-1}^n + \mathbf{v}_k^n T + \frac{\mathbf{C}_b^n \mathbf{a}_k^b T^2}{2}, \\ \mathbf{v}_k^n &= \mathbf{v}_{k-1}^n + \mathbf{C}_b^n \mathbf{a}_k^b T, \end{aligned} \quad (1)$$

where  $\mathbf{C}_b^n$  is the  $b$ - $n$ -frame transformation matrix.  $T$  is the sample time of system. Then the state equation of mobile target is shown as

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{G}_k \mathbf{W}_k, \quad (2)$$

where  $\mathbf{W}_k$  represents system noise vector,  $\mathbf{W}_k = [\boldsymbol{\varepsilon}^b \ \nabla^b]^T$ .  $\boldsymbol{\varepsilon}^b$  and  $\nabla^b$  are gyros drift and accelerometer bias, respectively.  $\mathbf{f}(\cdot)$  represents the  $n$ -dimension nonlinear vector function.  $\mathbf{G}_k$  is the model input matrix of system.

$$\mathbf{G}_k = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} T & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \frac{\mathbf{I}_{3 \times 3} T^2}{2} & \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} T & \mathbf{I}_{3 \times 3} \end{bmatrix}^T. \quad (3)$$

**2.2. Measurement Equations of SINS and WSN.** The measurement vector of SINS can be expressed as  $\mathbf{z}_{\text{SINS}} = [\mathbf{p}_{\text{SINS}}^n \ \mathbf{v}_{\text{SINS}}^n]^T$ , where  $\mathbf{p}_{\text{SINS}}^n$  and  $\mathbf{v}_{\text{SINS}}^n$  are the position and velocity of SINS. The measurement equation of SINS can be expressed in a generic form as

$$\mathbf{z}_{k,\text{SINS}} = \mathbf{H}_{k,\text{SINS}} \mathbf{x}_k + \mathbf{v}_{k,\text{SINS}}, \quad (4)$$

where  $\mathbf{v}_{k,\text{SINS}}$ , the Gaussian process noise, is the measurement noise of SINS as  $\mathbf{v}_{\text{SINS}} = [\mathbf{v}_{p,\text{SINS}} \ \mathbf{v}_{v,\text{SINS}}]^T$ . The measurement transformation matrix  $\mathbf{H}_{k,\text{SINS}}$  is denoted as

$$\mathbf{H}_{k,\text{SINS}} = \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}. \quad (5)$$

The measurement vector of WSN can be defined as  $\mathbf{z}_{\text{WSN}} = (\mathbf{p}_{\text{WSN}}^n)^T$ , where  $\mathbf{p}_{\text{WSN}}^n$  is the position of WSN. Then the measurement equation of WSN can be expressed as

$$\mathbf{z}_{k,\text{WSN}} = \mathbf{H}_{k,\text{WSN}}\mathbf{x}_k + \mathbf{v}_{k,\text{WSN}}, \quad (6)$$

where  $\mathbf{v}_{k,\text{WSN}}$ , the Gaussian process noise, is the measurement noise of WSN as  $\mathbf{v}_{k,\text{WSN}} = (\mathbf{v}_{p,\text{WSN}})^T$ . The measurement transformation matrix  $\mathbf{H}_{k,\text{WSN}}$  is denoted as  $\mathbf{H}_{k,\text{WSN}} = [\mathbf{I}_{3 \times 3} \quad \mathbf{0}_{3 \times 3}]$ .

**2.3. Description for Asynchronous Sensing.** According to the systematic characteristics of SINS and WSN, the SINS and WSN measure the motion information of mobile target independently. The sample time of SINS and WSN can be denoted by  $T_{\text{SINS}}$  and  $T_{\text{WSN}}$ , and  $T_{\text{WSN}} > 10T_{\text{SINS}}$ . We can define  $N_k$  as the number of measurements for all sensors at the interval  $(t_{k-1}, t_k]$ . Then note that one or more measured values might be supplied by a sensor, or none of measured values might be supplied by this sensor.  $n_k^{\text{SINS}}$  and  $n_k^{\text{WSN}}$  are the number of measurements from SINS and WSN at  $(t_{k-1}, t_k]$ , respectively. So the relationship is given by

$$N_k = n_k^{\text{SINS}} + n_k^{\text{WSN}}. \quad (7)$$

$\lambda_k^i$  is the time interval between  $t_k$  and the moment for  $i$ th measurement of SINS ( $i = 1, 2, \dots, n_k^{\text{SINS}}$ ).  $\beta_k^i$  is the time interval between  $t_k$  and the moment for  $i$ th measurement of WSN ( $i = 1, 2, \dots, n_k^{\text{WSN}}$ ). According to the time stamps of data, all measurements are sorted with the order of measured time in fusion center, after all measurements have reached the fusion center at  $(t_{k-1}, t_k]$ . Because the update frequency of SINS is considerably greater than that of WSN, it may lead to nonmeasurement of WSN in some fusion periods when the fusion period is smaller than the sampling period of WSN. Considering the high accuracy clock frequency of SINS, the time range with a fixed number of sampling periods of SINS is set as the data fusion period. Figure 1 displays the timeline of measured value.

According to (4) and (6), the measured expressions of SINS and WSN can be obtained at  $(t_{k-1}, t_k]$  as

$$\begin{aligned} \mathbf{z}_{k-\lambda_k^i}^{\text{SINS}} &= \mathbf{H}_{k-\lambda_k^i}^{\text{SINS}}\mathbf{x}_{k-\lambda_k^i} + \mathbf{v}_{k-\lambda_k^i}^{\text{SINS}}, \quad i = 1, 2, \dots, n_k^{\text{SINS}}, \\ \mathbf{z}_{k-\beta_k^i}^{\text{WSN}} &= \mathbf{H}_{k-\beta_k^i}^{\text{WSN}}\mathbf{x}_{k-\beta_k^i} + \mathbf{v}_{k-\beta_k^i}^{\text{WSN}}, \quad i = 1, 2, \dots, n_k^{\text{WSN}}, \end{aligned} \quad (8)$$

where  $\mathbf{H}_{k-\lambda_k^i}^{\text{SINS}}$  and  $\mathbf{H}_{k-\beta_k^i}^{\text{WSN}}$  are the measurement matrixes of SINS and WSN, respectively, which apply  $i$ th measured value.  $\mathbf{v}_{k-\lambda_k^i}^{\text{SINS}}$  and  $\mathbf{v}_{k-\beta_k^i}^{\text{WSN}}$  are the zero-mean Gaussian white noise vector. Their covariance matrixes are  $\mathbf{R}_{k-\lambda_k^i}^{\text{SINS}}$  and  $\mathbf{R}_{k-\beta_k^i}^{\text{WSN}}$ , respectively. We can define  $\mathbf{z}_l = \{\mathbf{z}_{l-\lambda_l^i}\}_{i=1}^{N_l}$  and  $\mathbf{Z}^k = \{\mathbf{z}_l\}_{l=1}^k$ . The state estimation value (denoted as  $\hat{\mathbf{x}}_{k-1|k-1} = E^*[\mathbf{x}_{k-1}|\mathbf{z}_{k-1}]$ ) and covariance matrix of estimation error (denoted as  $\mathbf{P}_{k-1|k-1} = \text{cov}[\hat{\mathbf{x}}_{k-1|k-1}|\mathbf{z}_{k-1}]$ ) last time have been obtained in fusion center at the moment of  $t_{k-1}$ . Then the asynchronous fusion method can be described simply as follows: the expressions (denoted as  $\hat{\mathbf{x}}_{k|k} = E^*[\mathbf{x}_k|\mathbf{z}_k]$  and  $\mathbf{P}_{k|k} = \text{cov}[\hat{\mathbf{x}}_{k|k}|\mathbf{z}_{k-1}]$ ) would be calculated after a new measurement  $\mathbf{z}_k$  is obtained in fusion center at the moment of  $t_k$ .

**2.4. Pseudo Measurement Equation of Integrated System.** According to data transmission characteristics of SINS/WSN, the pseudo measurement equation of asynchronous state-space model can be divided into two kinds of fusion types. When the measurements of SINS existed only in one update period, the state equation used by SINS is updated with SINS data, instead of asynchronous fusion state equation updating. Otherwise the pseudo measurement equation would be built with SINS and WSN, and then the asynchronous fusion algorithm will be calculated.

(a) If there has been no measurement of WSN at  $(t_{k-1}, t_k]$ , the discussion is illustrated as follows: if there has been no measurement of WSN at  $(t_{k-1}, t_k]$ , the data fusion center will implement the pure SINS update in the fusion center. The state equation and measurement equation of SINS in a sample time (denoted as  $(t_{k-\lambda_k^i}, t_{k-\lambda_k^i})$ ) can be obtained as

$$\begin{aligned} \mathbf{x}_{k-\lambda_k^i} &= \mathbf{f}(\mathbf{x}_{k-\lambda_k^{i-1}}) + \mathbf{G}_{k-\lambda_k^{i-1}}\mathbf{W}_{k-\lambda_k^{i-1}}, \\ \mathbf{z}_{k-\lambda_k^i}^{\text{SINS}} &= \mathbf{H}_{k-\lambda_k^i}^{\text{SINS}}\mathbf{x}_{k-\lambda_k^i} + \mathbf{v}_{k-\lambda_k^i}^{\text{SINS}}. \end{aligned} \quad (9)$$

Every SINS measured value would be used to update the state-space equation at  $(t_{k-1}, t_k]$ , so the fusion algorithm of positioning system in whole fusion period can be obtained.

(b) If there have been some measurements of WSN at  $(t_{k-1}, t_k]$ , the discussion is illustrated as follows: according to (2), at the moment of  $t_{k-\lambda_k^i}$ , the state vector is expressed as

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-\lambda_k^i}) + \mathbf{G}_{k-\lambda_k^i}\mathbf{W}_{k-\lambda_k^i}. \quad (10)$$

From (10), then  $\mathbf{x}_{k-\lambda_k^i}$  is

$$\mathbf{x}_{k-\lambda_k^i} = \mathbf{f}^{-1}(\mathbf{x}_k - \mathbf{G}_{k-\lambda_k^i}\mathbf{W}_{k-\lambda_k^i}). \quad (11)$$

Put it into (8) and the measurement vector of SINS can be obtained as

$$\mathbf{z}_{k-\lambda_k^i}^{\text{SINS}} = \mathbf{H}_{k-\lambda_k^i}^{\text{SINS}}\mathbf{f}^{-1}(\mathbf{x}_k - \mathbf{G}_{k-\lambda_k^i}\mathbf{W}_{k-\lambda_k^i}) + \mathbf{v}_{k-\lambda_k^i}^{\text{SINS}}. \quad (12)$$

According to Taylor's formula, (12) can be further developed as

$$\begin{aligned} \mathbf{z}_{k-\lambda_k^i}^{\text{SINS}} &= \mathbf{H}_{k-\lambda_k^i}^{\text{SINS}}\mathbf{f}^{-1}(-\mathbf{G}_{k-\lambda_k^i}\mathbf{W}_{k-\lambda_k^i})\mathbf{x}_k \\ &+ \mathbf{H}_{k-\lambda_k^i}^{\text{SINS}}\mathbf{f}^{-1}(-\mathbf{G}_{k-\lambda_k^i}\mathbf{W}_{k-\lambda_k^i}) + \mathbf{v}_{k-\lambda_k^i}^{\text{SINS}}. \end{aligned} \quad (13)$$

So we can define

$$\begin{aligned} \bar{\mathbf{h}}_{\text{SINS}}^{-i}(\mathbf{x}_k) &= \mathbf{H}_{k-\lambda_k^i}^{\text{SINS}}\mathbf{f}^{-1}(-\mathbf{G}_{k-\lambda_k^i}\mathbf{W}_{k-\lambda_k^i})\mathbf{x}_k, \\ \bar{\boldsymbol{\eta}}_{k,\text{SINS}}^{-i} &= \mathbf{H}_{k-\lambda_k^i}^{\text{SINS}}\mathbf{f}^{-1}(-\mathbf{G}_{k-\lambda_k^i}\mathbf{W}_{k-\lambda_k^i}) + \mathbf{v}_{k-\lambda_k^i}^{\text{SINS}}, \\ \bar{\mathbf{z}}_{k,\text{SINS}}^{-i} &= \mathbf{z}_{k-\lambda_k^i}^{\text{SINS}}. \end{aligned} \quad (14)$$

Then (13) is

$$\bar{\mathbf{z}}_{k,\text{SINS}}^{-i} = \bar{\mathbf{h}}_{\text{SINS}}^{-i}(\mathbf{x}_k) + \bar{\boldsymbol{\eta}}_{k,\text{SINS}}^{-i}. \quad (15)$$

Similarly, the measurement vector of WSN can be obtained as

$$\bar{\mathbf{z}}_{k,\text{WSN}}^{-i} = \bar{\mathbf{h}}_{\text{WSN}}^{-i}(\mathbf{x}_k) + \bar{\boldsymbol{\eta}}_{k,\text{WSN}}^{-i}. \quad (16)$$

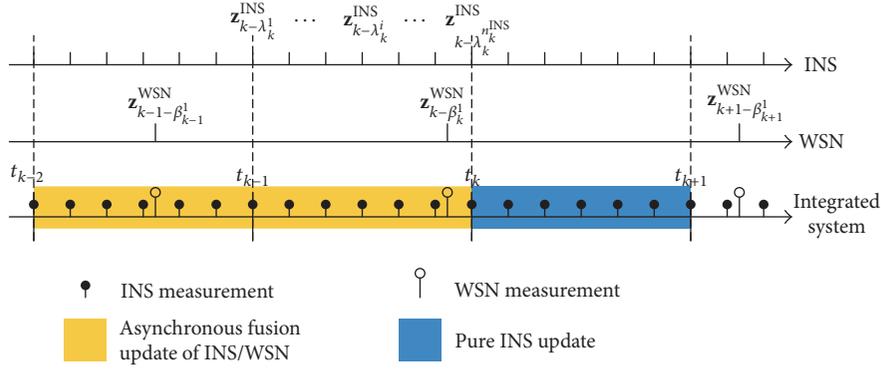


FIGURE 1: The schematic for the timeline of measured value.

According to the order of measured value from WSN and SINS, the parameters of pseudo measurement equation are defined as

$$\begin{aligned} \mathbf{z}_k &= \left[ (\bar{\mathbf{z}}_{k,\text{SINS}}^1)^T, \dots, (\bar{\mathbf{z}}_{k,\text{WSN}}^1)^T, \dots, (\bar{\mathbf{z}}_{k,\text{WSN}}^{n_k})^T, \dots, (\bar{\mathbf{z}}_{k,\text{SINS}}^{n_k})^T \right]^T, \\ \mathbf{h}(\mathbf{x}_k) &= \left[ (\bar{\mathbf{h}}_{\text{SINS}}^1)^T, \dots, (\bar{\mathbf{h}}_{\text{WSN}}^1)^T, \dots, (\bar{\mathbf{h}}_{\text{WSN}}^{n_k})^T, \dots, (\bar{\mathbf{h}}_{\text{SINS}}^{n_k})^T \right]^T, \\ \boldsymbol{\eta}_k &= \left[ (\bar{\boldsymbol{\eta}}_{k,\text{SINS}}^1)^T, \dots, (\bar{\boldsymbol{\eta}}_{k,\text{WSN}}^1)^T, \dots, (\bar{\boldsymbol{\eta}}_{k,\text{WSN}}^{n_k})^T, \dots, (\bar{\boldsymbol{\eta}}_{k,\text{SINS}}^{n_k})^T \right]^T. \end{aligned} \quad (17)$$

So the pseudo measurement equation can be obtained as

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \boldsymbol{\eta}_k. \quad (18)$$

Then the state-space model of asynchronous data fusion algorithm for the integrated positioning system can be achieved.

### 3. Unscented Kalman Filter for Positioning System

The state-space model based on SINS/WSN integrated positioning system which is reflected in (3) and (18) is a nonlinear model, so the Unscented Kalman Filter (UKF) would be applied to estimate the state vector [18]. We define the following: the initial state  $\mathbf{x}_0$  is independent from all noise. Initialized state vector and the state covariance matrix are  $E(\mathbf{x}_0) = \hat{\mathbf{x}}_0$  and  $\text{cov}(\mathbf{x}_0) = \mathbf{P}_0$ .

**3.1. Time Update.** With the given  $\hat{\mathbf{x}}_{k-1}$  and  $\mathbf{P}_{k-1}$ , the one-step prediction  $\hat{\mathbf{x}}_{k|k-1}$  and  $\mathbf{P}_{k|k-1}$  will be gotten through the Unscented Transformation (UT) as follows.

(a) *Sigma Points Calculation*

$$\boldsymbol{\xi}_{k-1}^{(0)} = \hat{\mathbf{x}}_{k-1},$$

$$\begin{aligned} \boldsymbol{\xi}_{k-1}^{(i)} &= \hat{\mathbf{x}}_{k-1} + \left( \sqrt{(n+\lambda)\mathbf{P}_{k-1}} \right)_i, \quad i = 1, \dots, n, \\ \boldsymbol{\xi}_{k-1}^{(i)} &= \hat{\mathbf{x}}_{k-1} - \left( \sqrt{(n+\lambda)\mathbf{P}_{k-1}} \right)_i, \quad i = n+1, \dots, 2n, \end{aligned} \quad (19)$$

where  $\lambda = \alpha^2(n + \kappa) - n$ ; the coefficient  $\alpha$  determines the diffused degree of the sigma points; here  $\alpha = 0.01$  and  $\kappa = 0$ .  $(\sqrt{(n+\lambda)\mathbf{P}_{k-1}})_i$  expresses the elements of matrix square root in the  $i$ th row.  $n$  is the number of dimensions for vector  $\mathbf{x}_k$ . From (3), we can get  $n = 12$ .

(b) *Sigma Points Transformation through the State Equation*

$$\begin{aligned} \boldsymbol{\xi}_k^{(i)} &= f_{k-1}(\boldsymbol{\xi}_{k-1}^{(i)}), \quad i = 0, 1, \dots, 2n, \\ \hat{\mathbf{x}}_{k|k-1} &= \sum_{i=0}^{2n} \omega_i^{(m)} \boldsymbol{\xi}_k^{(i)}, \\ \mathbf{P}_{k|k-1} &= \sum_{i=0}^{2n} \omega_i^{(c)} (\boldsymbol{\xi}_k^{(i)} - \hat{\mathbf{x}}_{k|k-1}) (\boldsymbol{\xi}_k^{(i)} - \hat{\mathbf{x}}_{k|k-1})^T + \mathbf{Q}_{k-1}, \end{aligned} \quad (20)$$

where

$$\omega_0^{(m)} = \frac{\lambda}{n + \lambda},$$

$$\begin{aligned}\omega_0^{(c)} &= \frac{\lambda}{n + \lambda} + (1 - \alpha^2 + \beta), \\ \omega_i^{(m)} &= \omega_i^{(c)} = \frac{1}{2(n + \lambda)}.\end{aligned}\quad (21)$$

$\omega_i^{(m)}$  is the weight coefficient for the mean associated with the  $i$ th point and  $\omega_i^{(c)}$  is the weight coefficient for the covariance associated with the  $i$ th point. The tuning parameter is  $\beta = 2$ .

**3.2. Measurement Update.**  $\hat{\mathbf{x}}_{k|k-1}$  and  $\mathbf{P}_{k|k-1}$  of sigma points from the measurement equation are calculated with the Unscented Transformation.

(a) *Sigma Points Calculation*

$$\begin{aligned}\xi_k^{(0)} &= \hat{\mathbf{x}}_{k|k-1}, \\ \xi_k^{(i)} &= \hat{\mathbf{x}}_{k|k-1} + \left( \sqrt{(n + \lambda) \mathbf{P}_{k|k-1}} \right)_i, \quad i = 1, \dots, n, \\ \xi_k^{(i)} &= \hat{\mathbf{x}}_{k|k-1} - \left( \sqrt{(n + \lambda) \mathbf{P}_{k|k-1}} \right)_i, \\ & \quad i = n + 1, \dots, 2n.\end{aligned}\quad (22)$$

(b) *One-Step Prediction for Measured Vector*

$$\begin{aligned}\xi_k^{(i)} &= \mathbf{h} \left( \xi_{k-1}^{(i)} \right), \quad i = 0, 1, \dots, 2n, \\ \hat{\mathbf{z}}_{k|k-1} &= \sum_{i=0}^{2n} \omega_i^{(m)} \xi_k^{(i)}, \\ \mathbf{P}_{\bar{\mathbf{z}}_k} &= \sum_{i=0}^{2n} \omega_i^{(c)} \left( \xi_k^{(i)} - \hat{\mathbf{z}}_{k|k-1} \right) \left( \xi_k^{(i)} - \hat{\mathbf{z}}_{k|k-1} \right)^T + \mathbf{R}_k, \\ \mathbf{P}_{\bar{\mathbf{x}}_k \bar{\mathbf{z}}_k} &= \sum_{i=0}^{2n} \omega_i^{(c)} \left( \xi_k^{(i)} - \hat{\mathbf{x}}_{k|k-1} \right) \left( \xi_k^{(i)} - \hat{\mathbf{z}}_{k|k-1} \right)^T.\end{aligned}\quad (23)$$

**3.3. Performing Update.** According to the new measurement  $\mathbf{z}_k$ , the filter of the system is updated, and then the state estimates can be obtained:

$$\begin{aligned}\hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \left( \mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1} \right), \\ \mathbf{K}_k &= \mathbf{P}_{\bar{\mathbf{x}}_k \bar{\mathbf{z}}_k} \mathbf{P}_{\bar{\mathbf{z}}_k}^{-1}, \\ \mathbf{P}_k &= \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{P}_{\bar{\mathbf{z}}_k}^{-1} \mathbf{K}_k^T,\end{aligned}\quad (24)$$

where  $\mathbf{K}_k$  is the gain matrix of filter.

Finally, the novel asynchronous data fusion algorithm has been built according to the state-space model and UKF. Figure 2 shows the flow chart for the novel asynchronous data fusion method using state-space model and UKF based on the SINS/WSN positioning system. The traditional asynchronous fusion algorithm cannot judge whether the WSN measurement exists in this fusion period, so it must calculate the asynchronous state-space equation in every

fusion period. It will consume more calculation and time than proposed asynchronous fusion algorithm; then the positioning error of traditional asynchronous fusion algorithm will be increased.

## 4. Simulation Results

Simulation experiments have been carried out to evaluate the performance of the proposed approach in comparison with the traditional fusion methods and the other filter algorithms based on SINS/WSN integrated positioning processing. Program codes are constructed with the Matlab 2010 version software. We can define the initial position and attitude angle as zero on the 3D Cartesian coordinate system. The vector of initial velocity is (10, 10, 0) m/s in three directions. The initial acceleration is zero in three directions, respectively. According to the performance parameters of SINS and WSN, the simulation initial conditions are expressed as  $\mathbf{Q} = \text{diag}(0.002, 0.002, 0.002, 0.002, 0.002, 0.002)$ ,  $\mathbf{R}_{\text{SINS}} = \text{diag}(30, 30, 30, 2, 2, 2)$ ,  $\mathbf{R}_{\text{WSN}} = \text{diag}(50, 50, 50)$ , and  $\mathbf{P} = \text{diag}(10, 10, 10, 1, 1, 1, 20, 10, 10, 10, 10, 10)$ . The sampling periods of SINS and WSN are  $T_{\text{SINS}} = 0.2$  s and  $T_{\text{WSN}} = 1$  s, respectively. The fusion period is selected at 0.8 s. The simulation lasts for 60 s. We can simulate the above process with UKF, Extended Kalman Filter (EKF), and Particle Filter (PF); the simulation results are shown in Figures 3, 4, and 5. The detailed derivation and algorithm procedures of EKF and PF can be found in [19, 20].

Figure 3 shows the positioning trajectory of proposed asynchronous data fusion method with UKF, EKF, and PF based on the SINS/WSN integrated positioning system. The reference trajectory is expressed as a black line with dot and that with EKF algorithm is expressed as red dash with triangle; in addition that with UKF algorithm is expressed as a blue line with dot and that with PF algorithm is expressed as a green line. Obviously the positioning system with UKF can track the reference trajectory effectively. Additionally the positioning system with PF can track the reference trajectory at the start of simulation, while that with PF has engendered the cumulative error over time. However the positioning system with EKF causes the serious divergence at the end of simulation. Because the PF algorithm needs a large amount of calculation, the position error of positioning system with PF has been accumulated over time. The EKF algorithm needs the first order linearization of the nonlinear function to be done; furthermore the EKF has to calculate the Jacobi matrix of the nonlinear function. Moreover the estimation accuracy of EKF will be seriously decreased and even diverged, when the model of positioning system has strong nonlinearity.

In Figure 4, the position errors of proposed asynchronous data fusion method with UKF, EKF, and PF algorithms are displayed. The UKF method is depicted in blue dash dotted line and the EKF method is in red dotted line and the PF method is in green real line. The maximum of position error with EKF is 38.1 m and the maximum of that with PF is 15.8 m; in addition the maximum of position error with UKF is 7.2 m. The UKF method reduces the position errors about 81.1% and 54.4% compared with EKF and PF methods. The UKF is a feasible filtering method for the improved asynchronous data

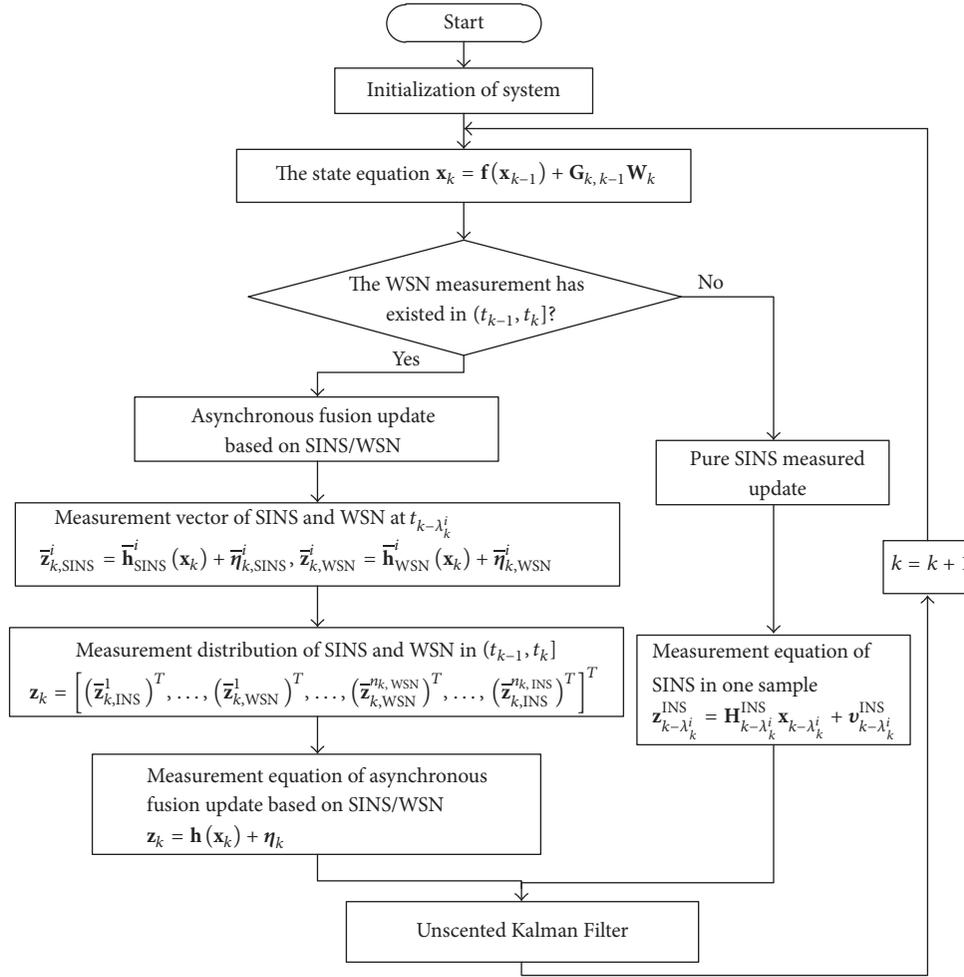


FIGURE 2: Flow chart for the asynchronous data fusion method based on the SINS/WSN positioning system.

fusion method based on SINS/WSN integrated positioning system.

Figure 5 shows the attitude errors of proposed asynchronous data fusion method with UKF, EKF, and PF algorithms. It is obvious that positioning systems with UKF and PF have small attitude error, while the attitude error of positioning system with EKF has engendered the divergence at the end of simulation. As a consequence, the maximum attitude errors with UKF and PF are 1.1 deg and 2.8 deg, respectively. Positioning accuracy of attitude with UKF meets the acquirement of indoor navigation.

In order to verify the effect of the SINS/WSN integrated positioning system, we design the simulation experiment with different fusion algorithms. Firstly, the time difference of data arrival between SINS and WSN always has not been considered with the simple synchronous data fusion methods, so this method can be described as the synchronous data fusion method. Moreover the SINS/WSN integrated positioning system with UKF is carried out based on three fusion methods which include the asynchronous data fusion method proposed by this paper and traditional asynchronous data fusion and synchronous data fusion methods. Finally, the results are shown in Figures 6 and 7.

Figure 6 shows the position errors for integrated positioning system based on the proposed method and traditional asynchronous data fusion (denoted as UKF-asyn) and synchronous data fusion (denoted as UKF-syn) methods. The arrival time of SINS data is over half second faster than that of WSN data. Position errors of asynchronous data fusion are larger than that of traditional asynchronous data fusion and proposed method, because the synchronous data fusion cannot consider the time difference between the SINS and WSN. The maximum position error with synchronous data fusion algorithm is 33.3 m, while the maximum position errors with traditional asynchronous data fusion algorithm and the proposed method are 10.8 m and 5.3 m, respectively. The traditional asynchronous data fusion algorithm cannot discuss whether the measurement of WSN exists in the fusion period, so it consumes more calculation time.

Figure 7 shows the average position errors for many time differences between SINS and WSN. We can define the SINS data as half or one second faster than the WSN data; in addition SINS data are half or one second slower than the WSN data. So the horizontal ordinate can be expressed as  $-1$ ,  $-0.5$ ,  $0$ ,  $0.5$ , and  $1$ . From Figure 7, we can see that the positioning error based on synchronous data fusion is

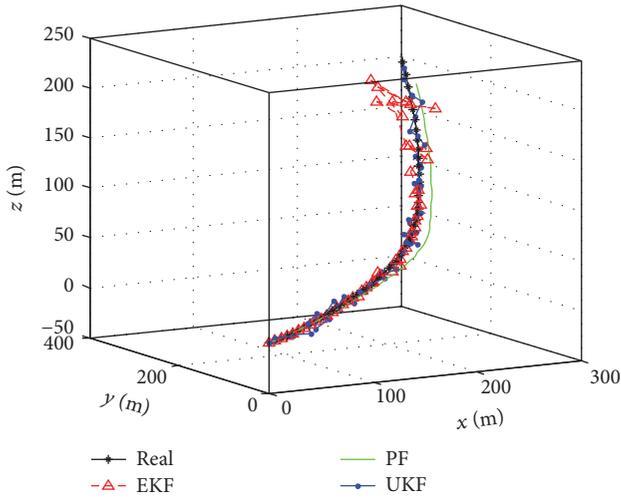


FIGURE 3: Positioning performance of improved asynchronous data fusion method with UKF, EKF, and PF algorithms.

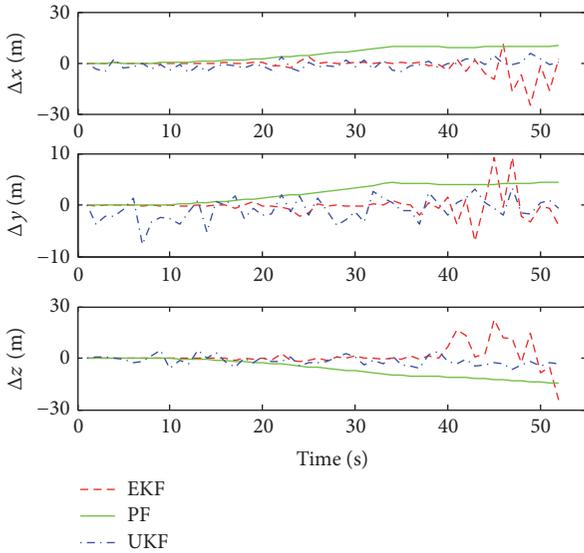


FIGURE 4: Position errors of improved asynchronous data fusion method with UKF, EKF, and PF algorithms.

the minimum without time difference between SINS and WSN. Similarly, the minimum position errors based on both the traditional asynchronous data fusion and proposed method appeared at the zero time difference. The position error with synchronous data fusion is the smallest of three fusion methods at none time difference, because the synchronous data fusion algorithm needs less calculation. The position error with synchronous data fusion has been increased quickly, when the SINS data is over a half second faster than WSN data. However, the position errors with traditional asynchronous data fusion and proposed method are hardly increased with increase of the time difference. The traditional asynchronous data fusion algorithm and the proposed method perform effectively stability for the varied time difference between SINS and WSN, and note that the

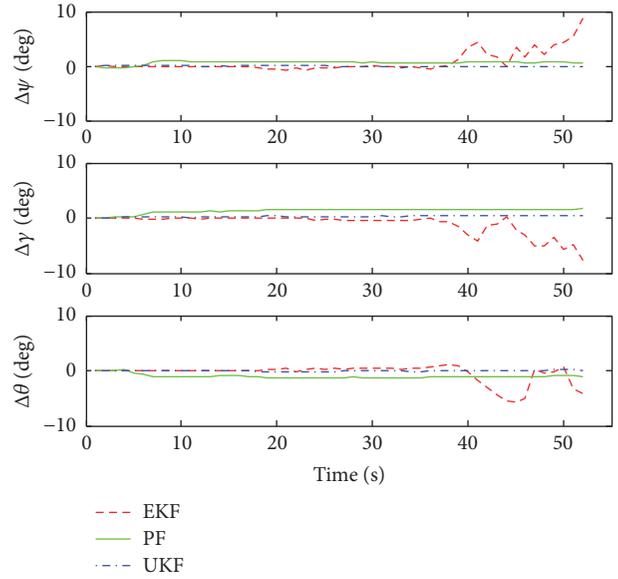


FIGURE 5: Attitude errors of improved asynchronous data fusion method with UKF, EKF, and PF algorithms.

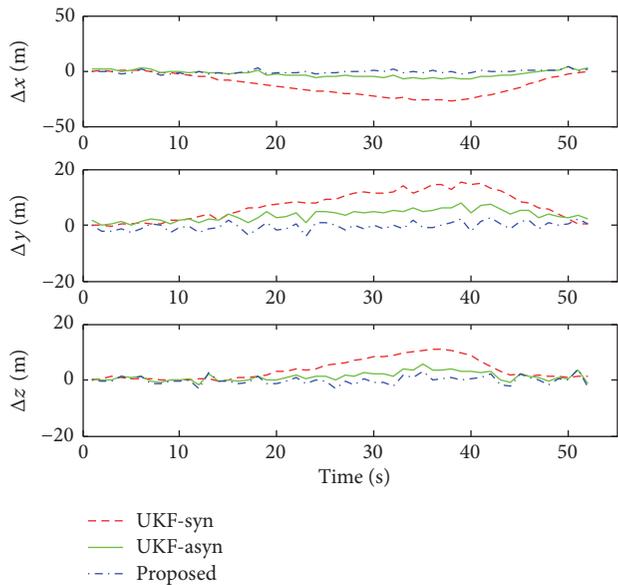


FIGURE 6: Position errors for integrated positioning system based on different fusion methods.

accuracy of proposed method is higher than traditional asynchronous data fusion algorithm.

## 5. Experimental Results

In this section, we perform several tests to evaluate the proposed asynchronous data fusion method based on SINS/WSN integrated navigation system in the room. An electric vehicle is used as the mobile target in the tests. The initial parameters of the positioning system are given as follows:

- (1) The WSN consisted of four anchor nodes and a mobile node. The mobile node is attached on the mobile

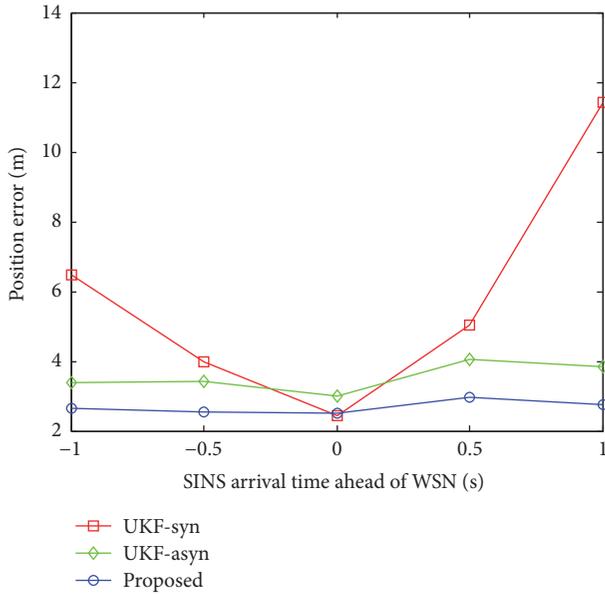


FIGURE 7: Average position errors of positioning system for various time differences between SINS and WSN.

target and the anchor nodes are deployed in the four corners of the room. A rectangle location area is constituted by four anchor nodes. Time synchronization for TOA approach among anchor nodes can be accomplished through the Ethernet cable. The power is supplied for anchor nodes through the twisted pair and the mobile node is operated through batteries. The sampling period of WSN is 1 s.

- (2) The used CleverNavi type SINS includes six-degree-of-freedom IMU ADIS16350 which consisted of a triaxial accelerometer and a triaxial gyroscope. The RS232 serial communication is used only for data transmission between SINS and computer. The baud rate is 115200 bit/s, and the sampling period is 0.01 s. Attitude reference precision of SINS is  $10''$  when it is towards the fixed north. The latitude accuracy is not less than  $1'$ , and the accuracy of gravitational acceleration is not less than  $2 \times 10^{-5} g$ . The power is supplied for the SINS through a storage battery in the electric vehicle.
- (3) The SINS is installed on the mobile target and the inertial data is transported with two Bluetooth models. One Bluetooth model is connected to the SINS with ribbon cable; the other one is connected to the computer by a wired USB-serial connection (Bluetooth 1.1 and USB 2.0). The maximum received distance between two Bluetooth models is up to 60 m in ideal conditions (free space). The wireless signal is broadcasted by the mobile node of WSN real time and received by anchor nodes. The distance values among the mobile node and anchor nodes are firstly collected by anchor nodes and then forwarded to the router which is connected with a computer through the Ethernet cable. Meanwhile inertial data of SINS

and the measured distance value of WSN are used to be computed in fusion center. As a consequence, the position and attitude of integrated positioning system can be obtained. Moreover the electric vehicle is controlled by an operator through a remote controller; the electric vehicle can move along a predesigned trajectory. Meanwhile the velocity of electric vehicle can be detected by a speed-sensor on a wheel. As a consequence the reference trajectory of mobile target can be obtained. The structure block for the experimental system is presented in Figure 8. Figure 9 shows the experimental diagram of integrated positioning system with SINS/WSN.

The frame of nodes deployment and a predetermined trajectory are shown in Figure 10. The positions of anchor nodes are (0.6, 7.4) m, (5.4, 7.4) m, (5.4, 0.6) m, and (0.6, 0.6) m, respectively. The starting point of mobile target is (4.75, 1.00) m. In order to control the dimension of matrix and calculation burden of fusion center, the sampling period time of SINS can be set as 0.1 s. The fusion period is selected as 0.8 s. The experiment lasts for 41 s.

Figure 11 shows the experimental performance of the SINS/WSN integrated positioning system with different data fusion algorithms. The real trajectory is shown with a black line. The trajectories with traditional asynchronous data fusion algorithm (denoted as UKF-asyn) and synchronous data fusion algorithm (denoted as UKF-syn) are expressed with the green dash line and red dash line, respectively. The trajectory with proposed asynchronous data fusion method is shown with a blue dash dotted line. Note that the trajectories with three algorithms can track the real trajectory accurately at the beginning of motion process. The trajectory with synchronous data fusion algorithm begins to produce the position error at the first corner of the trajectory. Furthermore, the position error with synchronous data fusion algorithm has deviated from the real trajectory with time. The maximum position error with synchronous data fusion algorithm is 1.11 m. However the trajectory of positioning system based on traditional asynchronous data fusion algorithm can track the real trajectory with small position error. The position error with the proposed method is smaller than that with traditional asynchronous data fusion algorithm. Because of the accumulative error of SINS, the positioning system with the proposed method still has a small position error for real trajectory. The average position error with the proposed method and traditional asynchronous data fusion algorithm are 0.28 m and 0.70 m, respectively. The tracking accuracy of the positioning system with the proposed method surpasses obviously that with synchronous data fusion and traditional asynchronous data fusion algorithms, 68.8% and 36.9% improvements, respectively. The position errors of integrated positioning system with different fusion methods are shown in Figure 12. Table 1 shows performance comparison for different fusion algorithms based on the SINS/WSN integrated positioning system.

In order to evaluate the real-time performance of the proposed method and traditional asynchronous data fusion algorithm, the integrated positioning system is used to test with

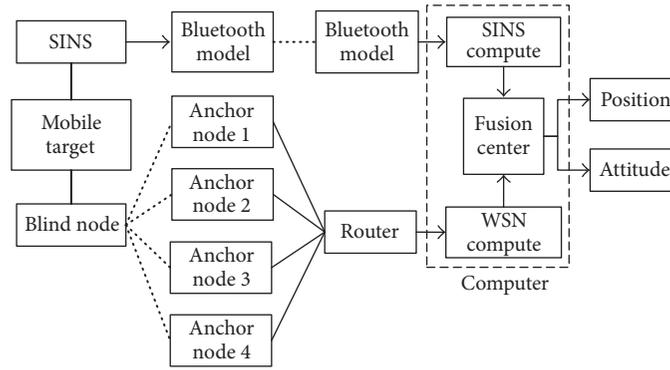


FIGURE 8: The structure block for the experimental system.

TABLE 1: Performance comparison for different fusion algorithms based on the integrated positioning system.

Item		UKF-syn	UKF-asyn	Proposed
Position error range (m)	X	-0.9918~0.5301	-0.6213~-0.5452	-0.4224~0.3368
	Y	-0.1534~1.3289	-0.3394~1.0541	-0.1422~0.5413
	Z	-2.3391~-0.0261	-1.1257~-0.0797	-0.4686~-0.1257
Variance of error	X	0.2409	0.1353	0.0592
	Y	0.2122	0.1888	0.0465
	Z	0.6061	0.1738	0.0696
Maximum (m)		2.4019	1.3630	0.5854
Average (m)		1.1099	0.7003	0.2777

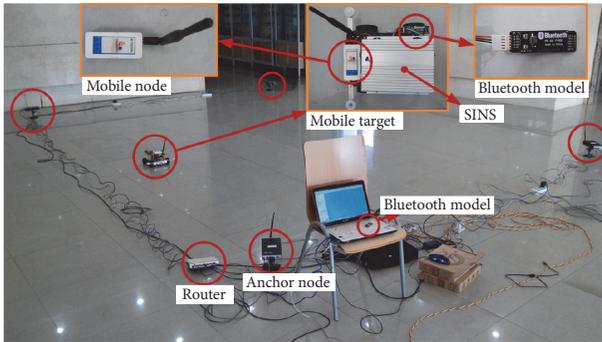


FIGURE 9: Experimental diagram of integrated positioning system with SINS/WSN.

different fusion periods. We set different fusion periods from 0.2 s to 1.6 s with every 0.2 s change. Then the consumption time for operation on the Matlab software can be obtained with different fusion periods as shown in Figure 13. The Matlab software is operated on the ASUS X86 laptop with 2 GB RAM and CPU Intel (R) Core(TM)2 Duo T6670 @2.20 GHz. The consumption time for traditional asynchronous data fusion algorithm and the proposed method is depicted in blue line with rectangle and red line with circle, respectively. As the fusion period is increased, consumption time with both traditional asynchronous data fusion algorithm and proposed method is increased. However the consumption time with traditional asynchronous data fusion algorithm is much than that with proposed method, when the fusion period is 0.2 s. The traditional asynchronous data fusion algorithm and

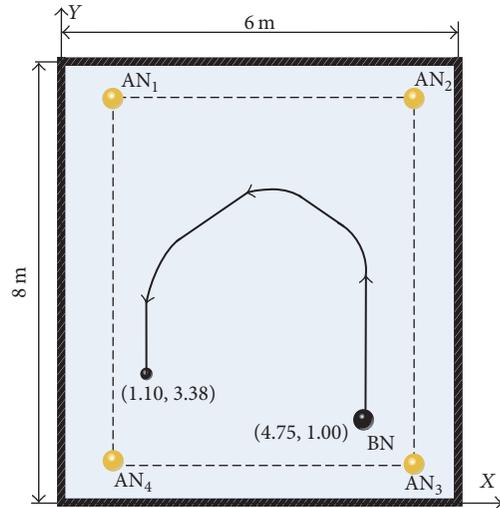


FIGURE 10: The frame of nodes deployment.

proposed method have approximate consumption time while the fusion period is more than 1.0 s. So the method proposed by this paper consumes less calculation burden and has better real-time performance in comparison with the traditional asynchronous data fusion algorithm.

### 6. Conclusion

Because of data alignment discrepancies between SINS and WSN, the integrated positioning system with conventional

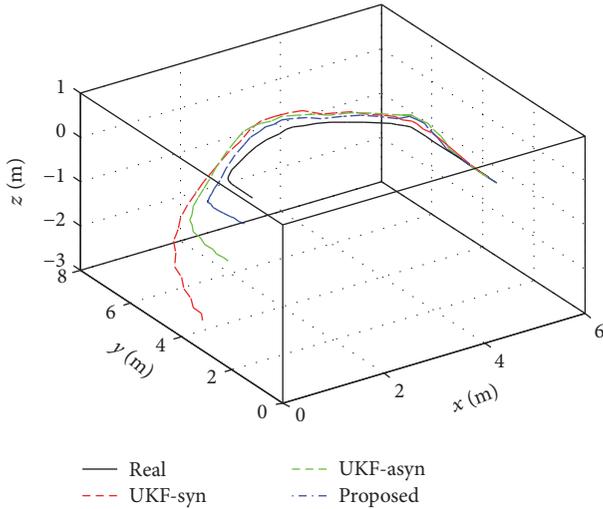


FIGURE 11: Experimental performance of the SINS/WSN integrated positioning system with fusion algorithm.

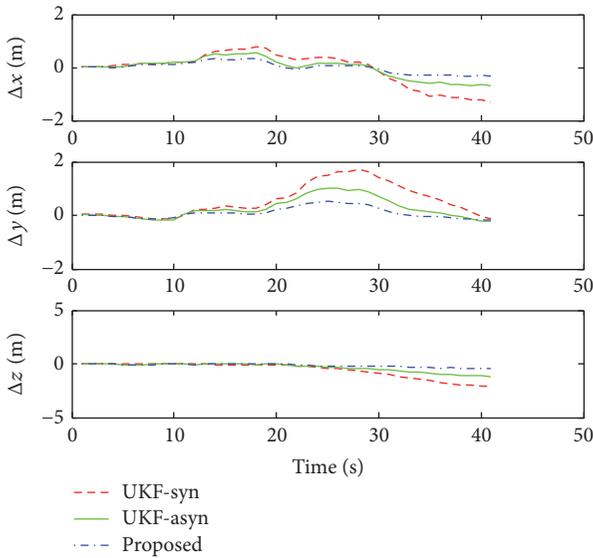


FIGURE 12: The position error of integrated positioning system with different fusion methods.

synchronous data fusion algorithm will bring the large positioning error. The traditional asynchronous data fusion algorithm needs large calculation burden and causes the low positioning accuracy for high-dynamic system. This paper proposes a novel asynchronous data fusion algorithm with UKF for the SINS/WSN integrated positioning. Asynchronous data fusion algorithm based on positioning system is analyzed in simulation with variety of conditions. Simulation results indicated that the maximum position error with UKF is 7.2m when the mobile target moves about 600 meters; meanwhile the position error with UKF is reduced by 81.1% and 54.4% compared with EKF and PF methods, respectively. Additionally the position error with the proposed method is the smallest compared with the synchronous data fusion and traditional asynchronous

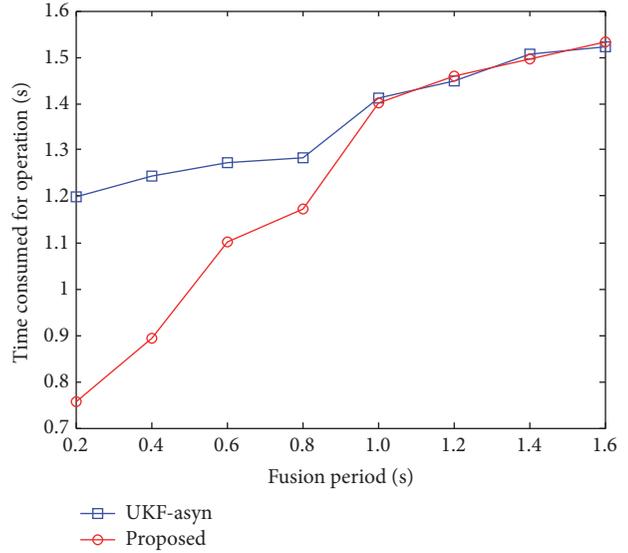


FIGURE 13: The consumption time for operation with different fusion periods.

data fusion algorithms. Note that the position error with synchronous data fusion is grown seriously; the proposed method and traditional asynchronous data fusion algorithm performed stability as the time difference increased. The experimental results present the fact that the positioning system with the proposed method can track the real trajectory effectively. The position error with the proposed method is 0.28 m and surpasses that with synchronous data fusion and traditional asynchronous data fusion algorithms, 68.8% and 36.9% improvements. Moreover the proposed method needs less calculation and has better real-time performance in comparison with the traditional asynchronous data fusion algorithm. As a consequence, the asynchronous data fusion method with improved state-space model and UKF is a feasible positioning method for the data alignment discrepancies.

**Conflicts of Interest**

The authors declare that there are no conflicts of interest regarding the publication of this paper.

**Acknowledgments**

This work was supported by the Joint Funds of the National Natural Science Foundation of China (U1610111) and by the “333 Project” of Jiangsu Province (BRA2015300), a project supported by the Fundamental Research Funds for the Central Universities (2015B04814), and a project supported by Young Scholars Development Fund of SWPU (201799010002).

**References**

[1] Y. Zhuang and N. El-Sheimy, “Tightly-coupled integration of WiFi and MEMS sensors on handheld devices for indoor pedestrian navigation,” *IEEE Sensors Journal*, vol. 16, no. 1, pp. 224–234, 2016.

- [2] H. Yang, W. Li, and C.-M. Luo, "Fuzzy adaptive Kalman filter for indoor mobile target positioning with INS/WSN integrated method," *Journal of Central South University*, vol. 22, no. 4, pp. 1324–1333, 2015.
- [3] M. Jafari, T. A. Najafabadi, B. Moshiri, S. S. Tabatabaei, and M. Sahebameyan, "PEM stochastic modeling for MEMS inertial sensors in conventional and redundant IMUs," *IEEE Sensors Journal*, vol. 14, no. 6, pp. 2019–2027, 2014.
- [4] Q. Yuan and I.-M. Chen, "3-D localization of human based on an inertial capture system," *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 806–812, 2013.
- [5] G.-B. Chang and M. Liu, "Hybrid Kalman and unscented Kalman filters for INS/GPS integrated system considering constant lever arm effect," *Journal of Central South University*, vol. 22, no. 2, pp. 575–583, 2015.
- [6] A. El-Shafie, A. Najah, and O. A. Karim, "Amplified wavelet-ANFIS-based model for GPS/INS integration to enhance vehicular navigation system," *Neural Computing and Applications*, vol. 24, no. 7-8, pp. 1905–1916, 2014.
- [7] A. Colombo, D. Fontanelli, D. MacLi, and L. Palopoli, "Flexible indoor localization and tracking based on a wearable platform and sensor data fusion," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 4, pp. 864–876, 2014.
- [8] Y. Zhang, D. Wei, W. Fu, and B. Yang, "Target positioning with GDOP assisted nodes selection algorithm in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 404812, 10 pages, 2014.
- [9] C. Luo, W. Li, H. Yang, B. Ying, and G. Xin, "Implementation of mobile target positioning technology integrating SINS with WSN measurements," *Journal of Sensors*, vol. 2014, Article ID 673179, 12 pages, 2014.
- [10] H. Hur and H.-S. Ahn, "Discrete-time  $H_\infty$  filtering for mobile robot localization using wireless sensor network," *IEEE Sensors Journal*, vol. 13, no. 1, pp. 245–252, 2013.
- [11] X. Chen, Y. Xu, and Q. Li, "Application of adaptive extended kalman smoothing on INS/WSN integration system for mobile robot indoors," *Mathematical Problems in Engineering*, vol. 2013, Article ID 130508, 8 pages, 2013.
- [12] C. Luo, W. Li, M. Fan, H. Yang, and Q. Fan, "A collaborative positioning algorithm for mobile target using multisensor data integration in enclosed environments," *Computer Communications*, vol. 44, pp. 26–35, 2014.
- [13] H. N. Abu Abdelsalam, M. M. Atia, C. Goodall, A. Noureldin, and N. El-Sheimy, "Assessment of time delays and synchronization errors in real-time INS/GNSS systems and its impact on the navigation performance," in *Proceedings of the 25th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS '12)*, pp. 3192–3197, September 2012.
- [14] Y. Y. Hu, Z. S. Duan, and C. Z. Han, "Optimal batch asynchronous fusion algorithm," in *Proceedings of the IEEE International Conference on Vehicular Electronics and Safety (ICVES '05)*, pp. 237–240, October 2005.
- [15] I. Skog and P. Handel, "Time synchronization errors in loosely coupled GPS-aided inertial navigation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1014–1023, 2011.
- [16] C.-K. Yang and D.-S. Shim, "The effect of measurement for time synchronization error in the tightly coupled GPS/INS integration," in *Proceedings of the 9th Asian Control Conference (ASCC '13)*, pp. 1–5, Istanbul, Turkey, June 2013.
- [17] W. Gao, Y. Zhang, and Q. Sun, "An asynchronous fusion algorithm of the SINS/GPS/CNS based on factor graph," in *Proceedings of the IEEE International Conference on Chinese Control Conference (CCC '13)*, pp. 4995–4999, 2013.
- [18] J. Wang, A. Hu, X. Li, and Y. Wang, "An improved PDR/magnetometer/floor map integration algorithm for ubiquitous positioning using the adaptive unscented Kalman filter," *ISPRS International Journal of Geo-Information*, vol. 4, no. 4, pp. 2638–2659, 2015.
- [19] L. J. Liu, B. Qi, S. M. Cheng, and Z. R. Xi, "High precision estimation of inertial rotation via the extended Kalman filter," *European Physical Journal D*, vol. 69, no. 11, pp. 261–270, 2015.
- [20] A. Perttula, H. Leppakoski, M. Kirkko-Jaakkola, P. Davidson, J. Collin, and J. Takala, "Distributed indoor positioning system with inertial measurements and map matching," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 11, pp. 2682–2695, 2014.

## Review Article

# Sequential Monte Carlo Localization Methods in Mobile Wireless Sensor Networks: A Review

**Ammar M. A. Abu Znaid,<sup>1</sup> Mohd. Yamani Idna Idris,<sup>1</sup> Ainuddin Wahid Abdul Wahab,<sup>1</sup> Liana Khamis Qabajeh,<sup>2</sup> and Omar Adil Mahdi<sup>1</sup>**

<sup>1</sup>*Faculty of Computer Science and Information Technology, University of Malaya, 50603 Lembah Pantai, Kuala Lumpur, Malaysia*

<sup>2</sup>*Computer Engineering and Science Department, Information Technology and Computer Engineering Faculty, Palestine Polytechnic University, Wadi-Alhariya St, P.O. Box 198, Hebron, State of Palestine*

Correspondence should be addressed to Mohd. Yamani Idna Idris; [yamani@um.edu.my](mailto:yamani@um.edu.my)

Received 13 December 2016; Accepted 13 March 2017; Published 3 April 2017

Academic Editor: Mohannad Al-Durgham

Copyright © 2017 Ammar M. A. Abu Znaid et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The advancement of digital technology has increased the deployment of wireless sensor networks (WSNs) in our daily life. However, locating sensor nodes is a challenging task in WSNs. Sensing data without an accurate location is worthless, especially in critical applications. The pioneering technique in range-free localization schemes is a sequential Monte Carlo (SMC) method, which utilizes network connectivity to estimate sensor location without additional hardware. This study presents a comprehensive survey of state-of-the-art SMC localization schemes. We present the schemes as a thematic taxonomy of localization operation in SMC. Moreover, the critical characteristics of each existing scheme are analyzed to identify its advantages and disadvantages. The similarities and differences of each scheme are investigated on the basis of significant parameters, namely, localization accuracy, computational cost, communication cost, and number of samples. We discuss the challenges and direction of the future research work for each parameter.

## 1. Introduction

The digital world is becoming increasingly important in our daily lives with the heavy utilization of numerous small, cheap devices called sensor nodes. These sensor devices can be controlled and can communicate and cooperate remotely to investigate far and hazardous areas [1–4]. Sensor nodes are utilized in different fields, such as the Internet of Things [5, 6], health care [7], zoo monitoring [8], underwater exploration [9], intelligent city [10, 11], military applications [12], routing optimization [13], and dynamic mapping [14, 15].

The localization schemes in wireless sensor networks (WSNs) can be classified into two types, namely, static and mobile networks [16]. A static network is constructed with stationary sensor nodes; the sensors are deployed randomly or on the basis of a previous plan. By contrast, the sensor nodes in a mobile network are flexible to maximize their benefits in improving WSNs coverage and power consumption and in discovering other areas with a limited number of sensors [17].

Generally, the localization schemes of a mobile sensor are classified as range-based and range-free schemes [18, 19]. However, in this work, we classified the localization schemes into three groups, namely, range-based, range-free, and hybrid schemes. The range-based scheme uses additional hardware such as antenna to estimate the location of a blind node (i.e., a node without location information), whereas the range-free scheme uses network connectivity. The hybrid scheme is a combination of the range-free scheme for noise cases and the range-based scheme for stable cases. In all the aforementioned schemes, anchor nodes (i.e., nodes with location information) broadcast their location information per time slot to assist blind nodes in estimating their location.

Range-free localization schemes are classified into four categories, namely, hop count, fingerprint algorithm, Monte Carlo scheme, and hybrid schemes (SMC and hop distance). The hop count estimates the location of a blind node through an average of hop distance. Hence, each node maintains the minimum hop number of the anchor node in the network.

In the fingerprint algorithm, the location of a blind node is estimated in two stages. The first stage involves the construction of an offline database by measuring the signal strength in the deployment area, and the second stage involves the real-time estimation of the location of a blind node by matching the signal strength of this blind node with the offline database. The Monte Carlo scheme uses the probability distribution function (PDF) to estimate the location of a blind node [19]. The hybrid schemes (SMC and hop distance) advance localization accuracy by utilizing the DV-hop (distance vector-hop) technique on MCL (Monte Carlo localization).

A majority of range-free schemes use the sequential Monte Carlo (SMC) technique to estimate the location of blind nodes in dynamic systems within three steps, namely, initial, sample, and filter stages [20, 21]. The location of mobile sensors is an important parameter in WSNs. Thus, a high level of localization accuracy can improve the confidence and quality of sensing data. In the present study, we classified the performance of SMC schemes according to three categories, namely, localization accuracy, computational cost, and communication cost.

Localization accuracy is measured with the variance of the Euclidean distance between the estimated location and the real location [11]. The localization accuracy in SMC schemes is mostly affected by two parameters, namely, the density of anchor nodes and number of samples [22–24]. Hence, a large number of anchor nodes can improve localization accuracy by broadcasting rich location information in the area. Moreover, a sufficient number of valid samples can improve localization accuracy. However, the performance of SMC schemes is extremely dependent on the distribution function of previous samples.

The computational cost to generate a sufficient valid sample can be measured with the number of iterations required to find a sufficient valid sample. SMC requires a sequential repetition of sample and filter steps until a sufficient valid sample is obtained. The efficiency of the samples is also affected by the bounded sample area and sample evaluation [25].

The communication cost in range-free localization schemes can be determined with the number of messages that are sent during the localization process [26]. Consequently, the accuracy in range-free schemes is highly dependent on the density of anchor nodes and normal nodes (node's new location in the last time slot), which can increase the number of messages sent. Moreover, the size of messages affects communication cost.

The following are the contributions of the present study: a comparison of existing surveys on WSNs localization, a classification of state-of-the-art SMC schemes and a thematic taxonomy, a comprehensive survey of state-of-the-art localization operation parameters, a discussion of critical aspects, and the identification of challenges and open issues.

This paper consists of eight sections organized as follows. Section 2 presents the comparison of current surveys on sensor localization. Section 3 defines the thematic taxonomy of existing localization schemes. Section 4 explains the

elementary approach related to SMC and the evaluation parameters for the localization process. Section 5 reviews the state-of-the-art SMC schemes by discussing their advantages and disadvantages. Section 6 presents a comparison of state-of-the-art SMC techniques. Section 7 presents the discussion and future works. Finally, Section 8 concludes the review.

## 2. Comparison of Surveys on WSNs Localization

Localization problems have been studied in various WSNs schemes; a survey of these schemes can be found in [18, 27, 31, 32]. The present study presents a comprehensive review of the localization problem in mobile WSNs. However, to highlight and distinguish our contribution from other surveys, we summarized and compared the existing surveys on localization problems in WSNs, as shown in Table 1.

In general, previous schemes maintain static networks, whereas current schemes maintain mobile networks. However, the localization schemes in both networks can be classified as range-based and range-free [37]. The survey in [28] classified the state of sensors into four types, namely, static landmark node and static node, mobile landmark node and static node, static landmark node and mobile node, and mobile landmark node and mobile node.

The survey of range-free schemes in [18] classified these schemes into the following categories: APIT, DV-hop, multihop, centroid, and gradient. Another survey classified range-free localization schemes in emerging applications (cyber physical systems and cyber transportation systems) into proximity-based localization, one-hop localization, and multihop localization. Moreover, range-based schemes were classified in [16] into beacon-based distributed localization, relaxation-based distributed algorithm, coordinate system stitching-based localization, and hybrid localization. Beacon-based distributed localization can be further classified into three categories, namely, diffusion, bounded box, and gradient.

The survey in [29] classified mobile sensor networks in disaster scenarios, in which mobile nodes aid in the search for disaster locations. The localization schemes in static networks are classified as range-free and range-based schemes, whereas those in mobile networks are classified as robotic, MCL, and range-based schemes. Another survey on harsh environments [30] classified localization schemes into range-based and range-free, anchor-based and anchor-free, and distributed and centralized schemes.

The survey of localization classification and technique evaluation [31] classified localization schemes as geometrical techniques, multidimensional scaling, stochastic proximity embedding, convex and nonconvex optimization, and hybrid. An indoor application survey discussed the potential improvement of the human mobility model by utilizing smartphones [27]. Moreover, this survey investigated smartphone sensors according to location accuracy, deployment cost, location context, cost, quality, and measurement errors.

In [33], the localization schemes were classified into target localization and self-localization. Additionally, this

TABLE 1: Previous survey of wireless sensor localization.

Reference	Taxonomy	Comparison parameters	Years
[18]	APIT, DV-hop, multihop, centroid, gradient	Node density, cost, accuracy, overhead, scalability	2015
[27]	Types of sensors, types of mobility, measurement errors	Location accuracy, deployment cost, location context, quality and cost of smartphone, and measurement errors	2015
[28]	Static landmark and static node, mobile landmark and static node, static landmark and mobile node, mobile landmark and mobile node	Localization accuracy, coverage, time, landmark density, node density, energy consumption	2013
[29]	Static (range-free, range-based), mobile (robotic, MCL, range-based)	(Centralized, distributed), dimensional analysis, simulator, (range-free, range-based), scalability, communication radius	2013
[30]	(range-based, range-free), (anchor-based, anchor-free), (distributed, centralized)	accuracy, hardware cost, computation cost, and communication cost	2012
[31]	Geometrical techniques, multidimensional scaling, stochastic proximity embedding convex, and nonconvex optimization and hybrid	Accuracy, coverage, complexity, scalability, robustness, and cost	2012
[32]	Proximity-based localization, one-hop localization and multihop localization	Without comprehensive comparison	2012
[33]	Target/source localization and node self-localization	Non-line-of-sight, energy-constrained network, tradeoff between localization performance and energy consumption, cooperative node localization, and localization in heterogeneous network	2012
[16]	Beacon-based distributed localization, relaxation-based distributed localization, the Coordinate system stitching-based localization, and hybrid localization	Objective (centralized, distributed), description, accuracy, computation cost	2010
Proposed	Range-based, range-free, and hybrid. Range-free (localization accuracy, communication cost, and computation cost)	Velocity, anchor and normal node density, degree of irregularity, size of sample area, number of messages, and message size	2016

survey reviewed the localization challenges in non-line-of-sight node selection, optimizing the tradeoff between energy depletion performance, cooperative nodes, and localization in a heterogeneous radio range.

The present survey investigates the state-of-the-art localization schemes in mobile WSNs in microscopic classification. The schemes are categorized as range-based, range-free, and hybrid. The range-free scheme is further subcategorized into fingerprint, Monte Carlo, hop distances, and hybrid (i.e., SMC and hop distance). Furthermore, we classify the SMC scheme according to its main operational parameters, namely, localization accuracy, communication cost, and computation cost, in microscopic classification. The comparison of the localization schemes assists network end users and administrators in tracking and identifying the location of areas under investigation. Thus, appropriate schemes are selected to localize mobile WSNs. Throughout this study, we further discuss the challenges and open issues related to each location parameter.

### 3. Localization Scheme Classification

Estimating the location of mobile sensors is a challenging task in WSNs because of the frequent changes in the location of mobile nodes per time slot, the whole topology, and connectivity of networks. Additionally, the sensor node's hard-

ware limitations, such as limited power sources, memory, processor unit, and communication range, further complicate the estimation process [38]. Therefore, WSNs need a smart and robust technology to estimate sensor location. We classified localization schemes into three categories, namely, range-based, range-free, and hybrid; the SMC in range-free schemes was classified on the basis of localization accuracy, communication cost, and computation cost, as shown in Figure 1.

**3.1. Range-Based Localization.** In range-based schemes, the blind node finds its location using its absolute distance from the anchor nodes. Range-based schemes use different types of hardware to calculate distance, such as time of arrival (ToA). ToA measures the distance between the time of arrival and the time of departure between nodes. Then, light speed is used to calculate the distance between nodes on the basis of a speed equation. However, ToA needs additional hardware to synchronize the transmission times between sensor nodes. The time synchronization increases the traffic in networks and delays the localization process [39].

The study in [40] proposed a time difference of arrival (TDoA) between sound and light to improve ToA. TDoA uses additional acoustic hardware to measure the difference between light and sound signals from the source. The angle of arrival (AoA) and triangle geometry between neighbors are

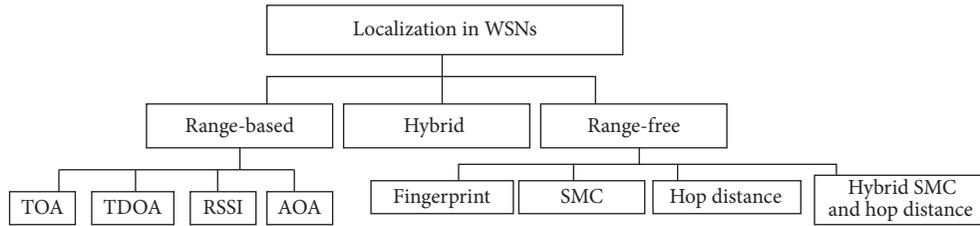


FIGURE 1: Taxonomy of localization schemes in mobile WSNs, including range-free and sequential Monte Carlo schemes.

also used to calculate blind node location. In AoA, the sensor node uses antennas to measure the angle between neighbors [41].

The received signal strength indicator (RSSI) measures the distance according to the difference in signal strengths [42]. The RSSI assumes that signal strength degrades over distance; this characteristic is used to measure distance without additional hardware. However, signal strength is affected by noise, such as physical phenomena and weather conditions; these distortions reduce the accuracy of distance measurement.

The global positioning system (GPS) is typically used to localize objects in outdoor applications. However, GPS is inapplicable to indoor applications because GPS requires the lines-of-sight of at least three satellite signals at the same time to determine the location of an object [43]. Moreover, GPS signals are affected by obstacles, walls, and physical phenomena. The other limitations of GPS include high power consumption, high cost, and large size.

**3.2. Range-Free Localization.** Range-free schemes estimate blind node location through network connectivity without additional hardware. Thus, the blind node requires the following: information about nodes that are within its radio range, the location estimation of nodes, and the ideal radio range of each sensor. The anchor nodes in range-free schemes broadcast their locations at each time slot to help the blind node in estimating its location. Generally, the blind node needs at least three anchor node locations in the neighborhood to estimate its location. Range-free schemes are more cost-effective than range-based schemes. Range-free schemes can be classified into the following types: hop distance, fingerprint, SMC, and hybrid schemes utilizing SMC and hop distance.

**Hop Distance.** Hop distance uses the average hop to estimate the distance between anchor nodes. The localization process in DV-hop follows three steps, namely, location broadcast, distance calculation, and location estimation [44]. In location broadcast, the anchor node broadcasts its location information and initializes the hop count to zero among its neighbors. The receiver node keeps the minimum hop count for each anchor node and disregards the large hop count from the same anchor nodes. Then, the receiver increases the hop count by one and sends it to the neighbors. Hence, each node has a record of the minimum hop count of all anchor nodes. In distance calculation, the node calculates the

average distance with each anchor node over the hop count of all anchor nodes. In location estimation, the blind node calculates its location by interlocking the matrixes of anchor node location and the matrixes of distance with anchor nodes [45]. The disadvantage of hop distance is that it requires a uniform distribution of anchor nodes in the whole network to achieve high accuracy. Consequently, DV-hop is limited to specific applications.

**Fingerprint.** The fingerprint localization approach estimates blind node location in two steps, namely, creation of an offline database and online location estimation. The offline database is constructed from signal characteristics (called fingerprints) and the location recorded from the whole part of the area of interest. Then, location is estimated for the mobile user by matching the signal fingerprint from the user with that in the database server. Once the signal fingerprint matches that in the database server, the estimated location is sent back to the user. The main drawback of the fingerprint localization approach is the creation and updating of the database. Creating the database requires some expert personnel to collect fingerprints from areas of interest; updating the offline database is a time-consuming task when changes, such as the addition or removal of a new access point in the area of interest, are made in the environment. Moreover, mobile sensors can share similar fingerprints that degrade accuracy and promote ambiguity. This drawback of the fingerprint localization approach requires a qualified engineer who would measure signal strength [46].

**Sequential Monte Carlo (SMC).** Mobile sensors change their locations frequently over time. Hence, finding their current locations requires relocalization at each time slot. SMC is an efficient method for a dynamic system; SMC employs the PDF in the previous time slot and observes it at the current time to estimate the current location by using a weighted particle filter [47].

SMC makes the following two assumptions: (1) time is divided into discrete time units and (2) enough samples are required at each time slot. The SMC scheme estimates blind node location in a distributed manner on the basis of the connectivity information “who is within the communication range of whom” [48].

The localization process in SMC involves three stages (as in Algorithm 1), namely, the initial, sample, and filter stages. In the initial stage, the blind node estimates its location by generating samples randomly from the whole area. In the

sample stage, the blind node draws samples in the current time slot on the basis of the samples from the previous time slot bounded by a maximum velocity. Hence, the node generates samples through the following transition equation:

$$p(S_t | S_{t-1}) = \begin{cases} \frac{1}{\pi v_{\max}^2} & \text{if } d(S_t | S_{t-1}) \leq v_{\max}, \\ 0 & \text{if } d(S_t | S_{t-1}) > v_{\max}, \end{cases} \quad (1)$$

where  $v_{\max}$  is the node maximum velocity and  $d(S_t | S_{t-1})$  is the distance of the sample location between the current time and the previous time.

In the filter stage, the samples are weighted according to the anchor node constraint in the current time. Each valid sample must be within one or two hops of the three anchor node constraints. Otherwise, the sample is filtered out. SMC repeats the sample and filter stages sequentially until sufficient valid samples are discovered.

*Algorithm 1* (phase of SMC localization algorithm).

- (1) *Phase One.* Initial phase
- (2) Generate samples  $S$  randomly from the whole area.
- (3) *Phase Two.* Generating samples
- (4) Sample set  $C_t = \{\}$
- (5) For each sample  $S$  in previous time ( $l_{t-1}$ ), generate a new sample according to
- (6) Sample  $l_t^{(i)} \sim p(l_t | l_{t-1}^{(i)})$
- (7) Weight of  $l_t^{(i)}$  as  $\tilde{w}_t^{(i)} = p(o_t | l_t^{(i)})$
- (8)  $C_t = C_t \cup \{(l_t^{(i)}, \tilde{w}_t^{(i)})\}$
- (9) *Phase Three.* Filtering
- (10)  $C_t' = \{(l_t^{(i)}, \tilde{w}_t^{(i)}) | (l_t^{(i)}, \tilde{w}_t^{(i)}) \in C_t \text{ and } \tilde{w}_t^{(i)} > 0\}$
- (11) Normalize the weight of valid samples  $w_t^i = \tilde{w}_t^i / \sum_{i=1}^N \tilde{w}_t^i$
- (12) Set the average of the samples as the blind node location.

( $i$ ) is the index of samples,  $o$  is observation at current time, and  $C$  is the sample set.

The filtration efficiency of the SMC localization scheme is mostly affected by anchor node density in the neighborhood. For example, under low anchor node density, a blind node is not always able to identify three anchor nodes in the first hop and second hop, especially when the sensor moves with high velocity; this process occurs because the first hop neighbors that communicate with radio range  $R$  are unable to identify within its range the second hop sensor that communicates with radio range  $2R$ .

*Hybrid Schemes (SMC and Hop Distance).* The multihop version of Monte Carlo localization (MMCL) [49] improves localization accuracy and reduces the dependence on anchor nodes by utilizing the DV-hop technique on MCL. MMCL measures the average hop distance between anchor nodes and then uses MCL to estimate blind node location. The DV-hop schemes have two drawbacks. First, these schemes need

a uniform distribution of anchors to achieve high accuracy. Second, broadcasting the location information of anchor nodes to multiple hops increases the communication cost.

The hybrid scheme presented in HMCL [50] utilizes hop distance and the SMC technique to improve localization accuracy. The sample area is constructed over the intersection area between anchor boxes. The anchor boxes are formed over the midpoint between anchor nodes. This scheme can reduce the size of a sample area and improve the localization accuracy through a virtual anchor node. The disadvantage of this scheme is that additional computation is required to estimate the distance and angle between the anchor node and the virtual anchor node.

*3.3. Hybrid Localization Scheme (Range-Free and Range-Based).* The combination of range-based and range-free schemes can improve localization accuracy in WSNs. The RSSI is a simple range-based scheme that measures the distance between two nodes by evaluating the signal strength indicator without additional hardware. Signal strength declines over distance. Hence, the RSSI utilizes this phenomenon to measure distance in the localization process. Consequently, communication and computational costs are reduced in the SMC technique [51].

The range-based MCL (RMCL) scheme combines range-based and range-free schemes during the localization process to overcome the high radio measurement error that reduces localization accuracy in range-based schemes. RMCL is a hop distance scheme that maintains the hop count and measurement range at a minimum for each anchor node. However, broadcasting the minimum measurement range for each anchor node increases the communication cost in this hop distance method. Moreover, computing the weights in RMCL is a complex task [52, 53].

The Monte Carlo box localization algorithm based on RSSI (MCBBR) [54] uses a reference genetic algorithm (linear crossing and rectangular crossing) to enhance the localization accuracy of the RMCL scheme and RSSI observation to optimize the sample area. In MCBBR, the localization accuracy is determined with the following four steps, namely, constructing the sampling box, establishing the sample number, optimizing the sample, and estimating the location. The real implementation of RMCL [52, 55] shows that the RSSI improves the accuracy of personal location inside an operation environment. Another improvement of RMCL [53] involves the use of SMC to increase localization accuracy when the range measurement has high variation; this improved scheme also utilizes range measurement to reduce computational costs.

The log-normal statistical model is used in the RSS-based Monte Carlo scheme (RSS-MCL) to improve localization accuracy. The RSS amount is used in the movement model and observation model; in the filter stage, the RSS observation is used to measure the distance between the sample and the anchor nodes. The invalid samples are filtered out without additional calculation. RSS-MCL can reduce the computational and communication costs in the filter stage. However, RSS-MCL suffers from high computational cost in

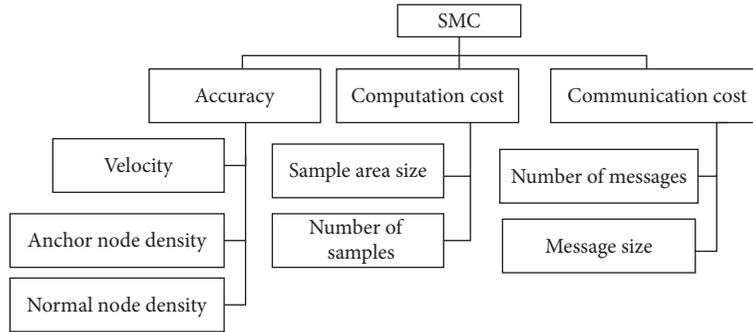


FIGURE 2: Evaluation parameter in SMC localization scheme.

the sample stage because the log-normal model is embedded with complex equations [52].

In real-world applications, range measurement is affected by path loss, fading, and shadowing phenomena. Hence, radio range can be protected by environmental factors, such as obstacles, rain, wind, and humidity; it can also be affected by the indoor environment. However, the range noise of the RSSI minimizes localization accuracy. Other studies [21, 56] presented the SMC scheme to enhance the localization accuracy associated with the noise measurement amount.

#### 4. Evaluation Parameters in SMC Localization

The SMC localization in mobile WSNs is mainly evaluated according to localization accuracy, computational cost, and communication cost, as presented in Figure 2.

**4.1. Localization Accuracy.** Localization accuracy is the most important parameter of WSNs. A high level of localization accuracy can help decision-makers to identify the precise location and coverage area of data. Localization accuracy can be measured with the variance between a real location and an estimated location, as shown in (2). For simplicity during the simulation test, the SMC technique is employed with the assumption that the anchor nodes know their real locations without error at all times.

$$\text{Localization accuracy} = \frac{1}{n} \sum_{i=1}^n \|e_i - l_i\|, \quad (2)$$

where  $n$  is the number of sensor nodes,  $e_i$  is the estimated location, and  $l_i$  is the real location. The error in the equation is given in terms of radio range and is thus divided by the sensor radio range.

The localization error in the initial step is reduced quickly when the new observations arrive. In the stability step, the localization error is maintained at around the same mean error. Thus, the effects of mobility and connectivity are in equilibrium. The localization accuracy of the SMC technique is mostly affected by the movement velocity, anchor node density, normal node density, and degree of irregularity.

**The Movement Velocity.** The mobility of sensor nodes can maximize the benefits of WSNs in various aspects. This

mobility allows sensors to communicate with a large number of neighboring anchor nodes. Hence, localization accuracy can be improved with the minimum number of anchor nodes. Mobility also conserves energy and prolongs network lifetime by changing routing paths [57]. Static WSNs use the same routing path, through which messages are sent and received frequently, even though the sensor is near the sink node; this frequency exhausts energy and causes network partition [27, 58, 59]. In real-world applications, the mobility of sensor nodes allows animals to be traced in zoos and patients to be monitored in hospitals, in addition to their other applications. However, this mobility presents an additional challenge in the handshaking case, in which the sensor is outside the neighbors' range to transmit and receive data [60, 61].

The mobility model is classified into three categories, namely, controlled, predefined (map), and random. The details of these categories are explained in [62]. In most schemes, the SMC technique is used to select a random waypoint model to transmit nodes. The waypoint model is a simple and independent model. Moreover, the sensor node can choose its new direction and velocity randomly without exceeding its maximum velocity [63]. The pause time is set to 0 in most schemes; this zero pause time allows the sensor to move without stopping [64].

The velocity of a sensor node affects localization accuracy differently. A sensor node with a low level velocity achieves the highest localization accuracy because this node is still in the range of the sample from the previous location, which this node reuses to estimate a new location accuracy. A sensor node with a high level velocity can exert a negative effect on localization accuracy if it moves far from the sample in the previous location and becomes unreachable. However, a high velocity guide sensor explores additional areas per time slot.

**Anchor Node Density.** The localization accuracy of all schemes can be enhanced with the increase in anchor node density in the region. A high number of anchor nodes allow the broadcast of many observations throughout the region. However, as the density of anchor nodes increases, the dependence on the global positioning system (GPS) and the extra overlap between anchor nodes increase as well. The extra overlap between anchor nodes is undesirable because it produces a redundant sample without improving localization accuracy. Moreover, the high density of anchor nodes limits the sample

area. A narrow sample area requires additional time for blind nodes to generate proper samples. The SMC technique addresses these drawbacks by employing a high number of anchor nodes in the region to maintain a high localization accuracy. In the literature, some schemes such as Monte Carlo localization (MCL) [26] and Monte Carlo localization boxed (MCB) [65] are fully dependent on the information of anchor node location, whereas others combine both anchor and normal nodes in the localization process.

*Normal Node Density.* The information on normal node location can be used during the positioning process to enhance localization accuracy and reduce the dependence on anchor nodes. The utilization of normal nodes can enhance localization accuracy in two ways. First, normal nodes retransmit the location information of the anchor node to its neighbors. Second, the location information of the normal nodes is used in the localization process; using this information in the sample step narrows the sample area and filters out the invalid samples in the filter step. However, the use of normal node location in the localization process is susceptible to error and significant communication cost in the network. Therefore, this localization process requires a precise and lightweight method.

*Degree of Irregularity.* The variation of the radio range between sensors leads to communication failure, which degrades the localization accuracy of WSNs [66, 67]. For simplicity, radio range is assumed to be a full circular range in most of previous schemes' simulation experiments. However, this assumption does not present the actual radio range in real-world applications; in reality, radio range is affected by sensor characteristics, such as antenna direction and sensor power, and by the types of transmission media, such as humidity, temperature, obstacles, and wind speed. These factors can distort radio range at different degrees.

*4.2. Computational Cost.* Computational cost is quantified from the iteration to generate enough valid samples in each time slot. The main parameters that affect computational cost are the size of the sample area and the number of samples. The sample and filter stages are repeated until enough valid samples are found; this process is costly because it wastes additional power and delays the localization process.

High velocity and high anchor node density negatively affect sample efficiency in the following ways. A high velocity maximizes the sample area. Thus, the sample generation and filtering steps are repeated several times to draw enough valid samples for a large area. A high anchor node density narrows the sample area. Hence, the generation and filtering steps are repeated to generate dissimilar samples.

*Sample Area Size.* In the literature, various strategies are used to draw samples. An example is the random generation of a sample over a previous sample bounded by a circle with a radius equal to the maximum velocity and anchor node bounded box. However, the shape of the sample area is irregular and is mostly affected by the number of anchor nodes in the neighborhood.

*Number of Samples.* The main idea of the SMC technique is to estimate the location of blind nodes by averaging the weighted samples (or particles). Therefore, the number of valid samples is an important parameter in localization accuracy. A large number of samples can slow down the localization process by repeating the generation and evaluation steps. Thus, a typical maximum number of samples is set to 50 [26].

The size of the sample area depends on the anchor node density in the first hop and second hop and on maximum velocity. A large number of anchor nodes in the neighborhood equates to a narrow sample area, and vice versa. Drawing a large number of samples in a narrow region is a critical issue because an additional calculation must be performed to remove redundant and closed samples. A large number of samples are required to cover a large sample area. Therefore, a constant number of samples do not represent a sufficient solution for all sizes of sample areas.

The simulation results for different schemes are presented in Table 3; 50 samples are enough to estimate an accurate location. Accordingly, most of the studies in the literature used 50 samples as the maximum number of samples, whereas other studies used an adaptive approach based on the sample area to set the number of samples. Nevertheless, the relation between the number of samples and the sample area is a challenging issue in WSNs.

In the SMC method, drawing valid samples involves the following two steps: (1) drawing candidate samples and (2) evaluating candidate samples. Drawing candidate samples is more costly than evaluating them [68]. Typically, sample efficiency is affected by the number of valid samples and the bounded area of the samples. Hence, a direct relationship exists between the number of samples and the sample area.

Sample evaluation is a measurement of the distance between two points or a comparison between the distance and its predefined value (the communication radio range  $R$ ). The operation cost for measuring the distance between two points is approximately 100 times that for comparing distance and its predefined value, as shown in [68], because the sample generation is repeated until the sample overcomes the anchor node communication radio range.

*4.3. Communication Cost.* The main purpose of the range-free localization scheme is to reduce the dependency on hardware by utilizing network connectivity during the estimation of blind node location. The estimation process requires network connectivity to broadcast messages from sensor nodes. Therefore, communication cost is computed with the number of messages broadcasted during the localization process [26]. The number of messages is affected by the number of anchor nodes and normal nodes used in the localization process. The size of the message also affects communication cost.

*Number of Messages.* In the SMC method, the anchor nodes broadcast their location information to the first hop and second hop; the normal nodes forward these messages to their neighbors. The number of messages that are broadcasted is a significant parameter during the localization process

because blind nodes need enough location information to estimate their location. However, a large number of messages may include redundant and closed samples.

The message of location information is categorized into two types according to its content. The first type of message contains the location coordinate, and the second type contains the sample. The coordinate message commonly defines the exact location of an anchor node on the Cartesian plane; the sample message contains the potential coordinate of the normal node on the Cartesian plane. The sample message can improve localization accuracy, but it increases the communication cost. Nevertheless, the relation between communication cost and localization accuracy is a challenging research area in WSNs.

**Message Size.** The size of messages transmitted is not fixed in SMC schemes, as presented in [69]. The anchor message contains the IP header, transmitter ID, anchor location, and number of hops. The standard size of an anchor message is 34 bytes in all schemes. By contrast, the size of a normal node message varies between schemes.

The most vital parameter in the localization process is the localization accuracy. The precise location can increase the confidence on the sensed data and the data originality. The achievement of high localization accuracy is difficult in terms of increasing the computation and communication cost. Thus, the optimal solution should maintain high accuracy with low computation and communication cost. To investigate and highlight this issue, localization schemes in the literature are classified based on these parameters.

## 5. State-of-the-Art SMC Localization Schemes

Monte Carlo localization (MCL) scheme is pioneered from the SMC schemes; in MCL, time is divided into discrete time slots, the pause time is set to 0, and all sensors move per time slot. After each movement, the node estimates its new location by utilizing the new observation from the anchor nodes in the neighborhood. Therefore, the sample and filter steps are repeated until the sensor collects enough valid samples. The weaknesses of MCL are that it requires high anchor node density to achieve sustainable accuracy and that it uses slow sampling method. The sample and filter steps are repeated up to 1000 times per each sample in MCL.

Dual and mixture MCL schemes improve the accuracy of MCL by inverting the probability function in the dual Monte Carlo scheme during the sample and filter steps [70]. The disadvantages of the dual Monte Carlo scheme are high computational cost and low sample efficiency. The authors slightly improved the sample efficiency in the mixture Monte Carlo scheme by mixing dual Monte Carlo samples and MCL samples. The negative effect of the mixture Monte Carlo scheme is that it has a lower accuracy than the dual Monte Carlo scheme.

The study [71] presented  $MSL^*$  and MSL (mobile sensor network localization and static sensor network localization) to improve the accuracy of MCL. The  $MSL^*$  scheme uses the location information of both anchor and normal nodes from the first hop and second hop. The location information

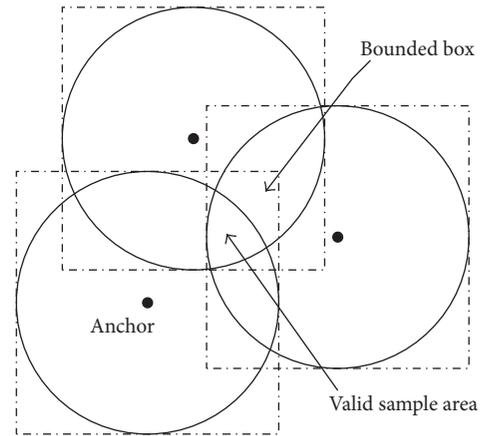


FIGURE 3: Bounded area of valid sample area MCB scheme.

contains the samples in the current time slot and their weights. However, broadcasting all node samples increases the communication cost. To reduce the communication cost, MSL is used to broadcast only the location coordinates of the anchor and normal nodes. This strategy reduces the communication cost and localization accuracy.

The  $MSL^*$  scheme adds the additional parameter of maximum velocity ( $\alpha = 0.1R$ ) in the sample generation to satisfy static networks. Each normal node sample in  $MSL^*$  has a partial weight in the range of zero to one; the anchor node sample maintains a weight value of 1 at all times. The node keeps its sample on the basis of its weight. Weight is estimated with a power function according to the number of normal nodes in the neighborhood. The node uses the close neighbor's samples to evaluate its samples. Hence, this node is greatly affected by the number of nodes in the neighborhood. Moreover, the power function in  $MSL^*$  entails a higher computational cost than the distance measurement between two point methods. The broadcasting of anchor, normal nodes samples, and their weight in  $MSL^*$  highly increase the communication cost.

In our previous LCC scheme (low communication cost) [72], the communication cost of  $MSL^*$  is reduced by selecting the closed normal nodes in the neighborhood instead of selecting all normal nodes as in  $MSL^*$ . LCC reduces the communication cost of  $MSL^*$  by 18% and maintains the same localization accuracy of  $MSL^*$ .

The Monte Carlo localization boxed (MCB) scheme uses the bounded box for each anchor node in the first and second hops to improve the sampling efficiency of MCL. The box is drawn around the node center with radii of  $R$  and  $2R$  in the first and second hops, respectively, as in Figure 3. The valid sample area is restricted on the intersection area between the anchor node boxes. Unlike MCL, MCB effectively improves sampling efficiency by bounding the valid sample area. Hence, the sample and filter steps require 100 repetitions to generate the valid sample. The number of anchor nodes in the neighborhood and the maximum velocity affect the shape of the sample area. The shape of the sample area is irregular in MCL; thus, a complex calculation is needed to determine

the bounded area. However, such calculation is impossible in sensor nodes. For simplicity, the box surrounding the sample area is used to assess the shape of the sample area, as shown in Figure 3. This implementation improves the sampling efficiency of MCL by 93% and maintains the same accuracy level as that of MCL.

The weighted Monte Carlo localization (WMCL) scheme improves localization accuracy by utilizing the location information of normal nodes [68], in addition to that of anchor nodes, as in MSL\*. The WMCL scheme reduces the size of the sample area and improves the sample efficiency of MCB by employing the two-hop anchor node neighbors' negative effect and normal node location information. The estimated location of the normal node contains a fraction of error. To overcome this challenge, the normal nodes utilize the maximum location error in  $x$ -axis and  $y$ -axis for bounding the sample area. Thus, the sample area in WMCL is bounded by both the anchor node constraints and normal node location information. The normal nodes estimate the errors in both axes. In filtering out the invalid samples efficiently, the weight for the anchor node samples is set to 1 at all times, and the partial weight for the normal node is set in the range of 0 to 1.

The partial weight for the normal node samples in WMCL is calculated as follows. First, the distance between all sensors samples in the neighborhood are estimated. Second, the intersection of the bounded box is utilized to reduce the communication cost of the first step. Finally, the radio range, maximum velocity, and maximum localization error from the previous time are utilized. These processes filter out the invalid samples more efficiently than MSL\* and quicken the sampling step. Unlike MCB, WMCL reduces the sample area by 78% and improves the sampling efficiency by up to 95%. Moreover, WMCL uses the normal node location information in the sample and filter steps, whereas MCB utilizes only the anchor node location information in the filter step.

The negative effect of two-hop anchor nodes in the literature is defined as follows: "node  $x$  is not within distance  $d$  of node  $y$ ." Range-free schemes utilize this definition ("node  $x$  is not within the radio range of  $y$ ") to enhance localization accuracy. The negative effect of two-hop anchor nodes and normal node location information can enhance localization accuracy and sample efficiency by 87% and 95%, respectively, as in WMCL. The shadow area in Figure 4 can be ignored without losing any valid samples; this fact can be explained as follows.  $q$  is assumed to be the two-hop anchor node for normal node  $p$ . Thus, the shadow area does not contain  $p$  because, otherwise,  $q$  is the one-hop neighbor of node  $p$ . The negative effect of two hops is a critical and precise issue. For example, if the distance between node  $p$  and  $q$  is underestimated, then the negative constraints can reduce localization accuracy. On the contrary, if the distance between node  $p$  and  $q$  is overestimated, then the practical location may be lost.

The movement direction of anchor nodes between the previous time slot and the current time slot is used in the constraint rule-optimized Monte Carlo localization (COMCL) scheme [73]. COMCL utilizes the locations of anchor nodes

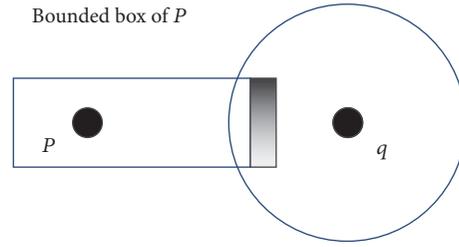


FIGURE 4: Improve the size of the bounded box: the shadowed area should be cut.

in the previous and current time slots to track the movement direction of these anchor nodes within the upper and lower bounds. COMCL classifies the location of the anchor nodes per time into two types: moving backward and moving forward in time. The location information in COMCL requires the following three steps: (1) construct the anchor node constraint, (2) construct the sample area, and (3) optimize and filter out the invalid samples. COMCL can involve more efficient and faster filtering steps than WMCL; nevertheless, it adds additional calculation. Each anchor node requires calculating the distance with neighbors and comparing the distance with upper and lower bounds to track the movement direction.

The range-based Monte Carlo boxed (RMCB) scheme compares and utilizes both range-based and range-free schemes to answer the question "when does range-based localization work better than range-free localization?" RMCB is suitable for both static and mobile WSNs with a heterogeneous radio range [74]. RMCB improves the sample area and efficiency in WMCL using a positive anchor node effect behind the negative effect used in WMCL. To ensure the efficiency of RMCB, the authors employed the same hardware devices for both RMCB and WMCL. The result shows that RMCB can improve WMCL in different parameters.

In [69], an improved MCL (IMCL) scheme was used to enhance the localization accuracy in MCL by adding constraints of movement direction in the previous schemes to the anchor and normal nodes. IMCL selects the normal nodes in the first hop's neighbors whose locations are constructed by the anchor node constraint. Moreover, IMCL employs the circular sector in the localization process to filter out the invalid samples. Each normal node divides the circular range into eight sectors; the longest sample sector is used to filter out the invalid samples. Computing the longest distance of samples and the angle of each sector increases the computational burden in IMCL and can delay the location estimation.

PMCB [75] (permeant Monte Carlo localization Boxed) scheme uses a time series to forecast the position of a blind node in case no anchor nodes exist in the neighborhood. Otherwise, SMC is used to estimate the location. The time series reduces the dependency on the anchor node. However, a recursive step is required to calculate the linear prediction coefficients in each time slot.

The orbit scheme improves localization accuracy by utilizing the characteristics of a star graph. The graph is

constructed with one root and five leaves to optimize the number of neighbors in the network. The orbit coordinates the neighbor's node constraint within the star graph to improve localization accuracy. The orbit scheme is highly affected by node density. In this scheme, five nodes may not constantly be discovered in the neighborhood [76]. Thus, the accuracy in the low normal nodes density will be reduced.

In [77], a Gaussian process regression was formulated with observations to improve localization accuracy. The observations on noise measurement, localization error, and previous distribution are correlated with the posterior predictive statistics. Hence, the posterior predictive statistics utilize MCL sampling and Laplace's method to improve localization accuracy. Laplace's method requires a complex calculation that is not applicable in thin device like sensor node.

In the sequential Monte Carlo-based localization algorithm (SMCLA), each sensor node maintains a table to store the localization parameters: estimated location, velocity, direction, and motion type at the current time slot. The blind node in the initial four steps moves according to the waypoint model. Then, the motion type is estimated by evaluating the velocity, acceleration, and movement direction to generate samples. Hence, the blind node stores the last four pieces of location information in the table with their time stamps. The disadvantage of utilizing the time stamp is that it requires additional hardware for the time synchronization between sensor nodes and the table data protect additional memory size [78].

The variation of radio range is evaluated during the localization process in the sequential Monte Carlo localization (SMCL) scheme. A perfect circular sector is used to simulate the radio range in most schemes. The radio range in real-world applications is affected by noise, path loss, shadowing, and physical phenomena. Hence, DOI (degree of irregularity) is used to check the variation of the radio range in the SMCL scheme. The updating stage is added to the SMC method to measure the effective factor of each sample in location estimation [79].

In [80], a sample adaptive Monte Carlo localization (SAMCL) algorithm was employed; in SAMCL, the sample area is divided into small bins, and each valid sample is assigned to one bin. The new samples are selected if they are acquired inside an empty bin. Otherwise, they are ignored. Thus, the number of samples is counted by bin numbers. The generated sample can be acquired several times in nonempty sample. Thus it requires repeating the sample generation. Moreover, the size of small bin is a critical issue. The number of bins can be maximized in the large size and minimized in the small size. Thus the localization accuracy will be affected.

The uniform sampling Monte Carlo localization (USML) scheme modifies the sampling strategy of SMC by dividing the sample area into small squares; this scheme selects the samples on the basis of their uniform distribution over a small square. The uniform distribution can reduce the time needed to generate random samples over the whole area. However, this uniform distribution does not represent the real state of all systems. Therefore, random generation can improve localization accuracy more efficiently than uniform distribution [81].

Reduce redundant messages and hop distance overhead using the back off-based broadcasting mechanism. This mechanism uses the following assumption in the RSSI: a node that is far from the sender has a signal strength that is too weak to select messages with a signal strength exceeding a predefined threshold [50].

In [82], the location information messages were used to improve failure detection. Generally, sensor nodes in WSNs exchange heartbeat messages to detect neighbors. These messages can be utilized for failure detection during the localization process. Hence, the compound between the localization process and failure detection can reduce the number of exchanged messages in networks.

Localization accuracy can be improved by combining SMC schemes and the genetic algorithm as in Genetic and Weighting Monte Carlo Localization (GWMCL) [83]. Crossover and mutation can be used to draw samples from a virtual anchor node. Hence, linear crossover and rectangular crossover are used to filter out invalid samples on the basis of the distance between the anchor node and the blind node.

The geometry of the intersection points between sensor nodes is used to bound the polygon shape; the shape is used to filter out the invalid samples [34]. However, the shape of the sample area is irregular and depends on the number and location of anchor nodes in the neighborhood. Hence, constructing the polygon is not easy in all cases.

*5.1. Schemes Utilizing a Single Anchor Node.* A single mobile anchor node (or online localization) is used to save scarce resources of sensor nodes and improve the localization accuracy of MCL. A blind node requests a location estimation from an anchor node. Thus, the anchor node calculates the location of the blind node and sends it back to the blind node.

Mobile-assisted Monte Carlo localization (MA-MCL) scheme uses one anchor node with high resources to localize static blind nodes. The anchor node moves randomly to collect arriver static and leaver static of blind observation. Then, invalid samples are filtered out according to the movement direction. After finding the blind node observation, the anchor node calculates the location and sends it back to the blind node [84].

Wireless node-based Monte Carlo localization (WNMCL) is another scheme that utilizes a single anchor node in the localization process. WNMCL divides the sample area into separate clusters. The closed clusters are merged, and the merging is repeated until the number of separate clusters is found. The center of the separate cluster is used as the estimated location of the blind node [85].

A single mobile anchor node with different types of blind node observation, such as connectivity, AoA, ranging, and a mixture of all of these, was utilized to estimate location [35]. The blind node collects at least the connectivity range of the first neighbor and sends this range to the anchor node when it arrives. The localization process occurs in the anchor node side; the location is sent back to the blind node.

Utilizing a single anchor node with high resources in the localization process can save scarce resources in sensor nodes and avoid time synchronization. Moreover, security can be

TABLE 2: Summary of localization schemes categories.

Localization category	Dependent on hardware	Scalability	Accuracy	Noise	Environment	Cost
Range-based	Full	Low	High	High	Outdoor	High
Range-free	Partial	High	Low	Low	Indoor	Low
Hybrid	Partial	High	Medium	Medium	Indoor	Low

improved by securing a single anchor node. Nevertheless, the use of a single mobile node increases the overhead over the beacon node and maximizes the probability of network congestion. Moreover, the noise and overload of a single anchor node range can degrade the localization accuracy of the whole network.

*5.2. Schemes Utilizing MCL in Target Tracking.* The MCL scheme enhances target tracking by estimating target locations [86, 87]. The novel Monte Carlo-based tracking (NMCT) scheme utilizes the perpendicular bisector zoning technique and triangulation assumption in the point in triangle (PIT) scheme to gather and check the blind nodes within or outside the anchor node triangle [88]. The perpendicular line is used to find the bisector area and check the closeness of the anchor nodes in the neighborhood. Therefore, the possible location of the blind node can be estimated from the anchor node pairs in the PIT and perpendicular bisector line [89]. Therefore, the valid sample area can be bounded, and the invalid samples can be filtered out efficiently. The weakness of NMCT is that it assumes that anchor nodes are static nodes and that normal nodes are mobile nodes.

Oriented tracking-based Monte Carlo localization (OTMCL) scheme utilizes the movement orientation in the sample step to improve MCL accuracy [90]. The angle of the movement sector is calculated on the basis of the elaboration between the locations in the previous and current times. The drawback of OTMCL is the need to constantly find enough valid samples. Thus, OTMCL uses the bounded box in MCB to generate samples.

The binary detection Monte Carlo localization (BD MCL) scheme utilizes the binary assumption in MCL to examine the node with the maximum range or outside range [91]. BD MCL maintains and records the interval time of each mobile sensor in the range. Hence, the mobile sensor with a large time interval has a high weightage sample. The use of the time interval requires the anchor node to synchronize the time between nodes; this synchronization may not be applicable in thin devices.

The movement continuity phenomenon of mobile sensors was used in [36] to estimate locations and movement directions. The study proposed the use of the linear prediction method and required the normal node to maintain the location information from four previous time slots. In this method, the sample area is divided into separate posterior density function regions on the basis of the movement direction in the previous time slot. However, maintaining four previous locations increases memory usage. Moreover, the network needs a long period to stabilize.

State-of-the-art SMC localization schemes are presented to highlight the advantage and disadvantages of each scheme. The SMC technique is a recursive method that requires repeating the sample and filter steps. The sampling efficiency is improved by restricting the sample area. The bounded box method can optimize sample area size and improve the sampling efficiency more than other methods. The localization accuracy is mostly improved by utilizing the normal node location information. Nevertheless, the use of normal node location information can highly increase the communication cost.

## 6. Comparison of Range-Free SMC Localization Schemes

Localization schemes are categorized on the basis of additional hardware requirement, scalability, accuracy, noise, operation environment, and cost, as shown in Table 2. Among all schemes, the range-based one achieves the highest accuracy despite being fully dependent on special hardware. This scheme is followed by the hybrid scheme that utilizes the network connectivity in noise and the RSSI assumption in the normal case. The range-free scheme achieves the lowest accuracy; location is estimated using network connectivity. SMC locations schemes can be compared in terms of localization accuracy, computational cost, communication cost, a number of samples, dependency on anchor nodes, and network type, as summarized in Table 3.

*6.1. Comparison of the Localization Accuracy.* Accuracy is a vital and challenging issue in the localization process. Achieving high accuracy in SMC schemes can be achieved through high anchor nodes density. There are various schemes that fully depend on anchor nodes such as MCL, dual and mixture MCL, MCB, and PMCB, and there are others that combine anchor and normal nodes location information like MSL\*, WMCL, RMCL, COMCL, IMCL, and Orbit. Relying on anchor nodes increases the dependency on the hardware (each anchor node requires a GPS device) which increases the power consumption, cost, and size, whereas using normal nodes can increase the communication and computational cost. Moreover, location information of normal nodes is an estimated location that embedded with the fraction of error.

An efficient filtration method of invalid samples can significantly improve the accuracy in SMC technique. There are different filtration methods used in various SMC schemes. The MCL, dual and mixture MCL, and PMCB schemes draw a circle over the previous sample to observe whether it satisfies the anchor nodes constraints in the current time. In case no

TABLE 3: SMC localization scheme classification.

Studies	Filteration method	Sample method	Sample area shape	Generation type	Number of samples	Dependent on anchor nodes	Accuracy	Computation cost	Communication cost	Network type
MCL	Distance	Circle intersection	Irregular	Random	Constant	Full	Low	High	Low	Mobile
Dual MCL	Distance	Circle intersection	Irregular	Random	Constant	Full	Low	High	Low	Mobile
MCB	Bounded box	Bounded box	Regular	Random	Constant	Full	Low	Medium	Low	Mobile
MSL*	Closeness	Circle intersection	Irregular	Random	Constant	Partial	High	High	High	Mobile/static
LCC	Closeness	Circle intersection	Irregular	Random	Constant	Partial	High	High	Medium	Mobile/static
WMCL	Bounded box	Bounded box	Regular	Random	Constant	Partial	High	Low	Medium	Mobile/static
COMCL	Bounded box	Bounded box	Regular	Random	Constant	Partial	High	Low	High	Mobile
RMCL	Bounded box	Bounded box	Regular	Random	Constant	Partial	High	Low	Medium	Mobile/static
IMCL	Distance	Circle sector	Irregular	Random	Dynamic	Partial	High	Medium	Medium	Mobile
Orbit	Bounded box	Star graph structure	Regular	Regular grid	Constant	Partial	High	High	High	Mobile
HMCL	Bounded box	Virtual anchor node	Regular	Random	Constant	Full	Medium	Medium	Low	Mobile
PMCB	Bounded box	Time series/bounded box	Regular	Random	Dynamic	Full	Medium	Medium	Low	Mobile
MCBBR	RSSI/bounded box	Genetic algorithm	Regular	Random	Dynamic	Full	Low	Medium	Low	Mobile
SAMCL	Bounded box	Bin counter	Regular	Sequential	Dynamic	Full	Low	Medium	Low	Mobile
USML	Maximum location error	Uniform distribution (small squares)	Regular	Sequential	Constant	Partial	Medium	High	High	Mobile
[34]	Convex polygon	Range geometry	Irregular	Random	Constant	Full	Low	High	High	Mobile
GWMCL	Genetic algorithm	Virtual anchor node weight	Irregular	Crossover operator	Constant	Full	Low	High	High	Mobile
MA-MCL	Arrivers and leavers (single anchor node)	Square with constant length	Irregular	Random	Constant	Full (single anchor node)	High	Low	High	Static/mobile
[35]	Single anchor movement direction	Critical region	Irregular	Random	Constant	Full (single anchor node)	High	Low	High	Static/mobile
NMCT	Triangular area	Circle radius	Regular	Random	Constant	Full	Low	High	High	Static/mobile

TABLE 3: Continued.

Studies	Filtration method	Sample method	Sample area shape	Generation type	Number of samples	Dependent on anchor nodes	Accuracy	Computation cost	Communication cost	Network type
OTMCL	Orientation variance	Sampling sector	Irregular	Random	Constant	Full	Low	High	High	Mobile
BD MCL	Detection time	Binary detection	Irregular	Random	Constant	Full	Low	Medium	High	Mobile
[36]	A posteriori density Distribution regions	Linear prediction method	Irregular	Random	Constant	Full	Low	High	High	Mobile

Computation cost: high, requiring 1000 times to generate sample like MCL; medium, requiring 100 times like MCB; and low, less than 100 times like WMCL.

Communication cost: high, broadcasting all samples of anchor and normal nodes; medium, broadcasting all anchor node samples and limited number of normal nodes samples.

Accuracy: the results of real experiment in WMCL and RMCB are used as benchmark.

anchor nodes exist in the neighbors, PMCB uses a time series to forecast location [75]. The bounding of valid sample area in this method is an irregular shape. It is constructed by the intersection between radio range circles of anchor nodes. For this, the distance between each sample and anchor nodes in the first and second hops is utilized to check whether the sample satisfies the anchor nodes constraints. The main drawbacks of this method are that it highly depends on anchor node density and that it uses inefficient and slow filtration methods.

The MSL\* [71] uses the closeness value of the sample to filter out low weight samples. This method can improve the localization accuracy but produces a high communication between the neighbors. Each node broadcasts its samples and samples weight in each time slot to the neighbors in the first and second hops. The communication cost is highly increased in this scheme but, in our previous research named LCC, the communication cost is reduced by selecting the normal nodes that have high intersection elements with the normal nodes in the neighbor.

The MCB scheme uses the intersection area of the bounded box over the anchor nodes to restrict the valid samples area. The shape of the box is regular; thus, it can easily filter out the invalid samples. Nevertheless, this shape contains an invalid area in some corner. WMCL narrows the bounded box by using the negative effect of two-hop anchor and normal nodes location information to optimize the sample area and enhance localization accuracy. Furthermore, RMCB utilizes the positive effect of the two-hop anchor and normal nodes to optimize the sample area and improve the accuracy in WMCL. It should be noted that WMCL uses the normal node location information in the sample and filter steps, whereas MCB utilizes only the anchor node location information in the filter step. The restriction on the sample area is a challenging issue.

IMCL [69] employs the longest circular sector to filter out the invalid samples and a star graph with one root and five leaves is used in Orbit scheme to improve localization accuracy [92]. The circular sectors method requires additional computation to find the angle and longest sectors, and finding five nodes in the neighbor is inapplicable in each time slot.

The MCL sampling and Laplace's method use the statistical posterior prediction to filter out invalid samples. However, the improvement of accuracy is minimum and requires high computation [77]. Storing posterior location information of estimated location, velocity, direction, and motion type in the table needs more memory and can slow the localization process in SMCLA [78]. Another localization scheme is genetic algorithm implemented in SMC technique to filter out invalid samples [83]; the genetic algorithm requires large data and high execution time, so it is unsuitable for thin devices like sensor node.

The fast method and precious filtration of invalid samples are essential in mobile WSNs location estimation. Filtration within the irregular shape can be tedious, as it requires repetition of the filtration and sampling steps for several times to generate the valid samples. The bounded box method can constrain the sample area fast and improve the accuracy at the same time. The circular sector and star graph require

high normal node density in the neighbor which may not exist all the time. Statistical and genetic methods require large memory and instruction for execution. Thus, they are incompatible with a thin device like sensor node.

*6.2. Comparison of Computation Cost.* A fast location estimation is desirable in mobile WSNs location estimation. A slow sampling method can delay the movement and the sensor may move to other locations before generating enough valid samples. The computational cost is mainly counted by the number of iterations required to generate enough valid samples.

The amount of filtration is a function of both sampling method and shape of the sample area. MCL, Dual and mixture MCL and MSL\* generate samples randomly over previous samples within the sample area limited by a circle with a radius of maximum velocity. The sample and filter steps are repeated up to 1,000 times in some cases to find valid samples. This weakness is from the irregular shape of the sample area and filtration strategies. The MSL\* has advantages in keeping the highly weighted sample from the previous time slot. Thus, the new sample is generated over a low weighted sample. Dual and mixture MCL schemes are inverting the probability function in the dual Monte Carlo scheme during the sample and filter steps to improve localization accuracy [56]. The disadvantages of these schemes are high computational cost and low sample efficiency.

Bounding of sample area becomes more precise and the sampling efficiency improved by using anchor node bounded box intersection in the MCB scheme, in which the amount of filtration is reduced to 100 times per each sample. WMCL reduces the sample area by 78% and increases the sampling efficiency up to 95%. Moreover, WMCL uses the normal node location information in the sample and filter steps, whereas MCB utilizes only the anchor node location information in the filter step. RMCB improves the sample area and efficiency in WMCL by utilizing a positive anchor node effect behind the negative effect used in WMCL. The anchor node movement directions such as moving backward and moving forward are utilized in COMCL to maintain the sample area over WMCL assumption.

The aforementioned schemes used a static number of samples which is equal to 50 in different sample area size. In PMCB and IMCL schemes, the number of samples is based on the percentage of the sample area with respect to the maximum area of one anchor node in the neighborhood, as represented in (3). However, at least one anchor node is assumed to be in the first hop of the blind node neighbor in most simulations.

$$\text{Sample number} = \frac{(50 * (\Delta x) * (\Delta y))}{4R^2}, \quad (3)$$

where  $\Delta x$  and  $\Delta y$  are the height and length of the bounded box (sample area), respectively, and  $4R^2$  is the maximum area of one anchor node in the first hop.

In SAMCL scheme [77], the sample area is divided into small bins, and each valid sample is assigned to one bin; the number of samples is counted by bin numbers. Reference

[93] divided the sample area into small squares and the sample is distributed uniformly over the squares. The polygon shape property is used to construct the sample area in [34], where the location and number of anchor nodes are used to maintain the polygon shape.

Large number of iterations can slow the location estimation and consume more power. The main difficulty in the localization process is the shape of the sample area. Generating valid sample from the irregular shape is a tedious issue. The bounded box method can maintain the shape of the sample area to become a regular shape. Generating sample from such area is an easy thing and can reduce the number of iterations required to generate the valid sample. Dividing the sample area of the small bins or distributing sample uniformly over small squares is weak assumption and is unable to present the reality of the sample area; the using of random generation is more precise and can be more general. Adapting the number of samples with sample area size is an important thing but the most important issue is the efficiency of the sampling method.

**6.3. Comparison of Communication Costs.** The number of messages sent is the function of anchor node density in the schemes such as MCL, MCB, and dual MCL; and this number is equivalent to the anchor node number ( $A$ ) in the neighborhood. In  $MSL^*$  and MSL, the normal node location information is utilized along with the anchor node. The normal node in  $MSL^*$  broadcasts the samples in each time slot to the first and second hops, whereas the normal node in MSL only broadcasts its coordinates and not all samples. The communication costs of  $MSL^*$  and MSL are represented by  $(N * S + A)$  and  $(N + A)$ , respectively, where  $N$  is the number of normal nodes,  $S$  is the number of samples in each time slot (50 samples), and  $A$  is the number of anchor nodes in the neighborhood.

Among all schemes, the  $MSL^*$  scheme achieves the highest communication cost, whereas MCL achieves the lowest communication cost. In our LCC scheme, the communication cost in  $MSL^*$  is reduced by 18% by selecting the closed normal node in the neighborhood.

The assumption in  $MSL^*$  is adopted in WMCL and RMCL; the normal node broadcasts its sample to the first hop. WMCL and RMCL modify the sample with the information on message size; the location of the maximum error is defined in  $x$ -axis and  $y$ -axis. The COMCL scheme embeds the range of the bounded box from the previous time slot in the sample of the normal node. However, the communication cost of COMCL is 1.04 times higher than that of WMCL. The simulation results in WMCL show that the communication cost is more significantly affected by the size of the message than by the densities of the anchor and normal nodes.

$MSL^*$  broadcasts the message with information on the IP header, transmitter ID, estimated position, number of hops, and coordinates of 50 valid samples in the previous time unit; this information costs 634 bytes. The normal node messages in WMCL or BB (bounded box) combine the IP header, transmitter ID, estimated position, number of hops, valid samples, and maximum error in  $x$ -axis and  $y$ -axis; this

information costs only 46 bytes. The normal node message in the IMCL scheme has a size of 66 bytes by combining the IP header, transmitter ID, estimated position, number of hops, valid samples, and eight sectors. The MCL, MCB, and dual MCL schemes yield the lowest communication costs because they only utilize the anchor node location information. The details of bytes sent in each scheme per time slot are listed in [69].

The number of messages is also affected by the number of hops used in the localization process. Normally, SMC utilizes the sensor in the first and second hops. However, the sensor node in the second hop can maximize the communication, especially when normal node samples are used. For example,  $MSL^*$  uses the normal node samples of the first and second hops; each anchor node and normal node broadcast their respective samples in each time slot to the first and second hops' neighbors. Therefore,  $MSL^*$  requires a high communication cost.

Table 3 presents the comparison between SMC schemes based on effective parameters. The localization accuracy is the most important variable especially when sensors move in high speed. Most of the schemes improve the localization by utilizing normal nodes location information that highly increases the communication and computation cost.

## 7. Discussion of Future Works and Localization Issues

Range-based schemes achieve a higher localization accuracy than range-free schemes. However, range-based schemes are highly dependent on additional hardware that consumes a large amount of power and increases the size and cost of sensor nodes, especially in a dense deployment. The battery replacement of sensor nodes is difficult, particularly when nodes are in remote and hazardous areas. Moreover, communication range and hardware signals are affected by noise and obstacles. Therefore, range-based schemes are unsuitable for certain types of applications.

By contrast, range-free schemes estimate locations using network connectivity and without any additional hardware. A range-free scheme is a challenging area. The obstacle of SMC is its dependency on anchor node density and a high number of valid samples to estimate an accurate location. Repeating the sample and filtering stages several times is a time-consuming process. Furthermore, hop distance schemes require a uniform distribution of anchor nodes, and fingerprint schemes are time-consuming because expert personnel are required to create the offline database and update the database every time the environment changes.

Connectivity information may remain unchanged when sensor nodes move a small distance without establishing a new connection or disestablishing the previous connection. Therefore, we can define the lower bounds in range-free schemes as the average distance in which the sensor node can move with the same connectivity information between the previous time and the current time. In this case, localization accuracy degenerates for range-free schemes [94].

The localization accuracy of range-free schemes is a challenging research area. The localization error in SMC increases rapidly when the velocity of mobile nodes increases. A high velocity can change the topology of WSNs quickly. Therefore, WSNs require an adaptive mobility model to transmit sensor nodes efficiently. Another issue in the SMC localization process is the accuracy highly affected by anchor node density and number of samples.

Sample efficiency is a significant parameter in the SMC method. However, the repetition of the sample and filter steps for several times delays the localization process. Other significant parameters are the number of samples and sample area channeling. The shape of the sample area is irregular, and the bounded area is difficult to find. The number of samples of this area requires the highlight method, whereas the use of the bounded box is embedded with a high percentage of error.

Messages in WSNs consume scarce resources and waste sensor battery life. Hence, localization schemes require a lightweight algorithm to avoid additional, redundant messages.

## 8. Conclusion

The localization of mobile sensors is a key issue in WSNs. Specifically, an accurate location can maximize the benefits of WSNs. A high localization accuracy can be achieved through an efficient and lightweight scheme that is adaptable to sensor characteristics. Constructing an efficient scheme on the basis of the SMC method can improve the localization accuracy in dynamic systems, such as mobile sensors. In this study, we introduced a thematic taxonomy to classify the current SMC localization schemes. Moreover, we presented a comprehensive survey of state-of-the-art SMC schemes and classified them according to their localization requirements. The critical aspects of existing SMC localization schemes were analyzed to identify the advantages and disadvantages of each scheme. Furthermore, the similarities and differences of each scheme were investigated on the basis of important parameters, such as localization accuracy, computational cost, communications cost, and number of samples. We discussed the challenges and open research issues related to the parameters. The future work on the localization accuracy of range-free schemes can be improved by combining RSSI technology and SMC schemes. The RSSI can reduce computational and communication costs by utilizing the signal strength indicator.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

The authors thank the University of Malaya for the financial support (UMRG Grant RP036A-15AET) and facilities to carry out the work. Also, the authors express their deepest appreciation to the management and staff of Al-Quds Open University.

## References

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–105, 2002.
- [2] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [3] J. H. Ryu, M. Irfan, and A. Reyaz, "A review on sensor network issues and robotics," *Journal of Sensors*, vol. 2015, Article ID 140217, 2015.
- [4] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
- [5] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [6] E. Borgia, "The internet of things vision: key features, applications and open issues," *Computer Communications*, vol. 54, pp. 1–31, 2014.
- [7] H. Alemdar and C. Ersoy, "Wireless sensor networks for health-care: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2688–2710, 2010.
- [8] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pp. 88–97, ACM, September 2002.
- [9] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad Hoc Networks*, vol. 3, no. 3, pp. 257–279, 2005.
- [10] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.
- [11] Y. R. Al-Mayouf, M. Ismail, N. F. Abdullah et al., "Efficient and stable routing algorithm based on user mobility and node density in urban vehicular network," *PLoS ONE*, vol. 11, no. 11, Article ID e0165966, 2016.
- [12] M. P. Đurišić, Z. Tafa, G. Dimić, and V. Milutinović, "A survey of military applications of wireless sensor networks," in *Proceedings of the Mediterranean Conference on Embedded Computing (MECO '12)*, pp. 196–199, IEEE, June 2012.
- [13] O. A. Mahdi, A. W. Abdul Wahab, M. Y. I. Idris, A. Abu Znaid, Y. R. B. Al-Mayouf, and S. Khan, "WDARS: a weighted data aggregation routing strategy with minimum link cost in event-driven WSNs," *Journal of Sensors*, vol. 2016, Article ID 3428730, 12 pages, 2016.
- [14] M. Y. I. Idris, H. Arof, N. M. Noor, E. M. Tamil, and Z. Razak, "A co-processor design to accelerate sequential monocular SLAM EKF process," *Measurement*, vol. 45, no. 8, pp. 2141–2152, 2012.
- [15] C. F. García-Hernández, P. H. Ibarguengoytia-González, J. García-Hernández, and J. A. Pérez-Díaz, "Wireless sensor networks and applications: a survey," *IJCSNS International Journal of Computer Science and Network Security*, vol. 7, no. 3, pp. 264–273, 2007.
- [16] A. Pal, "Localization algorithms in wireless sensor networks: current approaches and future challenges," *Network Protocols and Algorithms*, vol. 2, no. 1, pp. 45–73, 2010.
- [17] S. Li, D. Lowe, X. Kong, and R. Braun, "Wireless sensor network localization algorithm using dynamic path of mobile beacon," in *Proceedings of the 17th Asia Pacific Conference on Communications (APCC '11)*, pp. 344–349, IEEE, October 2011.

- [18] S. P. Singh and S. Sharma, "Range free localization techniques in wireless sensor networks: a review," *Procedia Computer Science*, vol. 57, pp. 7–16, 2015.
- [19] W.-C. Lai, Y.-Y. Su, C.-M. Lee et al., "A survey of secure fingerprinting localization in wireless local area networks," in *Proceedings of the 12th International Conference on Machine Learning and Cybernetics (ICMLC '13)*, pp. 1413–1417, IEEE, Tianjin, China, July 2013.
- [20] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.
- [21] L. Mihaylova, D. Angelova, and A. Zvikhachevskaya, "Sequential Monte Carlo methods for localization in wireless networks," in *Advances in Intelligent Signal Processing and Data Mining*, pp. 89–118, Springer, 2013.
- [22] A. Doucet, N. De Freitas, and N. Gordon, "An introduction to sequential Monte Carlo methods," in *Sequential Monte Carlo Methods in Practice*, pp. 3–14, Springer, 2001.
- [23] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multitarget filtering with random finite sets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1224–1245, 2005.
- [24] M. Vasim Babu and A. V. Ramprasad, "Discrete antithetic Markov Monte Carlo based power mapping localization algorithm for WSN," in *Proceedings of the IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT '12)*, pp. 56–62, IEEE, Ramanathapuram, India, August 2012.
- [25] T. Li, M. Bolić, and P. M. Djurić, "Resampling methods for particle filtering: classification, implementation, and strategies," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 70–86, 2015.
- [26] L. Hu and D. Evans, "Localization for mobile sensor networks," in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*, ACM, Philadelphia, Pa, USA, 2004.
- [27] Z. Yang, C. Wu, Z. Zhou, X. Zhang, X. Wang, and Y. Liu, "Mobility increases localizability: a survey on wireless indoor localization using inertial sensors," *ACM Computing Surveys (CSUR)*, vol. 47, no. 3, article 54, 2015.
- [28] G. Han, H. Xu, T. Q. Duong, J. Jiang, and T. Hara, "Localization algorithms of wireless sensor networks: a survey," *Telecommunication Systems*, vol. 52, no. 4, pp. 2419–2436, 2013.
- [29] S. Gu, Y. Yue, C. Maple, C. Wu, and B. Liu, "Challenges in mobile localisation in wireless sensor networks for disaster scenarios," in *Proceedings of the 19th International Conference on Automation and Computing (ICAC '13)*, pp. 60–65, IEEE, London, UK, September 2013.
- [30] U. Nazir, N. Shahid, M. A. Arshad, and S. H. Raza, "Classification of localization algorithms for wireless sensor network: a survey," in *Proceedings of the International Conference on Open Source Systems and Technologies (ICOSST '12)*, IEEE, 2012.
- [31] E. Niewiadomska-Szynkiewicz, "Localization in wireless sensor networks: classification and evaluation of techniques," *International Journal of Applied Mathematics and Computer Science*, vol. 22, no. 2, pp. 281–297, 2012.
- [32] H. Suo, J. Wan, L. Huang, and C. Zou, "Issues and challenges of wireless sensor networks localization in emerging applications," in *Proceedings of the International Conference on Computer Science and Electronics Engineering (ICCSEE '12)*, pp. 447–451, IEEE, Hangzhou, China, March 2012.
- [33] L. Cheng, C. Wu, Y. Zhang, H. Wu, M. Li, and C. Maple, "A survey of localization in wireless sensor network," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 962523, 12 pages, 2012.
- [34] T. C. Henderson, E. Grant, K. Luthy, and M. Craver, "Precision localization in Monte Carlo sensor networks," in *Proceedings of the ISCA 18th International Conference on Computer Applications in Industry and Engineering*, November 2005.
- [35] R. Huang and G. V. Záruba, "Monte Carlo localization of wireless sensor networks with a single mobile beacon," *Wireless Networks*, vol. 15, no. 8, pp. 978–990, 2009.
- [36] S. P. Fan, Y. J. Wen, and L. Zhou, "An enhanced Monte Carlo localization algorithm for mobile node in wireless sensor networks," in *Applied Mechanics and Materials*, Trans Tech Publications, 2013.
- [37] C. Alippi and G. Vanini, "A RSSI-based and calibrated centralized localization technique for wireless sensor networks," in *Proceedings of the 4th Annual IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 301–305, IEEE, Pisa, Italy, March 2006.
- [38] K. Römer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 54–61, 2004.
- [39] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54–69, 2005.
- [40] F. Gustafsson and F. Gunnarsson, "Positioning using time-difference of arrival measurements," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, IEEE, Hong Kong, June 2003.
- [41] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AOA," in *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications (INFOCOM '03)*, IEEE Societies, 2003.
- [42] Q. Shao, H. Xu, L. Jia, and P. Li, "The research of Monte Carlo localization algorithm based on received signal strength," in *Proceedings of the 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '11)*, pp. 1–4, IEEE, Wuhan, China, September 2011.
- [43] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global Positioning System: Theory and Practice*, Springer, 2012.
- [44] H. Safa, "A novel localization algorithm for large scale wireless sensor networks," *Computer Communications*, vol. 45, pp. 32–46, 2014.
- [45] V. H. Perez-Gonzalez, D. Munoz-Rodriguez, C. Vargas-Rosales, and R. Torres-Villegas, "Relational position location in ad-hoc networks," *Ad Hoc Networks*, vol. 24, pp. 20–28, 2015.
- [46] K. Kaemarungsi and P. Krishnamurthy, "Analysis of WLAN's received signal strength indication for indoor location fingerprinting," *Pervasive and Mobile Computing*, vol. 8, no. 2, pp. 292–316, 2012.
- [47] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [48] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, "Localization from connectivity in sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, no. 11, pp. 961–974, 2004.
- [49] J. Yi, S. Yang, and H. Cha, "Multi-hop-based Monte Carlo localization for mobile sensor networks," in *Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor*,

- Mesh and Ad Hoc Communications and Networks (SECON '07)*, pp. 163–171, IEEE, San Diego, Calif, USA, June 2007.
- [50] H. Chen, F. Gao, M. Martins, P. Huang, and J. Liang, “Accurate and efficient node localization for mobile sensor networks,” *Mobile Networks and Applications*, vol. 18, no. 1, pp. 141–147, 2013.
- [51] F. Bandiera, A. Coluccia, and G. Ricci, “A cognitive algorithm for received signal strength based localization,” *IEEE Transactions on Signal Processing*, vol. 63, no. 7, pp. 1726–1736, 2015.
- [52] W. P. L. Cully, S. L. Cotton, and W. G. Scanlon, “Empirical performance of RSSI-based Monte Carlo localisation for active RFID patient tracking systems,” *International Journal of Wireless Information Networks*, vol. 19, no. 3, pp. 173–184, 2012.
- [53] B. Dil, S. Dulman, and P. Havinga, “Range-based localization in mobile sensor networks,” in *Wireless Sensor Networks*, pp. 164–179, Springer, 2006.
- [54] G. Li, J. Zhang, J. Chen, and Z. Xu, “A Monte Carlo box localization algorithm based on RSSI,” in *Proceedings of the 33rd Chinese Control Conference (CCC '14)*, pp. 395–400, IEEE, Nanjing, China, July 2014.
- [55] R. Al Alawi, “RSSI based location estimation in wireless sensors networks,” in *Proceedings of the 17th IEEE International Conference on Networks (ICON '11)*, pp. 118–122, IEEE, Singapore, December 2011.
- [56] K. Heurtefeux and F. Valois, “Is RSSI a good choice for localization in wireless sensor network?,” in *Proceedings of the 26th IEEE International Conference on Advanced Information Networking and Applications (AINA '12)*, pp. 732–739, IEEE, Fukuoka, Japan, March 2012.
- [57] S. Halder and A. Ghosal, “A survey on mobility-assisted localization techniques in wireless sensor networks,” *Journal of Network and Computer Applications*, vol. 60, pp. 82–94, 2016.
- [58] R. Silva, Z. Zinonos, J. S. Silva, and V. Vassiliou, “Mobility in WSNs for critical applications,” in *Proceedings of the 16th IEEE Symposium on Computers and Communications (ISCC '11)*, pp. 451–456, IEEE, July 2011.
- [59] E. Natalizio and V. Loscri, “Controlled mobility in mobile sensor networks: advantages, issues and challenges,” *Telecommunication Systems*, vol. 52, no. 4, pp. 2411–2418, 2013.
- [60] Y. Wang and H. Wu, “DFT-MSN: the delay/fault-tolerant mobile sensor network for pervasive information gathering,” in *Proceedings of the 25th IEEE International Conference on Computer Communications*, pp. 1–12, IEEE, Barcelona, Spain, April 2006.
- [61] F. Deniz, H. Bagci, I. Korpeoglu, and A. Yazici, “An adaptive, energy-aware and distributed fault-tolerant topology-control algorithm for heterogeneous wireless sensor networks,” *Ad Hoc Networks*, vol. 44, pp. 104–117, 2016.
- [62] F. Bai and A. Helmy, “A survey of mobility models,” in *Wireless Adhoc Networks*, p. 206, University of Southern California, Los Angeles, Calif, USA, 2004.
- [63] T. Camp, J. Boleng, and V. Davies, “A survey of mobility models for ad hoc network research,” *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, 2002.
- [64] J. Yoon, M. Liu, and B. Noble, “Sound mobility models,” in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, ACM, 2003.
- [65] A. Baggio and K. Langendoen, “Monte Carlo localization for mobile wireless sensor networks,” *Ad Hoc Networks*, vol. 6, no. 5, pp. 718–733, 2008.
- [66] G. Zhou, T. He, S. Krishnamurthy, and J. A. Stankovic, “Impact of radio irregularity on wireless sensor networks,” in *Proceedings of the 2nd International Conference on Mobile Systems, Applications, and Services*, ACM, Boston, Mass, USA, June 2004.
- [67] X. Wu, S. Tan, and Y. He, “Effective error control of iterative localization for wireless sensor networks,” *AEU—International Journal of Electronics and Communications*, vol. 67, no. 5, pp. 397–405, 2013.
- [68] S. Zhang, J. Cao, C. Li-Jun, and D. Chen, “Accurate and energy-efficient range-free localization for mobile sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 6, pp. 897–910, 2010.
- [69] J.-P. Sheu, W.-K. Hu, and J.-C. Lin, “Distributed localization scheme for mobile sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 4, pp. 516–526, 2010.
- [70] E. Stevens-Navarro, V. Vivekanandan, and V. W. S. Wong, “Dual and mixture Monte Carlo localization algorithms for mobile wireless sensor networks,” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '07)*, pp. 4027–4031, IEEE, Hong Kong, March 2007.
- [71] M. Rudafshani and S. Datta, “Localization in wireless sensor networks,” in *Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN '07)*, pp. 51–60, ACM, Cambridge, Mass, USA, April 2007.
- [72] A. M. A. Abu Znaid, M. Y. I. Idris, A. W. A. Wahab et al., “Low communication cost (LCC) scheme for localizing mobile wireless sensor networks,” *Wireless Networks*, vol. 23, no. 3, pp. 737–747, 2017.
- [73] Z. Wang, Y. Wang, M. Ma, and J. Wu, “Efficient localization for mobile sensor networks based on constraint rules optimized Monte Carlo method,” *Computer Networks*, vol. 57, no. 14, pp. 2788–2801, 2013.
- [74] T. Adnan, S. Datta, and S. MacLean, “Efficient and accurate sensor network localization,” *Personal and Ubiquitous Computing*, vol. 18, no. 4, pp. 821–833, 2014.
- [75] B. Soltaninasab, M. Sabaei, and J. Amiri, “Improving Monte Carlo localization algorithm using time series forecasting method and dynamic sampling in mobile WSNs,” in *Proceedings of the 2nd International Conference on Communication Systems, Networks and Applications (ICCSNA '10)*, pp. 389–396, IEEE, Hong Kong, July 2010.
- [76] S. Maclean and S. Datta, “Reducing the positional error of connectivity-based positioning algorithms through cooperation between neighbors,” *IEEE Transactions on Mobile Computing*, vol. 13, no. 8, pp. 1868–1882, 2014.
- [77] M. Jadhaliha, Y. Xu, J. Choi, N. S. Johnson, and W. Li, “Gaussian process regression for sensor networks under localization uncertainty,” *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 223–237, 2013.
- [78] A. Alaybeyoglu, “An efficient Monte Carlo-based localization algorithm for mobile wireless sensor networks,” *Arabian Journal for Science and Engineering*, vol. 40, no. 5, pp. 1375–1384, 2015.
- [79] W. Wang and Q. Zhu, “Sequential Monte Carlo localization in mobile sensor networks,” *Wireless Networks*, vol. 15, no. 4, pp. 481–495, 2009.
- [80] D. Fox, “Adapting the sample size in particle filters through KLD-sampling,” *The International Journal of Robotics Research*, vol. 22, no. 12, pp. 985–1003, 2003.
- [81] H. Mirebrahim and M. Dehghan, “Monte Carlo localization of mobile sensor networks using the position information of neighbor nodes,” in *Ad-Hoc, Mobile and Wireless Networks*, pp. 270–283, Springer, 2009.
- [82] L. G. L. Martins, R. C. Nunes, J. B. S. Martins, and L. L. De Oliveira, “A hybrid method to detecting failures in mobile sensor networks using localization algorithms,” in *Proceedings of*

- the IEEE 11th International New Circuits and Systems Conference (NEWCAS '13)*, pp. 1–4, IEEE, Paris, France, June 2013.
- [83] J. Luan, R. Zhang, B. Zhang, and L. Cui, “An improved monte carlo localization algorithm for mobile wireless sensor networks,” in *Proceedings of the 7th International Symposium on Computational Intelligence and Design (ISCID '14)*, pp. 477–480, IEEE, Hangzhou, China, December 2014.
- [84] G. Teng, K. Zheng, and W. Dong, “MA-MCL: mobile-assisted Monte Carlo localization for wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 2011, Article ID 671814, 8 pages, 2011.
- [85] A. Kurecka, J. Konecny, M. Prauzek, and J. Koziorek, “Monte carlo based wireless node localization,” *Elektronika ir Elektrotechnika*, vol. 20, no. 6, pp. 12–16, 2014.
- [86] O. A. Mahdi, A. W. Abdul Wahab, M. Y. I. Idris et al., “A comparison study on node clustering techniques used in target tracking WSNs for efficient data aggregation,” *Wireless Communications and Mobile Computing*, vol. 16, no. 16, pp. 2663–2676, 2016.
- [87] O. A. Mahdi, A. W. Abdul Wahab, M. Y. I. Idris, A. Abu Znaid, S. Khan, and Y. R. B. Al-Mayouf, “ESAM: endocrine inspired sensor activation mechanism for multi-target tracking in WSNs,” in *4th International Conference on Wireless and Optical Communications*, vol. 9902 of *Proceedings of SPIE*, International Society for Optics and Photonics, Beijing, China, June 2016.
- [88] Q. Niu, T. Huan, and P. Chen, “NMCT: a novel Monte Carlo-based tracking algorithm using potential proximity information,” *International Journal of Distributed Sensor Networks*, vol. 2016, Article ID 7061486, 10 pages, 2016.
- [89] L. Girod and D. Estrin, “Robust range estimation using acoustic and multimodal sensing,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1312–1320, IEEE, November 2001.
- [90] M. H. T. Martins, H. Chen, and K. Sezaki, “OTMCL: orientation tracking-based Monte Carlo localization for mobile sensor networks,” in *Proceedings of the 6th International Conference on Networked Sensing Systems (INSS '09)*, pp. 1–8, IEEE, Pittsburgh, Pa, USA, June 2009.
- [91] J. Li, X. You, J. Xia, and H. Li, “A MCL-based localization algorithm for target binary-detection of mobile nodes,” in *Advances in Future Computer and Control Systems*, pp. 567–572, Springer, 2012.
- [92] S. MacLean and S. Datta, “Reducing the positional error of connectivity-based positioning algorithms through cooperation between neighbors,” *IEEE Transactions on Mobile Computing*, vol. 13, no. 8, pp. 1868–1882, 2014.
- [93] H. Mirebrahim and M. Dehghan, “Monte Carlo localization of mobile sensor networks using the position information of neighbor nodes,” in *Proceedings of the 8th International Conference on Ad-Hoc, Mobile and Wireless Networks*, Springer, Murcia, Spain, September 2009.
- [94] Y. W. E. Chan and B. H. Soong, “A new lower bound on range-free localization algorithms in wireless sensor networks,” *IEEE Communications Letters*, vol. 15, no. 1, pp. 16–18, 2011.