remote sensing

Special Issue Reprint

High-Precision and High-Reliability Positioning, Navigation, and Timing

Opportunities and Challenges

Edited by
Zhetao Zhang, Guorui Xiao, Zhixi Nie, Vagner Ferreira and Giuseppe Casula

mdpi.com/journal/remotesensing

MDPI

High-Precision and High-Reliability Positioning, Navigation, and Timing: Opportunities and Challenges

High-Precision and High-Reliability Positioning, Navigation, and Timing: Opportunities and Challenges

Guest Editors

Zhetao Zhang
Guorui Xiao
Zhixi Nie
Vagner Ferreira
Giuseppe Casula

MDPI

Basel • Beijing • Wuhan • Barcelona • Belgrade • Novi Sad • Cluj • Manchester

Guest Editors

Zhetao Zhang
Hohai University
Nanjing
China

Guorui Xiao
Information Engineering
University
Zhengzhou
China

Zhixi Nie
China University of
Petroleum (East China)
Dongying
China

Vagner Ferreira
Hohai University
Nanjing
China

Giuseppe Casula
Istituto Nazionale di
Geofifisica e Vulcanologia
(INGV)
Bologna
Italy

Editorial Office
MDPI AG
Grosspeteranlage 5
4052 Basel, Switzerland

This is a reprint of the Special Issue, published open access by the journal *Remote Sensing* (ISSN 2072-4292), freely accessible at: https://www.mdpi.com/journal/remotesensing/special_issues/93HKEX8787.

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

Lastname, A.A.; Lastname, B.B. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-7258-2921-7 (Hbk)
ISBN 978-3-7258-2922-4 (PDF)
https://doi.org/10.3390/books978-3-7258-2922-4

Contents

About the Editors

Zhetao Zhang

Dr. Zhetao Zhang is currently an Associate Professor at the School of Earth Sciences and Engineering at Hohai University, Nanjing, China. He received a Ph.D. degree with distinction in Geodesy from Tongji University, Shanghai, China, in 2019. He was a Postdoctoral Fellow and Visiting Ph.D. student at the Hong Kong Polytechnic University and the University of Calgary, respectively. His primary research focuses on precise positioning and navigation with GNSS and multi-sensor systems under complex conditions, including challenging environments, low-cost devices, multi-source data, etc.

Guorui Xiao

Dr. Guorui Xiao is currently an associate professor at Information Engineering University, China. He received his B.Sc. degree from the School of Geodesy and Geomatics at Wuhan University in 2011 and his Ph.D. degree from the Geodetic Institute of Karlsruhe Institute of Technology in Germany. His current research focuses on multi-GNSS and multi-sensor integrated precise positioning and applications.

Zhixi Nie

Dr. Zhixi Nie received his PhD Geodesy and Geomatics Engineering from China University of Petroleum (East China). He conducted his postdoctoral research at the University of Calgary from November 2017 to May 2019. Currently, he is an associate professor in the College of Oceanography and Space Informatics at China University of Petroleum (East China). His research focuses on GNSS precise positioning algorithms and applications.

Vagner Ferreira

Dr. Vagner Ferreira is a Professor at the School of Earth Sciences and Engineering, Hohai University, Nanjing, China. He earned his Ph.D. degree in Geodetic Science from the Federal University of Paraná, Brazil, with a prestigious exchange period at the Karlsruhe Institute of Technology (KIT), Germany. Dr. Ferreira has held appointments as a Visiting Professor at KIT and Griffith University, Australia. His research centers on developing cutting-edge methods to monitor the Earth’s dynamic systems using geodetic remote sensing technologies. His expertise spans satellite geodesy, climate change studies, hydrological modeling, and the integration of multi-sensor observations.

Giuseppe Casula

Dr. Giuseppe Casula currently holds a permanent senior technologist position at the Istituto Nazionale di Geofisica e Vulcanologia (INGV), Unit of Bologna, Italy, since 2003. He studied applied physics with a focus on geophysics at the University of Bologna, where he graduated in 1990 with a thesis on gravimetry. His scientific interests are centered around the use of satellite positioning geodetic techniques such as GNSS and Advanced differential interferometric synthetic aperture radar for the study of crustal deformation in seismic and volcanic areas and areas of hydro-geological instability such as those subject to landslides and subsidence.

remote sensing

MDPI

Editorial

High-Precision and High-Reliability Positioning, Navigation, and Timing: Opportunities and Challenges

Zhetao Zhang [1], Guorui Xiao [2], Zhixi Nie [3], Vagner Ferreira [1,*] and Giuseppe Casula [4]

[1] School of Earth Sciences and Engineering, Hohai University, Nanjing 211100, China; ztzhang@hhu.edu.cn
[2] Institute of Geospatial Information, Information Engineering University, Zhengzhou 450001, China; xgr@whu.edu.cn
[3] College of Oceanography and Space Informatics, China University of Petroleum, Qingdao 266580, China; niezhixi@upc.edu.cn
[4] Istituto Nazionale di Geofisica e Vulcanologia (INGV)—Sezione di Bologna, Viale Berti Pichat 6/2, 40127 Bologna, Italy; giuseppe.casula@ingv.it
* Correspondence: vagnergf@hhu.edu.cn; Tel.: +86-182-6040-0564

Abstract: The research scope of the papers published in this Special Issue mainly focuses on high-precision and high-reliability positioning, navigation, and timing (PNT) with Global Navigation Satellite System (GNSS) or multi-source sensors, resilient PNT with GNSSs or multi-source sensors in challenging environments, integrated PNT with GNSSs and multi-sensor systems, applications of PNT with GNSSs or multi-source sensors, etc.

Keywords: PNT; GNSS; multi-sensor system; challenging environment; applications of GNSSs and multi-sensor systems

Citation: Zhang, Z.; Xiao, G.; Nie, Z.; Ferreira, V.; Casula, G. High-Precision and High-Reliability Positioning, Navigation, and Timing: Opportunities and Challenges. *Remote Sens.* **2024**, *16*, 4403. https://doi.org/10.3390/rs16234403

Received: 2 October 2024
Accepted: 7 November 2024
Published: 25 November 2024

1. Introduction

Global Navigation Satellite Systems (GNSSs) provide high-precision positioning, navigation, and timing (PNT) capabilities in open areas and have gained widespread use in various fields, including high-precision monitoring and intelligent transportation [1]. However, their performance is hindered in challenging environments where signals are susceptible to reflection, refraction, diffraction, and blockage by buildings [2–4]. These factors can degrade signal quality, leading to inconsistent or disrupted PNT with GNSSs. Fortunately, certain sensors complement GNSSs, and thus multi-source sensors, including the inertial measurement unit (IMU), light detection and ranging (LiDAR), and vision and odometer sensors, are extensively explored and employed, particularly in autonomous driving and ground unmanned vehicles [5–8]. Simultaneous Localization and Mapping (SLAM) stands out as a notable application of multi-source sensor fusion, attracting significant attention due to its high robustness and accuracy [9,10]. The diversification of GNSS constellations, multi-source sensors, and observation environments put forward higher requirements for technology and algorithms to maintain high-precision and high-reliability PNT services [11]. Advanced algorithms serve as the key to solving practical application issues related to GNSSs and multi-source sensors, thereby expanding their scope of applications.

This Special Issue aimed to present studies covering improved methods and the latest challenges in PNT, especially in challenging environments, covering a wide range of research investigations and practical applications. Both theoretical and applied research contributions in GNSSs and multi-source sensor fusion technologies in all disciplines were considered. Topics may cover anything from high-precision and high-reliability PNT with GNSS or multi-source sensors, resilient PNT with GNSS or multi-source sensors in challenging environments, integrated PNT with GNSS and multi-sensor systems, and applications of PNT with GNSS or multi-source sensors. Therefore, new algorithms for high-precision positioning and navigation, fusion of multi-sensor systems, software development for data collection, integration, and processing, and their applications in various fields are all included.

2. Overview of Contributions

This section provides a synthesis of the key findings and contributions from each paper published in the Special Issue "High-Precision and High-Reliability Positioning, Navigation, and Timing: Opportunities and Challenges". Here, we examine each contribution to highlight its significance in advancing methodologies, addressing current challenges, and enhancing the overall understanding of PNT in diverse environments.

Dai et al. (contribution 1) proposed a structural health monitoring (SHM) scheme where an inertial measurement unit (IMU) and multi-antenna GNSS were tightly integrated. The phase centers of multiple GNSS antennas were transformed into the IMU center, which increased the observation redundancy and strengthened the positioning model. To evaluate the performance of the tight integration of an IMU and multiple GNSS antennas, high-rate vibrational signals were simulated using a shaking table (i.e., vibrational test platform), and the errors of horizontal displacements of different positioning schemes were analyzed using recordings of a high-precision ranging laser as the reference. The results demonstrated that applying triple-antenna GNSS/IMU integration for measuring the displacements can achieve an accuracy of 2.6 mm, which was about 33.0% and 30.3% superior to the accuracy achieved by the conventional single-antenna GNSS-only and GNSS/IMU solutions, respectively.

Zhang et al. (contribution 2) proposed an improved Carrier Smoothing Code (CSC) algorithm by considering Satellite-induced Code Bias (SICB) for Geostationary Orbit (GEO), Inclined Geosynchronous Orbit (IGSO), and Medium Earth Orbit (MEO) satellites in BeiDou Navigation Satellite System (BDS) constellations. The correction model of SICB for IGSO/MEO satellites was established by using a 0.1-degree interval piecewise weighted least squares Third-Order Curve Fitting Method (TOCFM). The Variational Mode Decomposition combined with Wavelet Transform (VMD-WT) was proposed to establish the correction model of SICB for the GEO satellite. To verify the proposed method, the SICB model was established by collecting 30 Multi-GNSS Experiment (MGEX) BDS stations in different seasons of the year, in which the BDS data of ALIC, KRGG, KOUR, GCGO, GAMG, and SGOC stations were selected for 11 consecutive days to verify the effectiveness of the algorithm. The results showed that there was obvious SICB in the BDS-2 Multipath (MP) combination, but the SICB in the BDS-3 MP was smaller and could be ignored. Compared with the modeling in the references, TOCFM was more suitable for IGSO/MEO SICB modeling, especially for the SICB correction at low elevation angles. After the VMD-WT correction, the root mean square error (RMSE) of SICB of B1I, B2I, and B3I in GEO satellites was reduced by 53.35%, 63.50%, and 64.71%, respectively. Moreover, Zhang et al. [2] carried out Ionosphere-Free Single Point Positioning (IF SPP), Ionosphere-Free CSC SPP (IF CSC SPP), CSC single point positioning with the IGSO/MEO SICB correction based on the TOCFA method (IGSO/MEO SICB CSC), and CSC single point positioning with the IGSO/MEO/GEO SICB correction based on VMD-WT and TOCFA (IGSO/MEO/GEO SICB CSC), respectively. Compared to IF SPP, after the IGSO/MEO/GEO SICB correction, the overall improvement was about 10%, and positioning improved significantly.

Guo et al. (contribution 3) designed and implemented a satellite–ground microwave time–frequency comparison system and method based on a three-frequency mode in response to the requirements for assessing the long-term stability of high-precision space atomic clocks. Ground-based experimental results demonstrated that the equipment layer could achieve a satellite–ground time comparison accuracy better than 0.4 ps (RMS), with the equipment delay stability (ADEV) for all three frequencies being better than 8×10^{-18} at 86,400. The authors constructed a satellite–ground time–frequency comparison simulation and verification platform by using the ground-based experimental results. This platform realized ultra-high-precision satellite–ground time–frequency comparison based on the China Space Station. After correcting various transmission delay errors, the satellite–

ground time comparison achieved an accuracy better than 0.8 ps and an ADEV better than 2×10^{-17} at 86,400 s (i.e., 24 h). This validation of the novel satellite–ground time–frequency comparison system and method, capable of achieving a stability of 10^{-17} order, was not only a significant contribution to the field of space time–frequency systems but also paved the way for future advancements and applications in space science exploration.

Cai et al. (contribution 4) proposed a low-cost and robust multi-sensor data fusion scheme for heterogeneous multi-robot collaborative navigation in indoor environments, which integrated data from IMUs, laser rangefinders and cameras, among others, into heterogeneous multi-robot navigation to address the challenge in the multi-robot collaborative systems. Based on Discrete Kalman Filter (DKF) and Extended Kalman Filter (EKF) principles, a three-step joint filtering model was used to improve the state estimation, and the visual data were processed using the YOLO deep learning target detection algorithm before updating the integrated filter. The proposed integration was tested at multiple levels in an open indoor environment following various formation paths. The results showed that the three-dimensional root mean square error (RMSE) of indoor cooperative localization was 11.3 mm, the maximum error was less than 21.4 mm, and the motion error in occluded environments was suppressed. The proposed fusion scheme was able to satisfy the localization accuracy requirements for efficient and coordinated motion of autonomous mobile robots.

Sun et al. (contribution 5) considered the code multipath to be influenced not only by the elevation and azimuth angle of certain stations to satellites but also to be related to satellite characteristics such as nadir angle. Hence, azimuth angle, elevation angle, nadir angle, and carrier-to-noise power density ratio were taken as multiple indicators to characterize the multipath significantly. Then, the authors proposed an Attention-based Convolutional Long Short-Term Memory (AT-Conv-LSTM) that fully exploited the spatiotemporal correlations of multipaths derived from multiple indicators. The main processing procedures using AT-Conv-LSTM were given. Finally, the AT-Conv-LSTM was applied to a station for 16 consecutive days to verify the multipath mitigation effectiveness. Compared with sidereal filtering, multipath hemispherical map (MHM), and trend-surface analysis-based MHM, the experimental results showed that using AT-Conv-LSTM could decrease the root mean square error and mean absolute error values of the multipath error by more than 60% and 13%, respectively. The proposed method could correct the code multipath to the centimeter level, which was one order of magnitude lower than the uncorrected code multipath. Therefore, the proposed AT-Conv-LSTM network could be used as a powerful alternative tool to realize multipath reduction and will be of broad practical value in the fields of standard and high-precision positioning services.

Yu et al. (contribution 6) presented a novel RGB-D dynamic Simultaneous Localization and Mapping (SLAM) method that improved the precision, stability, and efficiency of localization while relying on lightweight deep learning in a dynamic environment compared to the traditional static feature-based visual SLAM algorithm. Based on ORB-SLAM3, the GCNv2-tiny network instead of the ORB method improved the reliability of feature extraction and matching and the accuracy of position estimation; then, the semantic segmentation thread employed the lightweight YOLOv5s object detection algorithm based on the GSConv network combined with a depth image to determine potentially dynamic regions of the image. Finally, to guarantee that the static feature points are used for position estimation, dynamic probability is employed to determine the true dynamic feature points based on the optical flow, semantic labels, and the state in the last frame. The authors have performed experiments on the TUM datasets to verify the feasibility of the algorithm. Compared with the classical dynamic visual SLAM algorithm, the experimental results demonstrated that the absolute trajectory error was significantly reduced in dynamic environments and that the computing efficiency was improved by 31.54% compared with the real-time dynamic visual SLAM algorithm with close accuracy, demonstrating the superiority of DLD-SLAM in accuracy, stability, and efficiency.

Quan et al. (contribution 7) proposed a robust method based on factor graphs to improve the performance of integrated navigation systems. The authors proposed a detection method based on multi-conditional analysis to determine whether the GNSS was anomalous or not. Moreover, the optimal weight of GNSS measurement was estimated under anomalous conditions to mitigate the impact of GNSS outliers. The proposed method is evaluated through real-world road tests, and the results showed that the positioning accuracy of the proposed method was improved by more than 60% and the missed alarm rate was reduced by 80% compared with the traditional algorithms.

Li et al. (contribution 8) proposed a stationary detection method based on the fast Fourier transform (FFT) for a stopped land vehicle with an idling engine. An urban vehicular navigation experiment was conducted with the authors' GNSS/IMU integration platform. Three stops for 10 to 20 min were set to analyze, generate, and evaluate the FFT-based stationary detection method. The FFT spectra showed clearly idling vibrational peaks during the three stop periods. Through the comparison of FFT spectral features with decelerating and accelerating periods, the amplitudes of vibrational peaks were put forward as the key factors of stationary detection. For consecutive stationary detection in the GNSS/IMU integration process, a three-second sliding window with a one-second updating rate of the FFT was applied to check the amplitudes of the peaks. For the assessment of the proposed stationary detection method, GNSS observations were removed to simulate outages during three stop periods, and the proposed detection method was conducted together with the ZVU. The results showed that the proposed method achieved a 99.7% correct detection rate, and the divergence of the positioning error constrained via the ZVU was within 2 cm for the experimental stop periods, which indicated the effectiveness of the proposed method.

Wang et al. (contribution 9) investigated the feasibility of extracting ionospheric observables from the multi-GNSS single-frequency (SF) UU-PPP to reduce the cost of ionospheric modeling. Meanwhile, the between-satellite single-differenced (SD) method was applied to remove the effects of the receiver differential code bias (DCB) with short-term time-varying characteristics in regional ionospheric modeling. With the introduction of the proposed SD ionospheric model into the multi-GNSS kinematic RT SF-PPP, the initialization speed of vertical positioning errors can be improved by 21.3% in comparison with the GRAPHIC (GRoup And PHase Ionospheric Correction) SF-PPP model. After reinitialization, both horizontal and vertical positioning errors of the SD ionospheric-constrained (IC) SF-PPP can be maintained within 0.2 m. This proved that the proposed SDIC SF-PPP model could enhance the continuity and stability of kinematic positioning in the case of some GNSS signals missing or blocked. Compared with the GRAPHIC SF-PPP, the horizontal positioning accuracy of the SDIC SF-PPP in kinematic mode can be improved by 37.9%, but its vertical positioning accuracy may be decreased. Overall, the 3D positioning accuracy of the SD ionospheric-constrained RT SF-PPP can be better than 0.3 m.

Pang et al. (contribution 10) introduced a cost-effective Simultaneous Localization and Mapping (SLAM) system design that maintains high performance while significantly reducing costs. First, the authors developed a robust robotic platform based on a traditional four-wheeled vehicle structure, enhancing flexibility and load capacity. Then, they adapted the SLAM algorithm using the LiDAR-inertial odometry framework coupled with the Fast Iterative Closest Point (ICP) algorithm to balance accuracy and real-time performance. Finally, they integrated the 3D multi-goal Rapidly exploring Random Tree (RRT) algorithm with Nonlinear Model Predictive Control (NMPC) for autonomous exploration in complex environments. Comprehensive experimental results confirmed the system's capability for real-time, autonomous navigation and mapping in intricate indoor settings, rivaling more expensive SLAM systems in accuracy and efficiency at a lower cost.

Zhang et al. (contribution 11) presented a Sage–Husa Kalman filter where the noise uncertainty of strong motion acceleration was adaptively estimated to integrate GNSSs and strong motion acceleration for obtaining the displacement series. The performance of the proposed method was validated by a shake table simulation experiment and the

GNSS/strong motion co-located stations collected during the 2023 Mw 7.8 and Mw 7.6 earthquake doublet in southeast Turkey. The experimental results showed that the proposed method enhanced adaptability to the variation in strong motion accelerometer noise levels and improved the precision of the integrated displacement series. The displacement derived from the proposed method was up to 28% more accurate than that from the Kalman filter in the shake table test, and the correlation coefficient with respect to the references arrived at 0.99. The application to the earthquake event showed that the proposed method can capture seismic waveforms at a promotion of 46% and 23% in the horizontal and vertical directions, respectively, compared with the results of the Kalman filter.

Tian et al. (contribution 12) conducted a systematic and comprehensive evaluation of signal characteristics for BDS-3, BDS-2, GPS, and Galileo regarding carrier-to-noise ratio (C/N0), code noise, and multipath in the contribution using the data of the globally distributed MGEX stations. First, a comprehensive signal quality assessment method for BDS/Galileo/GPS satellites and signals was proposed, including C/N0 modeling and MP modeling. For BDS, the BDS-3 satellites apparently have higher signal power than the BDS-2 satellites at the same frequency, such as B1I and B3I, and the B2a signal of BDS-3 was superior to other signals with a signal power that was comparable with the superior Galileo E5 and GPS L5 signals. Among all the signals, the observation accuracy of E5 was the highest regardless of receiver type, and the next highest was BDS-3 B2a and GPS L5. Due to not being affected by the systematic code errors of BDS-2, the observations of BDS-3 satellites contained smaller multipath errors than those of BDS-2 satellites. As for the multipath suppression performance, the BDS-3 B2a signal, GPS L5, and Galileo E5 and E5b performed better than the other signals, which may be related to their wide signal bandwidths.

Mu et al. (contribution 13) proposed a cruise speed model based on the self-attention mechanism for speed estimation in Autonomous Underwater Vehicle (AUV) navigation systems. By utilizing variables such as acceleration, angle, angular velocity, and propeller speed as inputs, the self-attention mechanism was constructed using Long Short-Term Memory (LSTM) for handling the above information, enhancing the model's accuracy during persistent bottom-track velocity failures. Additionally, this study introduced the water-track velocity information to enhance the generalization capability of the network and improved its speed estimation accuracy. The sea trial experiment results indicated that compared to traditional methods, this model demonstrated higher accuracy and reliability with both position error and velocity error analysis when the used Pathfinder DVL fails, providing an effective solution for AUV combined navigation systems.

Gong et al. (contribution 14) proposed a new unified positioning algorithm using multi-sensor time difference in arrival (TDOA) and frequency difference in arrival (FDOA) measurements without prior target source information. The method represented the position of the target source using MPR and described the localization problem as a weighted least squares (WLS) problem with two constraints. The authors first obtained the initial estimates by WLS without considering the constraints and then investigated a two-step error correction method based on the constraints. The first step corrected the initial estimate using the Taylor series expansion technique, and the second step corrected the DOA estimate in the previous step using the direct error compensation technique based on the properties of the second constraint. Simulation experiments showed that the method was effective for the unified positioning of moving targets and could achieve the Cramer–Rao lower bound (CRLB).

Liu et al. (contribution 15) proposed a new method to locate UAVs via shape and spatial relationship matching (SSRM) of buildings in urban scenes as an alternative to UAV localization via image matching to address the challenge of capturing the unique characteristics of buildings due to their high density and similarity in shape within urban environments. SSRM first extracted individual buildings from UAV images using the SOLOv2 instance segmentation algorithm. Then, these individual buildings were subsequently matched with vector e-map data (stored in .shp format) based on their shape and spatial relationship to determine their actual latitude and longitude. Control points were generated according to the matched buildings, and finally, the UAV position was determined. SSRM can efficiently realize high-precision UAV localization in urban scenes. Under the verification of actual data, SSRM achieved localization errors of 7.38 m and 11.92 m in downtown and suburban areas, respectively, with better localization performance than the radiation-variation insensitive feature transform (RIFT), channel features of the oriented gradient (CFOG), and SSM algorithms. Moreover, the SSRM algorithm exhibited a smaller localization error in areas with higher building density.

Li et al. (contribution 16) developed a dual-media stereovision measurement simulation model and conducted comprehensive simulation experiments to analyze the impact of refraction-parameter deviations on measurements in underwater structure visual navigation. The results indicated that to achieve high-precision underwater measurement outcomes, the calibration method for refraction parameters, the distribution of the targets in the field of view, and the distance of the target from the camera must all be meticulously designed. These findings provided guidance for the construction of underwater stereo-vision measurement systems, the calibration of refraction parameters, underwater experiments, and practical applications.

3. Conclusions and Future Directions

The contributions to this Special Issue highlight significant advancements in high-precision and high-reliability PNT technologies. These advancements pave the way for more reliable, accurate, and versatile PNT solutions across various applications, from structural health monitoring to autonomous navigation systems. Key takeaways include:

(1) Integration of multiple sensors and data fusion techniques greatly enhances PNT accuracy and reliability, particularly in challenging environments.
(2) Machine learning and deep learning approaches show promise in improving GNSS signal processing, multipath mitigation, and SLAM algorithms.
(3) Innovative algorithms for error correction and adaptive filtering demonstrate substantial improvements in positioning accuracy across various applications.
(4) The BeiDou Navigation Satellite System (BDS) continues to evolve, with BDS-3 showing improved performance over BDS-2 in many aspects.

As the field continues to evolve, interdisciplinary approaches combining advances in sensor technology, signal processing, artificial intelligence (AI), and application-specific knowledge will likely play a crucial role in driving innovation in PNT systems. Therefore, based on the findings published in this Special Issue, future research directions in PNT may include:

(1) Further development of robust multi-sensor fusion algorithms for seamless indoor–outdoor navigation.
(2) Exploration of advanced AI techniques for real-time signal processing and error mitigation in GNSSs and other positioning systems.
(3) Investigation of novel approaches for PNT in emerging fields such as autonomous vehicles, UAVs, and underwater navigation.
(4) Continued improvement of GNSS performance, particularly in challenging urban and indoor environments.

These future directions aim to address the ongoing challenges in PNT technologies while pushing the boundaries of accuracy, reliability, and applicability across various domains.

Author Contributions: Conceptualization, Z.Z. and V.F.; writing—original draft preparation, Z.Z., G.X., Z.N., V.F. and G.C.; writing—review and editing, Z.Z., G.X., Z.N., V.F. and G.C. All authors have read and agreed to the published version of the manuscript.

Funding: Research Project on the Education and Teaching Reform of "Industry Education Integration" in Surveying and Mapping Geographic Information in Jiangsu Province in 2024.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

List of Contributions:

1. Dai, W.; Li, X.; Yu, W.; Qu, X.; Ding, X. Multi-Antenna Global Navigation Satellite System/Inertial Measurement Unit Tight Integration for Measuring Displacement and Vibration in Structural Health Monitoring. *Remote Sens.* **2024**, *16*, 1072. https://doi.org/10.3390/rs16061072.
2. Zhang, Q.; Ma, X.; Gao, Y.; Huang, G.; Zhao, Q. An Improved Carrier-Smoothing Code A-lgorithm for BDS Satellites with SICB. *Remote Sens.* **2023**, *15*, 5253. https://doi.org/10.3390/rs15215253.
3. Guo, Y.; Gao, S.; Pan, Z.; Wang, P.; Gong, X.; Chen, J.; Song, K.; Zhong, Z.; Yue, Y.; Guo, L.; et al. Advancing Ultra-High Precision in Satellite–Ground Time–Frequency Comparison: Ground-Based Experiment and Simulation Verification for the China Space Station. *Remote Sens.* **2023**, *15*, 5393. https://doi.org/10.3390/rs15225393.
4. Cai, Z.; Liu, J.; Chi, W.; Zhang, B. A Low-Cost and Robust Multi-Sensor Data Fusion Scheme for Heterogeneous Multi-Robot Cooperative Positioning in Indoor Environments. *Remote Sens.* **2023**, *15*, 5584. https://doi.org/10.3390/rs15235584.
5. Sun, J.; Tang, Z.; Zhou, C.; Wei, J. Characterization of BDS Multipath Effect Based on AT-Conv-LSTM Network. *Remote Sens.* **2024**, *16*, 73. https://doi.org/10.3390/rs16010073.
6. Yu, H.; Wang, Q.; Yan, C.; Feng, Y.; Sun, Y.; Li, L. DLD-SLAM: RGB-D Visual Simultaneous Localisation and Mapping in Indoor Dynamic Environments Based on Deep Learning. *Remote Sens.* **2024**, *16*, 246. https://doi.org/10.3390/rs16020246.
7. Quan, S.; Chen, S.; Zhou, Y.; Zhao, S.; Hu, H.; Zhu, Q. A Robust Position Estimation Method in the Integrated Navigation System via Factor Graph. *Remote Sens.* **2024**, *16*, 562. https://doi.org/10.3390/rs16030562.
8. Li, M.; Nie, W.; Suvorkin, V.; Rovira-Garcia, A.; Zhang, W.; Xu, T.; Xu, G. Stationary Detection for Zero Velocity Update of IMU Based on the Vibrational FFT Feature of Land Vehicle. *Remote Sens.* **2024**, *16*, 902. https://doi.org/10.3390/rs16050902.
9. Wang, A.; Zhang, Y.; Chen, J.; Liu, X.; Wang, H. Regional Real-Time Between-Satellite Single-Differenced Ionospheric Model Establishing by Multi-GNSS Single-Frequency Observations: Performance Evaluation and PPP Augmentation. *Remote Sens.* **2024**, *16*, 1511. https://doi.org/10.3390/rs16091511.
10. Pang, C.; Zhou, L.; Huang, X. A Low-Cost 3D SLAM System Integration of Autonomous Exploration Based on Fast-ICP Enhanced LiDAR-Inertial Odometry. *Remote Sens.* **2024**, *16*, 1979. https://doi.org/10.3390/rs16111979.
11. Zhang, Y.; Nie, Z.; Wang, Z.; Zhang, G.; Shan, X. Integration of High-Rate GNSS and Strong Motion Record Based on Sage–Husa Kalman Filter with Adaptive Estimation of Strong Motion Acceleration Noise Uncertainty. *Remote Sens.* **2024**, *16*, 2000. https://doi.org/10.3390/rs16112000.
12. Tian, Y.; Xiao, G.; Guo, R.; Zhao, D.; Zhang, L.; Xin, J.; Guo, J.; Han, Y.; Du, X.; He, D.; et al. A Comprehensive Signal Quality Assessment for BDS/Galileo/GPS Satellites and Signals. *Remote Sens.* **2024**, *16*, 2277. https://doi.org/10.3390/rs16132277.
13. Mu, X.; Yi, Y.; Zhu, Z.; Zhu, L.; Wang, Z.; Qin, H. Cruise Speed Model Based on Self-Attention Mechanism for Autonomous Underwater Vehicle Navigation. *Remote Sens.* **2024**, *16*, 2580. https://doi.org/10.3390/rs16142580.
14. Gong, W.; Song, X.; Zhu, C.; Wang, Q.; Li, Y. Closed-Form Method for Unified Far-Field and Near-Field Localization Based on TDOA and FDOA Measurements. *Remote Sens.* **2024**, *16*, 3047. https://doi.org/10.3390/rs16163047.
15. Liu, Y.; Bai, J.; Sun, F. Visual Localization Method for Unmanned Aerial Vehicles in Urban Scenes Based on Shape and Spatial Relationship Matching of Buildings. *Remote Sens.* **2024**, *16*, 3065. https://doi.org/10.3390/rs16163065.
16. Li, G.; Huang, S.; Yin, Z.; Zheng, N.; Zhang, K. Analysis of the Influence of Refraction-Parameter Deviation on Underwater Stereo-Vision Measurement with Flat Refraction Interface. *Remote Sens.* **2024**, *16*, 3286. https://doi.org/10.3390/rs16173286.

References

1. Yang, Y.; Mao, Y.; Sun, B. Basic Performance and Future Developments of BeiDou Global Navigation Satellite System. *Satell. Navig.* **2020**, *1*, 1. [CrossRef]
2. Su, M.; Zheng, J.; Yang, Y.; Wu, Q. A new multipath mitigation method based on adaptive thresholding wavelet denoising and double reference shift strategy. *GPS Solut.* **2018**, *22*, 40. [CrossRef]
3. Zheng, K.; Tan, L.; Liu, K.; Li, P.; Chen, M.; Zeng, X. Multipath mitigation for improving GPS narrow-lane uncalibrated phase delay estimation and speeding up PPP ambiguity resolution. *Measurement* **2023**, *206*, 112243. [CrossRef]
4. Hsu, L.-T. Analysis and Modeling GPS NLOS Effect in Highly Urbanized Area. *GPS Solut.* **2018**, *22*, 7. [CrossRef]
5. Zhang, B.; Hou, P.; Zha, J.; Liu, T. PPP-RTK functional models formulated with undifferenced and uncombined GNSS observations. *Satell Navig.* **2022**, *3*, 3. [CrossRef]
6. Gu, S.; Dai, C.; Fang, W.; Zheng, F.; Wang, Y.; Zhang, Q.; Lou, Y.; Niu, X. Multi-GNSS PPP/INS Tightly Coupled Integration with Atmospheric Augmentation and Its Application in Urban Vehicle Navigation. *J. Geod.* **2021**, *95*, 64. [CrossRef]
7. Zhu, F.; Chen, W.; Guo, F.; Zhang, X. Combining Context Connectivity and Behavior Association to Develop an Indoor/Outdoor Context Detection Model with Smartphone Multisensor Fusion. *IEEE Internet Things J.* **2023**, *11*, 2883–2898. [CrossRef]
8. Li, X.; Li, X.; Li, S.; Zhou, Y.; Sun, M.; Xu, Q.; Xu, Z. Centimeter-Accurate Vehicle Navigation in Urban Environments with a Tightly Integrated PPP-RTK/MEMS/Vision System. *GPS Solut.* **2022**, *26*, 124. [CrossRef]
9. Chang, L.; Niu, X.; Liu, T.; Tang, J.; Qian, C. GNSS/INS/LiDAR-SLAM Integrated Navigation System Based on Graph Optimization. *Remote Sens.* **2019**, *11*, 1009. [CrossRef]
10. Liu, T.; Li, B.; Yang, L.; Qiao, J.; Chen, W. Tightly Coupled Integration of GNSS/UWB/VIO for Reliable and Seamless Positioning. *IEEE Trans. Intell. Transp. Syst.* **2023**, *25*, 2116–2128. [CrossRef]
11. Zeng, P.; Zhang, Z.; Wen, Y.; He, X.; He, L.; Li, M.; Chen, W. Properties of Multi-GNSS Uncalibrated Phase Delays with Considering Satellite Systems, Receiver Types, and Network Scales. *Satell. Navig.* **2023**, *4*, 19. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

Article

Analysis of the Influence of Refraction-Parameter Deviation on Underwater Stereo-Vision Measurement with Flat Refraction Interface

Guanqing Li [1,*], Shengxiang Huang [2], Zhi Yin [1,3], Nanshan Zheng [1] and Kefei Zhang [1]

[1] School of Environment and Spatial Informatics, China University of Mining and Technology, Xuzhou 221116, China; zyin@jou.edu.cn (Z.Y.); znshcumt@cumt.edu.cn (N.Z.); profkzhang@cumt.edu.cn (K.Z.)
[2] School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China; sxhuang@whu.edu.cn
[3] School of Marine Technology and Geomatics, Jiangsu Ocean University, Lianyungang 222005, China
* Correspondence: guanqing.li@cumt.edu.cn

Abstract: There has been substantial research on multi-medium visual measurement in fields such as underwater three-dimensional reconstruction and underwater structure monitoring. Addressing the issue where traditional air-based visual-measurement models fail due to refraction when light passes through different media, numerous studies have established refraction-imaging models based on the actual geometry of light refraction to compensate for the effects of refraction on cross-media imaging. However, the calibration of refraction parameters inevitably contains errors, leading to deviations in these parameters. To analyze the impact of refraction-parameter deviations on measurements in underwater structure visual navigation, this paper develops a dual-media stereo-vision measurement simulation model and conducts comprehensive simulation experiments. The results indicate that to achieve high-precision underwater-measurement outcomes, the calibration method for refraction parameters, the distribution of the targets in the field of view, and the distance of the target from the camera must all be meticulously designed. These findings provide guidance for the construction of underwater stereo-vision measurement systems, the calibration of refraction parameters, underwater experiments, and practical applications.

Keywords: multi-medium visual measurement; stereo vision; refraction-parameter deviation; underwater navigation

Citation: Li, G.; Huang, S.; Yin, Z.; Zheng, N.; Zhang, K. Analysis of the Influence of Refraction-Parameter Deviation on Underwater Stereo-Vision Measurement with Flat Refraction Interface. *Remote Sens.* **2024**, *16*, 3286. https://doi.org/10.3390/rs16173286

Academic Editors: Zhetao Zhang, Guorui Xiao, Zhixi Nie, Vagner Ferreira, Giuseppe Casula and Pablo Rodríguez-González

Received: 14 June 2024
Revised: 1 September 2024
Accepted: 2 September 2024
Published: 4 September 2024

1. Introduction

Multi-medium visual measurement is a crucial branch of the visual-measurement field. Common multi-media visual-measurement scenarios include situations where light traverses two mediums, such as air–water, or three mediums, such as air–glass–air and air–glass–water [1–3]. Examples of such scenarios include the visual measurement of thermal deformation of alumina ceramic plates and stainless-steel plates under radiant heating [4] as well as the three-dimensional shape measurement of underwater bridge piers [5].

Multi-medium visual measurement has garnered extensive attention in areas such as underwater deformation monitoring and three-dimensional reconstruction [1,5–8]. In addition, it can also be applied to underwater navigation and positioning. In underwater engineering, such as immersed tunnels, the precise alignment of prefabricated elements requires high-precision underwater navigation and positioning [9]. Similarly, the autonomous recovery of autonomous underwater vehicles (AUVs) necessitates accurate navigation [10]. For navigation scenarios involving the docking of underwater structures at close range, visual measurement becomes a crucial method due to its rich information and high accuracy [11,12].

In underwater visual measurement, cameras are either placed in waterproof housings or positioned above the water surface. Consequently, a significant characteristic is that light propagation undergoes refraction at the interfaces of different media, causing the light rays to deviate from a straight-line path. Therefore, traditional air-based stereo-vision measurement models are no longer applicable under multi-media conditions [13]. When the camera images an object through a sealed housing's transparent window, the interface typically comes in shapes such as planar, hemispherical, and cylindrical [14–16]. Hemispherical ports counteract the refraction effect through their specific shapes, do not reduce the field of view, and can withstand high pressure in deep water, but they may induce image blurring and have higher manufacturing requirements [17,18]. Cylindrical interfaces have the advantage of allowing a larger field of view (in one direction) and are relatively simple and cost-effective to manufacture [19]. Although flat interfaces can significantly reduce the field of view and may introduce chromatic aberration, they are well-studied and have lower manufacturing costs [17]. Therefore, the focus of this paper is on underwater visual measurement with a flat interface.

The basic problems of underwater photogrammetry with a flat interface, including the basic formulas and the methods of automatic reduction to one perspective, were explored decades ago [20–22]. Since no later than the 1960s, researchers have proposed numerous methods for refractive compensation in planar underwater visual measurement. Broadly, these methods can be categorized into three solutions. The first approach involves placing two auxiliary calibration grids or a calibration frame within the object space, utilizing the calibration structure to determine the direction of light prior to its incident [23]. However, this method necessitates the customization of specific calibration grids or frames, making the implementation process relatively complex. The second approach entails focal length compensation or refraction absorption, wherein the pixel offset error induced by refraction is approximated as an error stemming from changes in lens focal length or distortion. By calibrating the camera's lens or distortion parameters, the influence of refraction can be mitigated [24–26]. Nevertheless, due to the nonlinearity of the refractive effect and the dependence of the refraction error's magnitude on the object's position and the angle of incidence, these methods only approximate the elimination of refraction effects [13,27]. The final method is geometric refraction correction, which involves establishing an underwater refraction imaging model through geometric analysis to explicitly account for the refractive effects, thereby theoretically ensuring the accuracy of the measurement results [8,28–37].

Research on geometric refraction correction primarily focuses on the development of calibration methods and measurement models. The two-stage underwater camera calibration represents a groundbreaking advancement in the field [29]. The optimization method can also be employed to calibrate parameters such as the normal vector of the refraction interface and the distance from the interface to the camera. However, it is essential to assign appropriate initial values in advance [30]. Leveraging the geometric property that two incident light rays from the same object entering the stereo camera lie on the same plane, the calibration parameters can also be optimized through 3D point remapping [33]. The refractive index varies with different light frequencies. The parameters of the underwater camera can be calibrated by calculating the offset in the imaging position of different light frequencies emitted by a specially designed calibration plate placed at the same location [31]. To eliminate measurement errors caused by the spherical refraction interface, an underwater calibration algorithm based on an advanced non-dominated sorting genetic algorithm is proposed. This approach, utilizing an integrated geometric refraction model, significantly enhances the performance of underwater visual measurement [35].

Regarding the influence of refraction on visual measurements, for multi-view (more than two views) underwater 3D reconstruction, the influence on the accuracy of the 3D reconstruction is evaluated quantitatively and systematically in [25]. Tong [38] used simulation and real experiments to analyze the influence of different refraction parameters and proposed measures to reduce the influence. However, due to the inevitability of measurement errors, the calibrated refraction parameters must contain deviations. When using the refraction-measurement model for refraction compensation, the impact of the parameter deviation on the visual measurement results is also worth studying. In addition, conducting underwater experiments is relatively challenging, so performing relevant analyses through simulation experiments is a preferable option. To the best of our knowledge, current research does not provide specific algorithms for simulation analysis. To investigate the impact of refraction-parameter deviations on stereo-visual measurement in dual-media conditions, this paper first established a simulation model for stereo-visual measurement in the air–water scenario. Then, a thorough analysis of the relationship between the stereo-visual measurement model and refraction-parameter deviations using simulation experiments was conducted. The conclusions can provide guidance for the construction of underwater stereo-vision measurement systems, refraction parameter calibration, underwater experiments, and practical applications.

The subsequent structure of the paper is arranged as follows. Section 2 introduces the light refraction geometry and measurement model for multi-media stereo vision, establishes a simulation algorithm for dual-media stereo-visual measurement and provides a simulation experimental design. The experimental results are presented and discussed in Sections 3 and 4, respectively. Section 5 presents the conclusions.

2. Methods

2.1. Measurement Model

The stereo-vision measurement system includes left and right cameras. It is assumed that the cameras image the target through a transparent window inside a watertight cabin. The material of the transparent window is generally glass. The light reflected or emitted by the underwater target reaches the camera through three mediums and two refraction interfaces. Both interfaces are assumed to be planar and parallel to each other. The refraction geometry is shown in Figure 1.

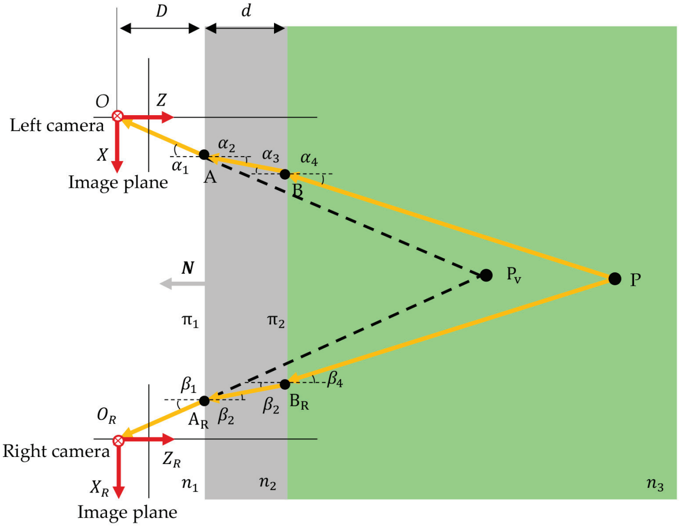


Figure 1. Refractive geometry for air–glass–water medium with two flat interfaces.

The camera coordinate system of the left camera $O^{-}XYZ$ is used as the reference coordinate system. P is the target point underwater and its coordinate is (X, Y, Z). The two refraction interfaces are denoted as π_1 and π_2. The distance from the left camera optical center to the interface π_1 is D, and the thickness of the glass layer is d. Due to refraction, the optical paths between P and the cameras are not straight lines, but the broken lines PB^{-}BA^{-}AO and PB$_R$$^{-}B_RA_R$$^{-}A_RO_R$ according to Snell's law. The refraction angles of the two rays are denoted by α_i and β_i. It is assumed that both cameras have been carefully calibrated. The camera coordinate system of the right camera is O_R-$X_R Y_R Z_R$, and the coordinates of the origin O_R in the reference coordinate system are denoted as $\boldsymbol{t} = \left(t_x,\ t_y,\ t_z\right)$, which is the translation vector of the right camera with respect to the left camera.

The target point P is the intersection point of the lines BP and $\mathrm{B_R}$P. If the equations of two lines are found, the intersection point is the coordinate of P. The derivation of the equations for the lines BP and $\mathrm{B_R}$P is detailed below [1].

After obtaining two images from the left and right cameras and completing target recognition and matching, the point $\mathrm{P_v}$ is obtained using conventional triangulation. The points O, A, and $\mathrm{P_v}$ are collinear. The points $\mathrm{O_R}$, $\mathrm{A_R}$, and $\mathrm{P_v}$ are also collinear. The coordinates of the points O, $\mathrm{O_R}$, and $\mathrm{P_v}$ are known. Therefore, the unit direction vectors of the lines OA and $\mathrm{O_R A_R}$ can be obtained as

$$\begin{cases} \boldsymbol{\zeta}^1 = \left(\zeta_x^1,\ \zeta_y^1,\ \zeta_z^1\right) = \frac{\boldsymbol{P}_{\mathrm{v}} - \boldsymbol{O}}{\|\boldsymbol{P}_{\mathrm{v}} - \boldsymbol{O}\|} \\ \boldsymbol{\xi}^1 = \left(\xi_x^1,\ \xi_y^1,\ \xi_z^1\right) = \frac{\boldsymbol{P}_{\mathrm{v}} - \boldsymbol{O}_{\mathrm{R}}}{\|\boldsymbol{P}_{\mathrm{v}} - \boldsymbol{O}_{\mathrm{R}}\|} \end{cases} \tag{1}$$

In terms of the direction vector of the line and a point on the line, the equations for the lines OA and $\mathrm{O_R A_R}$ are

$$\begin{cases} \frac{X}{\zeta_x^1} = \frac{Y}{\zeta_y^1} = \frac{Z}{\zeta_z^1} \\ \frac{X - t_x}{\xi_x^1} = \frac{Y - t_y}{\xi_y^1} = \frac{Z - t_z}{\xi_z^1} \end{cases} \tag{2}$$

The points A and $\mathrm{A_R}$ are the intersections of the lines OA and $\mathrm{O_R A_R}$ with the interface π_1. Assuming that the normal vector $\boldsymbol{N} = \left(N_x, N_y,\ N_z\right)$ of π_1 is known, the equation for π_1 can be expressed as

$$\boldsymbol{N}\boldsymbol{X}^T + D = 0 \tag{3}$$

Combining Equations (2) and (3), the coordinates of the points A and $\mathrm{A_R}$ can be obtained as

$$\begin{cases} \boldsymbol{A} = \left(A_x, A_y, A_z\right) = -\frac{D}{\boldsymbol{N}\left(\boldsymbol{\zeta}^1\right)^{\mathrm{T}}}\boldsymbol{\zeta}^1 \\ \boldsymbol{A}_R = -\frac{\boldsymbol{N}\boldsymbol{t}^T + D}{\boldsymbol{N}\left(\boldsymbol{\xi}^1\right)^T}\boldsymbol{\xi}^1 + \boldsymbol{t} \end{cases} \tag{4}$$

According to the vector inner product formula, the angles α_1 and β_1 are

$$\begin{cases} \alpha_1 = \frac{\boldsymbol{N} \cdot \boldsymbol{\zeta}^1}{\|\boldsymbol{N}\| \left\|\boldsymbol{\zeta}^1\right\|} \\ \beta_1 = \frac{\boldsymbol{N} \cdot \boldsymbol{\xi}^1}{\|\boldsymbol{N}\| \left\|\boldsymbol{\xi}^1\right\|} \end{cases} \tag{5}$$

Based on Snell's law, the angles α_2 and β_2 are

$$\begin{cases} \alpha_2 = \sin^{-1}\left(\frac{n_1 \cdot \sin\ \alpha_1}{n_2}\right) \\ \beta_2 = \sin^{-1}\left(\frac{n_1 \cdot \sin\ \beta_1}{n_2}\right) \end{cases} \tag{6}$$

For the left camera, the lines OA, AB, and the camera optical axis are coplanar, so the normal vector of this plane is

$$\begin{cases} \boldsymbol{n}_l = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ \boldsymbol{k} = \boldsymbol{\zeta}^1 \times \boldsymbol{n}_l = \begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix} \end{cases} \tag{7}$$

where $\boldsymbol{n}_l$ is the unit direction vector of the left camera optical axis. Through Rodriguez's formula, the rotation matrix $\boldsymbol{R}_{\mathrm{A}}$ and the unit direction vector $\boldsymbol{\zeta}^2$ of the line AB are determined as

$$\begin{cases} \boldsymbol{R}_{\mathrm{A}} = \boldsymbol{I} + [\boldsymbol{k}]_{\times} \sin\ \alpha_2 + [\boldsymbol{k}]_{\times}^2 (1 - \cos\ \alpha_2) \\ [\boldsymbol{k}]_{\times} = \begin{bmatrix} 0 & -k_3 & k_2 \\ k_3 & 0 & -k_1 \\ -k_2 & k_1 & 0 \end{bmatrix} \\ \boldsymbol{\zeta}^2 = \left(\zeta_x^2,\ \zeta_y^2,\ \zeta_z^2\right) = \frac{(\boldsymbol{R}_{\mathrm{A}}\boldsymbol{n}_1)^T}{\left\|(\boldsymbol{R}_{\mathrm{A}}\boldsymbol{n}_1)^T\right\|} \end{cases} \tag{8}$$

where $\boldsymbol{I}$ is the unit matrix and $[\boldsymbol{k}]_{\times}$ is the antisymmetric matrix generated by $\boldsymbol{k}$.

Considering the thickness d of the glass layer, we can obtain the coordinates of point B

$$\boldsymbol{B} = \left(B_x, B_y, B_z\right) = \boldsymbol{A} + \frac{d}{\cos\ \alpha_2}\boldsymbol{\zeta}^2 \tag{9}$$

Because $\alpha_2 = \alpha_3$, similar to Equations (6) and (8), the angle α_4 and the unit direction vector of line BP can be calculated as

$$\begin{cases} \alpha_4 = \sin^{-1}\left(\frac{n_2 \cdot \sin \alpha_2}{n_3}\right) \\ \boldsymbol{R}_{\mathrm{B}} = \boldsymbol{I} + [\boldsymbol{k}]_{\times} \sin\ \alpha_4 + [\boldsymbol{k}]_{\times}^2 (1 - \cos\ \alpha_4) \\ \boldsymbol{\zeta}^3 = \left(\zeta_x^3,\ \zeta_y^3,\ \zeta_z^3\right) = \frac{(\boldsymbol{R}_{\mathrm{B}}\boldsymbol{n}_1)^T}{\left\|(\boldsymbol{R}_{\mathrm{B}}\boldsymbol{n}_1)^T\right\|} \end{cases} \tag{10}$$

For the line BP, the direction vector $\boldsymbol{\zeta}^3$ and a point on the line, B, are known; therefore, its equation is

$$\frac{X - B_x}{\zeta_x^3} = \frac{Y - B_y}{\zeta_y^3} = \frac{Z - B_z}{\zeta_z^3} \tag{11}$$

For the right camera, the equation of the line $\mathrm{B_R}$P can be obtained following a similar derivation process

$$\frac{X - B_{Rx}}{\xi_x^3} = \frac{Y - B_{Ry}}{\xi_y^3} = \frac{Z - B_{Rz}}{\xi_z^3} \tag{12}$$

In theory, the intersection of the lines BP and $\mathrm{B_R}$P is point P, which is to be determined. However, the lines BP and $\mathrm{B_R}$P may not intersect due to the errors. Suppose there exists a point H on the line BP and a point M on the line $\mathrm{B_R}$P and the line from H to M forms the vector $\boldsymbol{Q}$

$$\begin{cases} \boldsymbol{H} = \boldsymbol{B} + s_1\boldsymbol{\zeta}^3 \\ \boldsymbol{M} = \boldsymbol{B}_{\mathrm{R}} + s_2\boldsymbol{\xi}^3 \\ \boldsymbol{Q} = \boldsymbol{H} - \boldsymbol{M} \end{cases} \tag{13}$$

where s_1 and s_2 are the coefficients. Our aim is to minimize the length of $\boldsymbol{Q}$. The length of the common perpendicular segment of two lines is the smallest. If $\boldsymbol{Q}$ is perpendicular to both lines BP and $\mathrm{B_R}$P, we have

$$\begin{cases} \boldsymbol{Q} \cdot \boldsymbol{\zeta}^3 = 0 \\ \boldsymbol{Q} \cdot \boldsymbol{\xi}^3 = 0 \end{cases} \tag{14}$$

According to Equation (14), s_1 and s_2 can be calculated, and then $\boldsymbol{H}$ and $\boldsymbol{M}$ can be obtained. The average of $\boldsymbol{H}$ and $\boldsymbol{M}$ is considered to be point P

$$\boldsymbol{P} = \frac{\boldsymbol{H} + \boldsymbol{M}}{2} \tag{15}$$

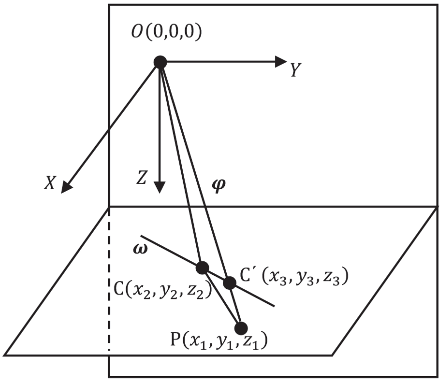
The model is derived for a three-medium scenario, but with minor adjustments, it is equally applicable to a two-medium scenario.

2.2. Dual-Medium Simulation Model

The refraction parameters involved in the above measurement model include the distance from the camera to the refraction interface, the normal direction of the refraction interface, and the refractive indices of different mediums. Calibration errors of these parameters may impact the measurement results. Therefore, it is necessary to assess the impact of refractive parameter deviations on measurement results. However, conducting such evaluations through underwater experiments is relatively complex. In contrast, simulation experiments are more efficient. Consequently, this paper investigates a simulation model for underwater stereo-visual measurement. For simplicity, the camera is placed in the air and the target in the water, constituting an air–water scenario.

Taking the left camera coordinate system as the reference coordinate system. Given the coordinate of the target P is (x_1, y_1, z_1); the distance from the left camera to the refractive interface is D, and the normal vector of the refractive interface is $\boldsymbol{N}$. The camera is in the air. The target is in the water. The refractive indices of air and water are n_1 and n_2, respectively.

For the left camera, the refraction geometry is shown in Figure 2. O represents the camera's optical center. P is the target. C is the refraction point. C$'$ is the intersection of the line OP with the refractive interface. The objective is to calculate the coordinate of point C.



$O(0,0,0)$
Y
X
Z
$\boldsymbol{\varphi}$
$\boldsymbol{\omega}$
$\mathrm{C}'\ (x_3, y_3, z_3)$
$\mathrm{C}(x_2, y_2, z_2)$
$\mathrm{P}(x_1, y_1, z_1)$

Figure 2. Refracted light in the dual-medium scenario.

The direction vector $\boldsymbol{\varphi}$ of line OP is

$$\boldsymbol{\varphi} = \boldsymbol{P} - \boldsymbol{O} \tag{16}$$

The entire refracted light PC-CO is in a refraction plane. Since PC and CO are coplanar with the normal vector $\boldsymbol{N}$ of the refraction interface, the normal vector of the refraction plane is

$$\boldsymbol{\chi} = \frac{\boldsymbol{\varphi} \times \boldsymbol{N}}{\|\boldsymbol{\varphi} \times \boldsymbol{N}\|} = \begin{bmatrix} \chi_1 & \chi_2 & \chi_3 \end{bmatrix} \tag{17}$$

where × denotes the cross-product. The unit normal vector of the intersection CC$'$ of the refraction plane and the refractive interface is

$$\boldsymbol{\omega} = \frac{\boldsymbol{\chi} \times \boldsymbol{N}}{\|\boldsymbol{\chi} \times \boldsymbol{N}\|} = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \end{bmatrix} \tag{18}$$

Determine the coordinate of a point on line CC$'$, denoted as (x_0, y_0, z_0). Then, the line CC$'$ can be expressed as

$$\begin{cases} x = x_0 + \lambda \boldsymbol{\omega}_1 \\ y = y_0 + \lambda \boldsymbol{\omega}_2 \\ z = z_0 + \lambda \boldsymbol{\omega}_3 \end{cases} \tag{19}$$

where λ is the coefficient.

According to the relationship between refractive index and the speed of light, if the speed of light in vacuum is c, the speeds of light in the air and water are

$$\begin{cases} v_a = \frac{c}{n_1} \\ v_w = \frac{c}{n_2} \end{cases} \tag{20}$$

Let the coordinate of point C be (x_2, y_2, z_2). The lengths of the two optical paths OC and CP are

$$\begin{cases} l_{OC} = \sqrt{x_2^2 + y_2^2 + z_2^2} \\ l_{CP} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \end{cases} \tag{21}$$

According to the relationship between speed, time, and distance, the propagation time of the light between O and P is

$$t = \frac{l_{OC}}{v_a} + \frac{l_{CP}}{v_w} \tag{22}$$

The refraction point C is a point that satisfies Equation (19). When all refraction parameters are known, Equation (22) is a function of coefficient λ. According to Fermat's principle, the actual path of light passing between two fixed points in space is always the shortest optical path (or propagation time). So, the derivative of t is equal to 0, which is

$$f(\lambda) = \frac{dt}{d\lambda} = 0 \tag{23}$$

Solving the above formula, we can get the value of λ. Combining it with Equation (19), the coordinate of point C can be obtained. So, the complete propagation path of the light is established. For the right camera, the simulation process is similar, the only difference is that l_{OC} in Equation (21) is transformed into the following form

$$l_{OC} = \sqrt{(x_R - x_2)^2 + (y_R - y_2)^2 + (z_R - z_2)^2} \tag{24}$$

where (x_R, y_R, z_R) is the coordinate of the origin of the right camera coordinate system in the reference coordinate system.

The target pixel coordinate can be easily calculated based on the coordinate of point C and the camera parameters using the perspective imaging model, thus realizing the simulation of the imaging process.

2.3. Simulation Experimental Design

The purpose of the simulation experiments is to evaluate the impact of the deviations of the refraction parameters in the measurement model on the results and to provide guidance for the calibration of the refraction parameters and the construction of the measurement system. The cameras are placed in the air and the target in the water, as shown in Figure 3. The intrinsic and external parameters of the left and right cameras are known. The two cameras share the same refraction interface. The refraction parameters that affect the measurement model are the distance D between the left camera and the refraction interface, the medium refractive indices n_1 and n_2, and the refraction interface normal $\boldsymbol{N}$. Since the refraction angle of the light depends on the relative refractive index, the relative refractive index $n_r = n_1/n_2$ is discussed in the simulation experiments.

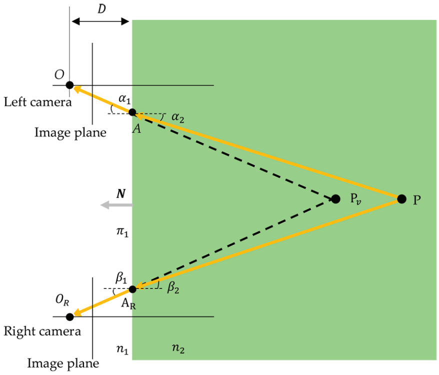


Figure 3. Air–water dual-medium visual-measurement scenario with one flat interface.

Take the left camera coordinate system as the reference coordinate system. We simulated a 2.8 × 2.8 m^2 square area and its distribution in the XOY plane of the left camera coordinate system is shown in Figure 4. The Z coordinate of the square area can take different values.

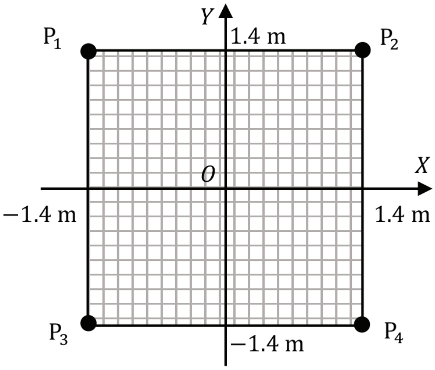


Figure 4. Target area and four feature points.

The intrinsic and external parameters of the left and right cameras and the target coordinates are known. The true values of the refraction parameters are set as shown in Table 1. The simulation process is shown in Figure 5. The experiments consist of four steps. (1) In step 1, based on the camera parameters, the refraction parameters without deviations, and the target coordinates, the simulation model proposed in Section 2.2 is used to simulate the coordinate of the refraction point. (2) In step 2, based on the camera parameters and the refraction point coordinate, the corresponding pixel coordinate is obtained using the perspective imaging model. (3) In step 3, based on the camera parameters, the target pixel coordinate, and the refraction parameters with certain deviations, the refractive measurement model described in Section 2.1 is used to estimate the target coordinate. (4) In step 4, the deviation between the estimated and the true coordinates of the target is calculated. The true coordinate of the target is $(X,\ Y,\ Z)$, and the estimated coordinate obtained in step 3 is $(X^e,\ Y^e, Z^e)$. The coordinate deviations are denoted as d_x, d_y, and d_z, respectively.

$$\begin{cases} d_x = X^e - X \\ d_y = Y^e - Y \\ d_z = Z^e - Z \end{cases} \tag{25}$$

Table 1. True values of the refraction parameters.

D	n_r	N
10 cm	$\frac{n_1}{n_2} = \frac{1}{1.33}$	$(0,0,1)$

The total coordinate deviation is given by d_p, where $d_p = \sqrt{d_x^2 + d_y^2 + d_z^2}$. Since the pixel coordinate of the target is obtained through the imaging model in step 2, they do not contain any errors. The refraction parameters used in step 3 are added with a certain amount of deviation, with the requirement that the deviation is added to only one parameter at a time. Three experiments were conducted. Experiment 1 discussed the sensitivity of the measurement model to the refraction parameters. Experiment 2 analyzed the influence of the fixed refraction-parameter deviation on the targets at different distances. Experiment 3 studied the change in the measurement results of fixed targets with the refraction parameter.

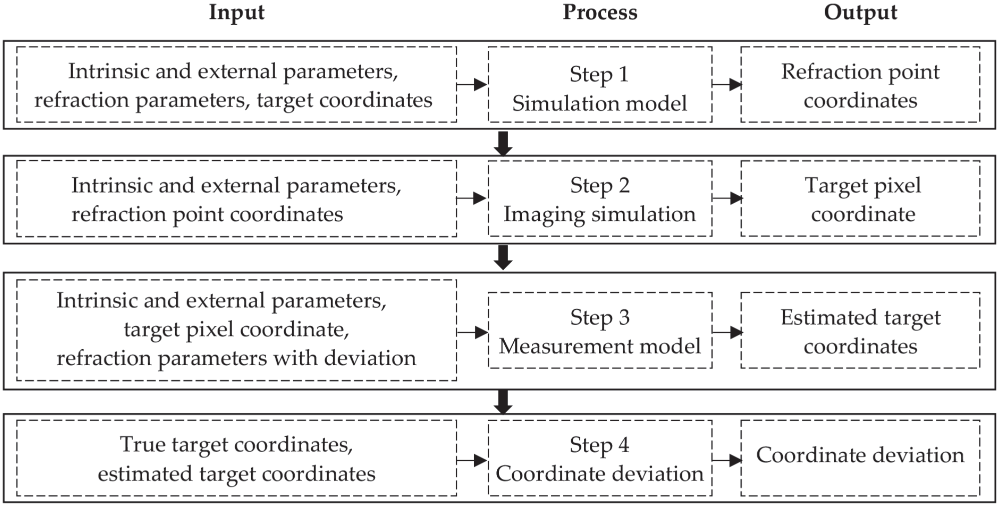

Input
Process
Output
Intrinsic and external parameters, refraction parameters, target coordinates
Step 1 Simulation model
Refraction point coordinates
Intrinsic and external parameters, refraction point coordinates
Step 2 Imaging simulation
Target pixel coordinate
Intrinsic and external parameters, target pixel coordinate, refraction parameters with deviation
Step 3 Measurement model
Estimated target coordinates
True target coordinates, estimated target coordinates
Step 4 Coordinate deviation
Coordinate deviation

Figure 5. Simulation process.

3. Results

3.1. Experiment 1

The aim of this experiment is to analyze the sensitivity of the dual-medium visual measurement model to deviations in refraction parameters. All targets are on the same plane with a Z-distance of 5 m as shown in Figure 4. After intentionally introducing deviations in the refraction parameters, the measured coordinates of the targets will differ from their true values, resulting in coordinate discrepancies. Sequentially introduce deviations in different refraction parameters until the maximum d_p reaches 1 cm.

First, analyze the impact of deviation in the relative refractive index. The true value of the relative refractive index is n_r. When the relative refractive index changes to $0.99843n_r$, the maximum d_p is 10 mm, as shown in Figure 6. In this case, the discrepancies in X and Y coordinate components are minimal, less than 0.5 mm, and d_p primarily depends on the discrepancy in Z coordinate component. Relative refractive index deviation is more likely to cause Z coordinate deviation. The maximum coordinate discrepancy occurs at the four corners in Figure 4.

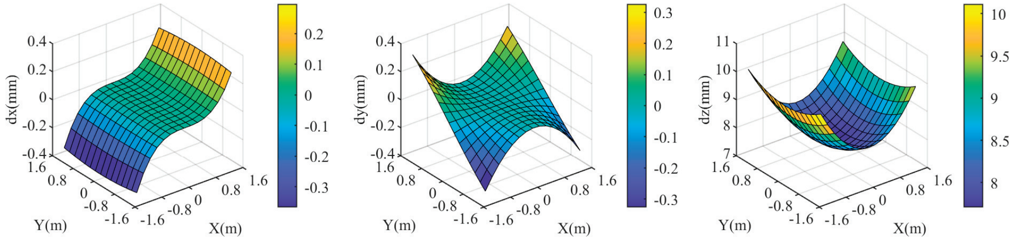


Figure 6. Coordinate deviation of the target area at $Z = 5$ m when the relative refraction is $0.99843n_r$. $\max\left(d_p\right) = 10$ mm.

Next, analyze the impact of deviation in the distance from the camera to the refractive interface. The true value of this distance is denoted as D. When the distance changes to $0.805D$, the maximum d_p is 10 mm, as shown in Figure 7. The deviation in D has a minimal impact on X and Y coordinate components of the target, less than 1 mm. It is more likely to cause discrepancies in the Z coordinate component, with d_p primarily depending on the discrepancy in the Z coordinate component. The maximum coordinate discrepancy occurs at the four corners in Figure 4.

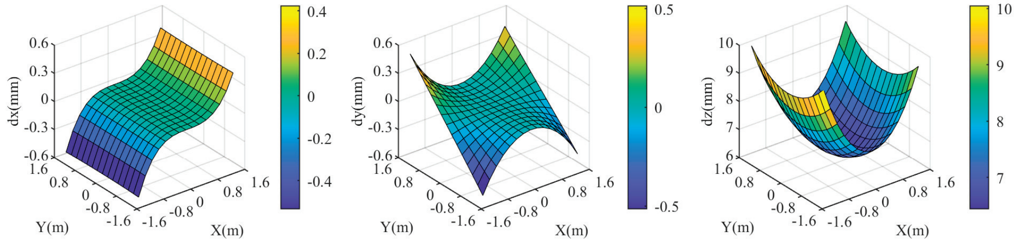


Figure 7. Coordinate deviation of the target area at $Z = 5$ m when the distance is $0.805D$. $\max\left(d_p\right) = 10$ mm.

Finally, analyze the impact of deviation in the normal direction of the refractive surface. The theoretical direction of the refractive surface normal is parallel to the optical axis of the left camera. The normal direction deviation is represented by the simultaneous rotation of its three attitude angles by the same angle. When the rotation angle is 0.195°, the maximum d_p is 10 mm, as shown in Figure 8. Deviations in the normal direction have a significant impact on all three coordinate components of the targets, especially the Y and Z coordinates. The extent of the impact varies across different points in the target area; points closer to the center are less affected, while points closer to the four corners are more affected.

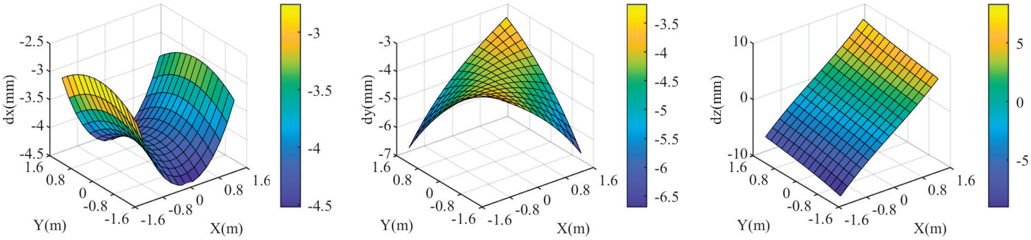


Figure 8. Coordinate deviation of the target area at $Z = 5$ m when the rotation angle of the normal direction of the refractive surface is 0.195°. $\max\left(d_p\right) = 10$ mm.

Combining Figures 6–8, it is evident that the dual-medium stereo-vision measurement model is highly sensitive to deviations in the relative refractive index. Even a deviation of 0.157% can result in a maximum comprehensive coordinate deviation of up to 10 mm in the target area. In contrast, the measurement model is less sensitive to deviations in the distance from the camera to the refractive interface and the normal direction of the refractive interface. This indicates that in underwater experiments or practical applications, special attention must be given to the calibration of the refractive indices of different mediums.

3.2. Experiment 2

In Experiment 1, we fixed the target area at $Z = 5$ m. However, with the same deviations in refraction parameters, the coordinate deviations of the target might differ if the distance between the target and the camera varies. Four different Z values were set at intervals of 2 m, ranging from 2 m to 8 m. Deviations were sequentially introduced in different refraction parameters: the relative refractive index changed to $0.99843n_r$, the distance from the camera to the refractive interface changed to $0.805D$, and the rotation angle of the refractive surface normal was 0.195°.

3.2.1. Target Plane

When the relative refractive index is $0.99843n_r$, the coordinate discrepancies of the three components in the target area are shown in Figures 9–11. For different Z values, the discrepancies in X and Y coordinate components remain relatively small. However, the smaller the Z value, the relatively larger the d_x and d_y discrepancies. At $Z = 2$ m, the maximum values of d_x and d_y are approximately ± 2 mm and ± 1 mm, respectively. The discrepancy in Z coordinate component, d_z, exhibits a significantly different variation pattern. The larger the Z value, the larger the d_z. At $Z = 2$ m, d_z is the smallest, with a maximum value of approximately 7.4 mm in the edge of the target area; at Z = 8 m, d_z is the largest, with both the maximum and minimum values of d_z in the target area being approximately 14 mm. Since d_z is significantly larger than d_x and d_y, when the relative refractive index deviation is constant, the closer the target is to the camera, the smaller the resulting d_p. The closer the target is to the center of the region, the smaller its d_p is.

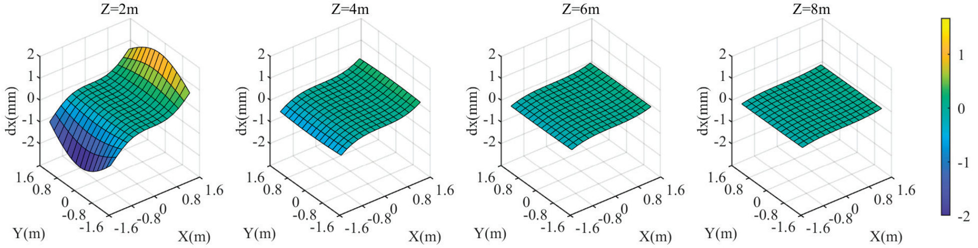


Figure 9. X coordinate deviation of the target area at different distances from the camera when the relative refraction is $0.99843n_r$.

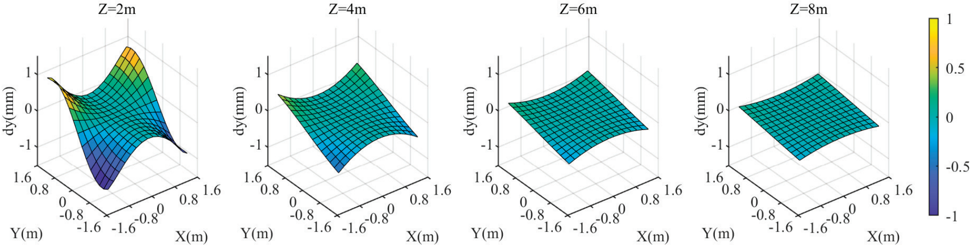


Figure 10. Y coordinate deviation of the target area at different distances from the camera when the relative refraction is $0.99843n_r$.

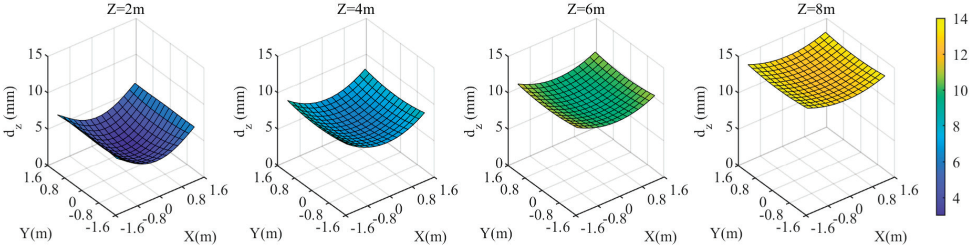


Figure 11. Z coordinate deviation of the target area at different distances from the camera when the relative refraction is $0.99843n_r$.

When the distance from the camera to the refractive interface is $0.805D$, the coordinate discrepancies of the three components in the target area are shown in Figures 12–14. When Z is greater than 4 m, as Z value increases, d_x, d_y, and d_z gradually decrease. At $Z = 2$ m, the maximum values of d_x and d_y are approximately ±18 mm, and the maximum value of d_z is approximately 54 mm, occurring at the four corners of the area. Overall, the farther the target plane is from the camera, the smaller the values of d_x, d_y, and d_z, resulting in a smaller d_p. Additionally, the closer the target is to the center of the area, the smaller the d_p.

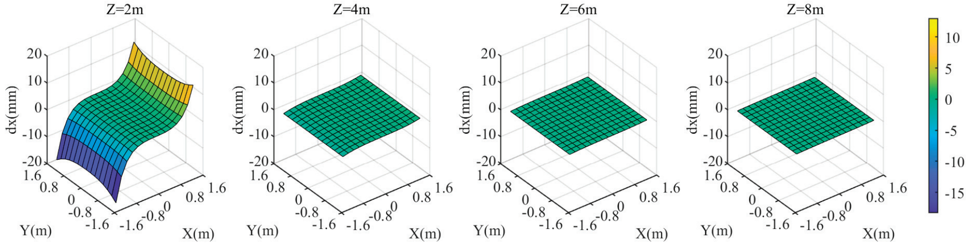


Figure 12. *X* coordinate deviation of the target area at different distances from the camera when the distance is 0.805*D*.

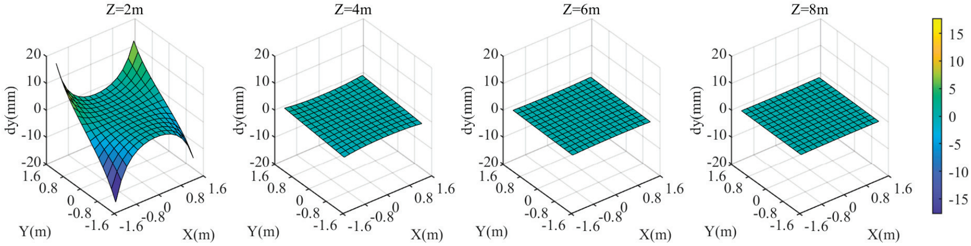


Figure 13. *Y* coordinate deviation of the target area at different distances from the camera when the distance is 0.805*D*.

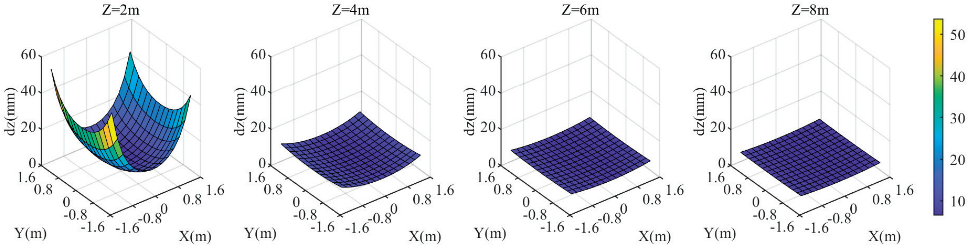


Figure 14. *Z* coordinate deviation of the target area at different distances from the camera when the distance is 0.805*D*.

When the three attitude angles of the refractive interface normal all change by 0.195°, the coordinate discrepancies of the three components in the target area are shown in Figures 15–17. As the *Z* value increases from 2 m to 8 m, d_x changes from positive to negative, and its absolute value generally increases. At *Z* = 2 m, there is a noticeable variation in d_x among different points in the target area, with d_x decreasing towards the center of the area. As the *Z* value increases, the variation in d_x among different points in the target area gradually diminishes. At *Z* = 8 m, d_x reaches −7 mm. The d_y component exhibits a similar trend with changes in *Z*; the larger the *Z* value, the greater the absolute value of d_y. At *Z* = 2 m, there is a noticeable variation in d_y among different points in the target area, with d_y decreasing towards the center of the area. The relationship between d_z and *Z* is relatively less pronounced. As *Z* increases from 2 m to 8 m, d_z slightly decreases. However, there are noticeable differences in d_z among different points in the target area.

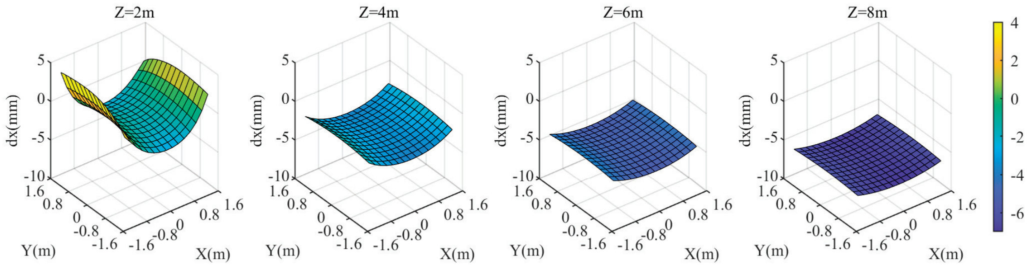


Figure 15. *X* coordinate deviation of the target area at different distances from the camera when the rotation angle of the normal direction of the refractive surface is 0.195°.

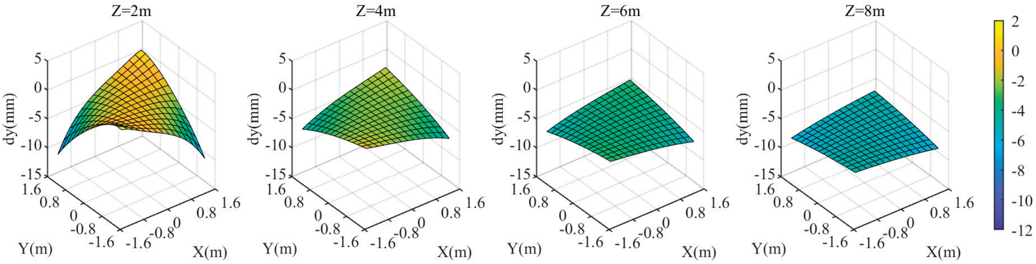


Figure 16. *Y* coordinate deviation of the target area at different distances from the camera when the rotation angle of the normal direction of the refractive surface is 0.195°.

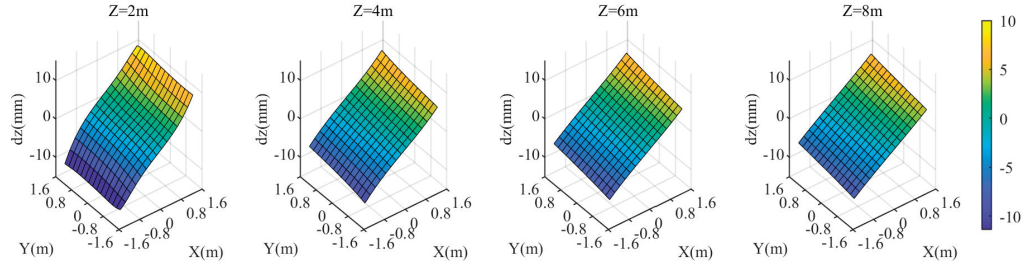


Figure 17. *Z* coordinate deviation of the target area at different distances from the camera when the rotation angle of the normal direction of the refractive surface is 0.195°.

3.2.2. Fixed Points

To further analyze the relationship between the coordinate deviations of the target and Z value, we selected the four corners of the target area shown in Figure 4 as feature points for the experiment. The Z values range from 0.5 m to 8 m. Based on the previous analysis, we know that when Z value is fixed, these four points have the largest d_p within the entire target area. We sequentially analyzed different refraction parameters, maintaining the same deviation magnitudes as discussed earlier.

When the relative refractive index is $0.99843n_r$, the coordinate discrepancies of the four points are shown in Figure 18. As Z increases, d_x and d_y exhibit similar trends: both initially increase rapidly, then decrease, with the rate of decrease being fast at first and then slowing down. The maximum values of d_x and d_y occur around Z = 2.2 m, about ±1 mm. For d_x, the trends for P_1 and P_3, as well as P_2 and P_4, are completely consistent. For d_y, the trends for P_1 and P_2, as well as P_3 and P_4, are very similar. d_z increases with increasing Z. Since d_z is much larger than d_x and d_y, d_p mainly depends on the magnitude of d_z. The trends for d_z and d_p are consistent for all four points.

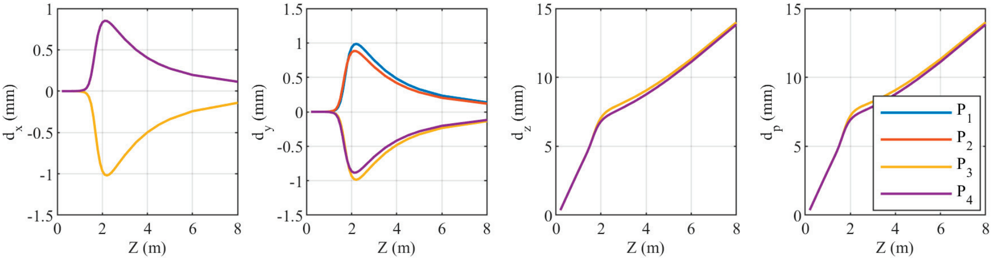


Figure 18. Relationship between the coordinate components of the four target points and Z when the relative refraction is $0.99843n_r$.

When the distance from the camera to the refractive interface is $0.805D$, the coordinate discrepancies of the four points are shown in Figure 19. Similar to Figure 18, as Z increases, d_x and d_y exhibit similar trends: both initially increase rapidly, then decrease, with the rate of decrease being fast at first and then slowing down. However, the maximum values of d_x and d_y occur around Z $=$ 1.7 m, reaching approximately ±30 mm. For d_x, the trends for P$_1$ and P$_3$, as well as for P$_2$ and P$_4$, are consistent. For d_y, the trends for P$_1$ and P$_2$, as well as for P$_3$ and P$_4$, are similar. d_z decreases with increasing Z, with the rate of decrease being fast at first and then slowing down, with an inflection point around $Z = 3$ m. Although the values of d_x and d_y around Z $=$ 1.7 m exceed ±20 mm, they are still an order of magnitude smaller than d_z. Therefore, d_p mainly depends on the magnitude of d_z. The trends for d_z and d_p are consistent for all four points; as Z value increases, d_p decreases.

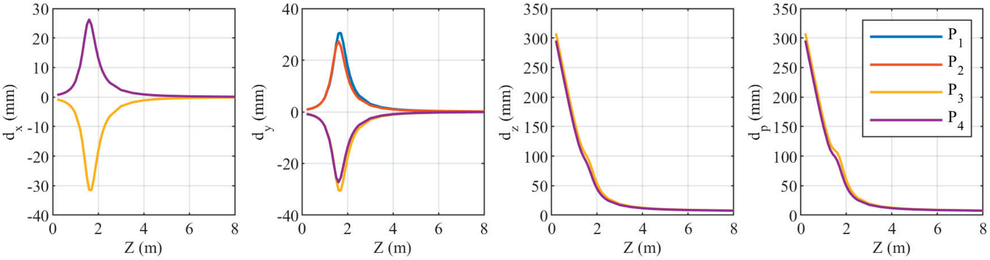


Figure 19. Relationship between the coordinate components of the four target points and Z when the distance is $0.805D$.

When the three attitude angles of the refraction interface normal change by 0.195°, the coordinate component deviations of the four points are shown in Figure 20. For Z less than 3 m, the variation trends of the three coordinate component deviations are quite complex. The variation trends of d_z differ significantly among different points. For points P$_2$ and P$_3$, d_z is significantly larger than d_x and d_y. In contrast, for points P$_1$ and P$_4$, the deviations in all three coordinate components are relatively small. When Z is greater than 3 m, d_x and d_y generally increase with increasing Z, while d_z shows no obvious variation. Additionally, d_x and d_y are similar in magnitude and significantly larger than d_z. For points P$_2$ and P$_3$, when Z is small, the deviation d_p is mainly determined by d_z. However, when Z is large, d_p is primarily influenced by d_x and d_y. For points P$_1$ and P$_4$, d_p is affected by all three coordinate components, and as Z increases, the influence of d_x and d_y becomes more significant.

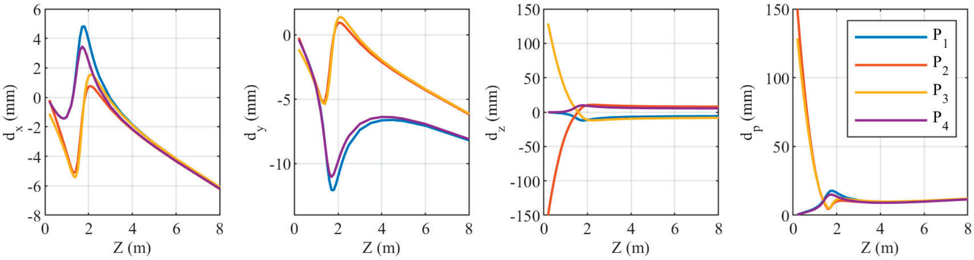


Figure 20. Relationship between the coordinate components of the four target points and Z when the rotation angle of the normal direction of the refractive surface is 0.195°.

3.3. Experiment 3

In the aforementioned experiments, we fixed the deviation of the refraction parameters. However, as the deviations of the refraction parameters vary, the target-coordinate deviations also change accordingly. Therefore, we conducted this experiment to investigate the relationship between target-coordinate deviations and refraction-parameter deviations. In this experiment, we continued to select the four feature points shown in Figure 4 as targets and set Z to 5 m. We then sequentially introduced deviation sequences into different refraction parameters.

The relationship between the deviations of the target coordinate and the three refraction-parameter deviations—relative refractive index, the distance from the camera to the refraction interface, and the direction of the refraction interface normal—is shown in Figures 21–23. d_x, d_y, and d_z all exhibit a linear relationship with the refraction-parameter deviations. The d_p plot displays a symmetric distribution with the true value of the refraction parameters as the axis of symmetry, indicating that both positive and negative deviations in the refraction parameters have the same effect on d_p. The smaller the deviation in the refraction parameters, the closer d_x, d_y, d_z, and d_p are to zero.

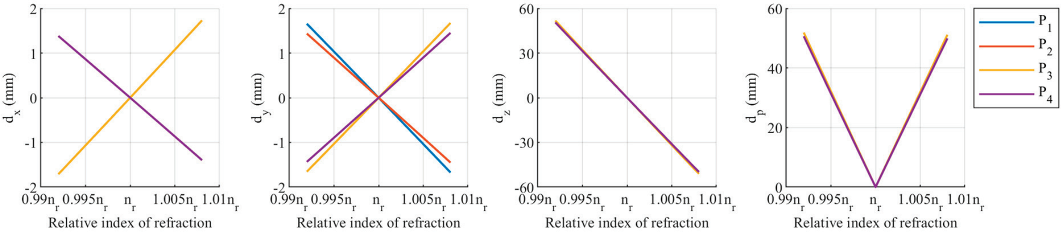


Figure 21. Relationship between target-coordinate deviation and relative refractive index deviation.

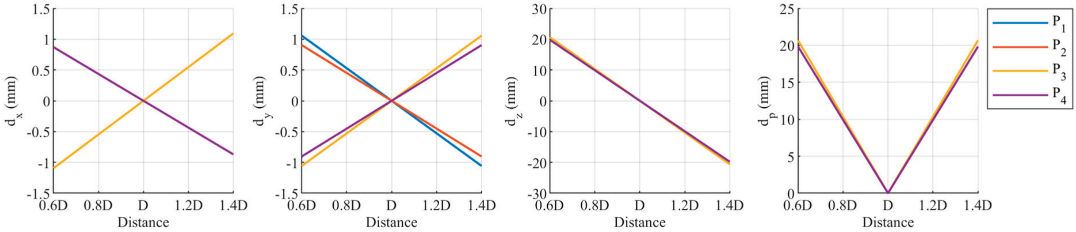


Figure 22. Relationship between target-coordinate deviation and camera-to-refractive-interface distance deviation.

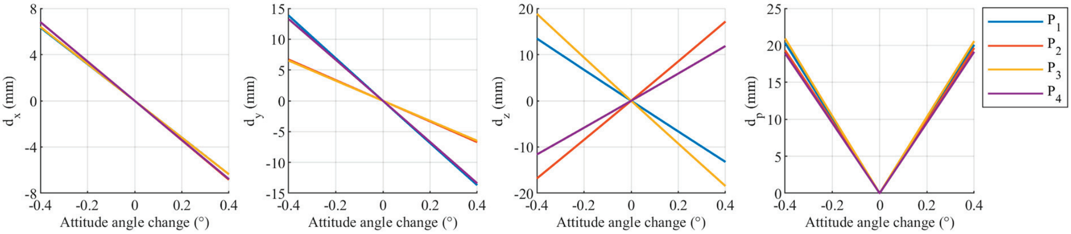


Figure 23. Relationship between target-coordinate deviation and normal direction of refraction interface.

4. Discussion

When the cameras are in the air and the target is in the water, the light path from the target to the camera undergoes refraction, causing the light to deviate from a straight line. In order to compensate for refraction in stereo-vision measurement, a theoretically rigorous approach is to use the refraction-measurement model. However, the refraction-measurement model involves refraction parameters such as the refractive indices of the medium, the distance from the camera to the refractive interface, and the normal of the refractive interface. These parameters generally require calibration. Due to the presence of errors, the calibrated refraction parameters inevitably differ from their true values. The experimental results in Section 3 demonstrate the impact of deviations in refraction parameters on stereo-vision measurement.

The impact of different refraction-parameter deviations on stereo-vision measurement varies, and the magnitude of this impact is related to the distribution of the target within the plane. Deviations in the relative refractive index and the distance between the target and the camera are more likely to cause discrepancies in the *Z* coordinate, while their effect on the *X* and *Y* coordinates is relatively minor. The total coordinate deviation of the target is primarily determined by the *Z*-coordinate deviation, and the closer the target is to the edge of the field of view, the greater its coordinate deviation. Deviations in the orientation of the refractive interface normal significantly affect all three coordinate components of the target. Regardless of the type of refraction-parameter deviation, the overall trend shows that the closer the target is to the edge of the field of view, the greater the coordinate deviation. A deviation of 0.157% in the relative refractive index can result in a maximum target-coordinate deviation of up to 1 cm, indicating that the stereo-visual measurement model is highly sensitive to deviations in the relative refractive index in the air–water condition. In underwater experiments or practical applications, special attention must be paid to the calibration of refractive indices of different media.

The impact of refraction-parameter deviations on stereo-visual measurement is related to the distance between the target and the camera, and the influence of different refraction parameters varies with the distance. When the deviation in the relative refractive index is constant, the closer the target is to the camera, the smaller the total coordinate deviation of the targets. When the deviation in the distance between the camera and the refractive interface is constant, as the distance between the target and the camera increases, the deviations in *X* and *Y* coordinates first increase rapidly, then decrease rapidly, and finally stabilize, while the deviation in *Z* coordinate decreases rapidly and gradually stabilizes. The deviation in the *Z* coordinate is an order of magnitude greater than that in the *X* and *Y* coordinates, so the overall coordinate deviation is determined by the *Z* coordinate deviation. For points at different positions within the field of view, the overall coordinate deviations caused by the deviations in the relative refractive index and the camera-refractive interface distance with the changes in the *Z* coordinate are approximately consistent. The impact of deviations in the orientation of the refractive interface normal varies more complexly with changes in the Z coordinate and is related to the distribution of the target. In underwater experiments or practical applications to mitigate the impact of deviations in the orientation of the refractive interface normal, attention must be paid to the distribution of the target within the field of view.

For the fixed targets, changes in the refraction parameters deviations result in linear variations in the X, Y, Z, and overall coordinate deviations. The larger the deviation in the refraction parameters, the greater the coordinate deviations. Whether the refraction parameter is greater or less than its true value, the impact on the overall coordinate deviation remains the same.

According to the experimental results, targets closer to the edge of the field of view are more susceptible to the influence of refraction-parameter deviations. This indicates that the current refraction-measurement model performs better for targets near the center of the field of view, while it may not fully eliminate refraction errors for targets near the edges.

5. Conclusions

This paper introduces a refraction-based visual-measurement model, establishes a stereo-vision simulation model, and conducts specific simulation experiments to study in detail the impact of deviations in various refraction parameters on stereo-visual measurement. The simulation accurately determines the coordinates of the refraction points based on the positions of the cameras and the targets, thereby obtaining the corresponding pixel coordinates used to estimate the targets' position. Experimental results demonstrate that the influence of different refraction-parameter deviations on stereo-visual measurement varies and is related to the distribution of the target within the field of view and the distance between the target and the camera. The overall coordinate deviation of the target is not necessarily minimized by simply increasing or decreasing the distance between the target and the camera; it must be determined based on the magnitude of the deviation in each refraction parameter. The stereo-visual measurement model is particularly sensitive to deviations in the relative refractive index in the air–water condition. Therefore, in underwater experiments or practical applications, careful calibration of the refractive indices of different media and attention to the distribution of targets within the field of view are crucial. The work and conclusions presented in this paper provide guidance for the construction of underwater stereo-vision measurement systems, refraction parameter calibration, underwater experiments, and practical applications. However, this study primarily conducted research by developing a simulation model and performing simulated experiments. Future work should include underwater experiments to validate the conclusions of this paper. In addition, the current refraction-measurement model is more suitable for targets located near the center of the field of view, whereas its effectiveness for targets at the edges is less than ideal, indicating the need for further optimization of the refraction-measurement model in the future.

Author Contributions: Conceptualization, G.L. and S.H.; Formal analysis, G.L.; Funding acquisition, G.L.; Investigation, G.L.; Methodology, G.L.; Project administration, N.Z.; Resources, G.L., Z.Y., S.H., N.Z. and K.Z.; Software, G.L.; Supervision, K.Z.; Validation, Z.Y., N.Z. and K.Z.; Visualization, G.L.; Writing—original draft, G.L.; Writing—review and editing, S.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Fundamental Research Funds for the Central Universities, grant number 2022QN1081.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Su, Z.; Pan, J.; Lu, L.; Dai, M.; He, X.; Zhang, D. Refractive three-dimensional reconstruction for underwater stereo digital image correlation. *Opt. Express* **2021**, *29*, 12131–12144. [CrossRef]
2. Ding, T.; Sun, C.; Chen, J. Cross-medium imaging model and calibration method based on refractive optical path for underwater morphology measurement. *Meas. Sci. Technol.* **2024**, *35*, 15205. [CrossRef]

3. Łuczyński, T.; Pfingsthorn, M.; Birk, A. The Pinax-model for accurate and efficient refraction correction of underwater cameras in flat-pane housings. *Ocean. Eng.* **2017**, *133*, 9–22. [CrossRef]
4. Chi, Y.; Yu, L.; Pan, B. Low-cost, portable, robust and high-resolution single-camera stereo-DIC system and its application in high-temperature deformation measurements. *Opt. Lasers Eng.* **2018**, *104*, 141–148. [CrossRef]
5. Wu, T.; Hou, S.; Sun, W.; Shi, J.; Yang, F.; Zhang, J.; Wu, G.; He, X. Visual measurement method for three-dimensional shape of underwater bridge piers considering multirefraction correction. *Autom. Constr.* **2023**, *146*, 104706. [CrossRef]
6. Beall, C.; Lawrence, B.J.; Ila, V.; Dellaert, F. 3D reconstruction of underwater structures. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems IEEE, Taipei, Taiwan, 18–22 October 2010; pp. 4418–4423.
7. Bruno, F.; Bianco, G.; Muzzupappa, M.; Barone, S.; Razionale, A.V. Experimentation of structured light and stereo vision for underwater 3D reconstruction. *ISPRS J. Photogramm. Remote Sens.* **2011**, *66*, 508–518. [CrossRef]
8. Chadebecq, F.; Vasconcelos, F.; Lacher, R.; Maneas, E.; Desjardins, A.; Ourselin, S.; Vercauteren, T.; Stoyanov, D. Refractive Two-View Reconstruction for Underwater 3D Vision. *Int. J. Comput. Vis.* **2020**, *128*, 1101–1117. [CrossRef]
9. Li, G.; Klingbeil, L.; Zimmermann, F.; Huang, S.; Kuhlmann, H. An Integrated Positioning and Attitude Determination System for Immersed Tunnel Elements: A Simulation Study. *Sensors* **2020**, *20*, 7296. [CrossRef] [PubMed]
10. Liu, S.; Xu, H.; Lin, Y.; Gao, L. Visual Navigation for Recovering an AUV by Another AUV in Shallow Water. *Sensors* **2019**, *19*, 1889. [CrossRef] [PubMed]
11. Cowen, S.; Briest, S.; Dombrowski, J. Underwater docking of autonomous undersea vehicles using optical terminal guidance. In Proceedings of the Oceans '97. MTS/IEEE Conference, Halifax, NS, Canada, 6–9 October 1997; Volume 2, pp. 1143–1147.
12. Liu, S.; Ozay, M.; Okatani, T.; Xu, H.; Sun, K.; Lin, Y. Detection and Pose Estimation for Short-Range Vision-Based Underwater Docking. *IEEE Access* **2019**, *7*, 2720–2749. [CrossRef]
13. Treibitz, T.; Schechner, Y.; Kunz, C.; Singh, H. Flat Refractive Geometry. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 51–65. [CrossRef]
14. Treibitz, T.; Schechner, Y.Y. Active Polarization Descattering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 385–399. [CrossRef] [PubMed]
15. Schechner, Y.Y.; Karpel, N. Recovery of Underwater Visibility and Structure by Polarization Analysis. *IEEE J. Ocean. Eng.* **2005**, *30*, 570–587. [CrossRef]
16. Yamashita, A.; Kawanishi, R.; Koketsu, T.; Kaneko, T.; Asama, H. Underwater sensing with omni-directional stereo camera. In Proceedings of the 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), Barcelona, Spain, 6–13 November 2011; pp. 304–311.
17. Menna, F.; Nocerino, E.; Fassi, F.; Remondino, F. Geometric and Optic Characterization of a Hemispherical Dome Port for Underwater Photogrammetry. *Sensors* **2016**, *16*, 48. [CrossRef] [PubMed]
18. She, M.; Nakath, D.; Song, Y.; Köser, K. Refractive geometry for underwater domes. *ISPRS J. Photogramm. Remote Sens.* **2022**, *183*, 525–540. [CrossRef]
19. Bosch, J.; Gracias, N.; Ridao, P.; Ribas, D. Omnidirectional Underwater Camera Design and Calibration. *Sensors* **2015**, *15*, 6033–6065. [CrossRef]
20. Shmutter, B. Orientation Problems in Two-Medium Photogrammetry. *Photogramm. Eng.* **1967**, *33*, 1421–1428.
21. Rinner, K. Problems of Two-Medium Photogrammetry. *Photogramm. Eng.* **1969**, *35*, 275–282.
22. Masry, S.E.; Konecny, G. New Programs for the Analytical Plotter. *Photogramm. Eng.* **1970**, *36*, 1269–1276.
23. Kwon, Y.; Casebolt, J.B. Effects of light refraction on the accuracy of camera calibration and reconstruction in underwater motion analysis. *Sports Biomech.* **2006**, *5*, 95–120. [CrossRef]
24. Fabio, M.; Erica, N.; Salvatore, T.; Fabio, R. A photogrammetric approach to survey floating and semi-submerged objects. In Proceedings of the Videometrics, Range Imaging, and Applications XII, and Automated Visual Inspection, Munich, Germany, 13–16 May 2013.
25. Kang, L.; Wu, L.; Yang, Y. Experimental study of the influence of refraction on underwater three-dimensional reconstruction using the SVP camera model. *Appl. Opt.* **2012**, *51*, 7591–7603. [CrossRef] [PubMed]
26. Lavest, J.M.; Rives, G.; Laprest, J.T. Dry camera calibration for underwater applications. *Mach. Vision. Appl.* **2003**, *13*, 245–253. [CrossRef]
27. Kang, L.; Wu, L.; Wei, Y.; Lao, S.; Yang, Y. Two-view underwater 3D reconstruction for cameras with unknown poses under flat refractive interfaces. *Pattern Recognit.* **2017**, *69*, 251–269. [CrossRef]
28. Agrawal, A.; Ramalingam, S.; Taguchi, Y.; Chari, V. A theory of multi-layer flat refractive geometry. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012.
29. Li, R.; Li, H.; Zou, W.; Smith, R.G.; Curran, T.A. Quantitative photogrammetric analysis of digital underwater video imagery. *IEEE J. Ocean. Eng.* **1997**, *22*, 364–375. [CrossRef]
30. Jordt-Sedlazeck, A.; Koch, R. Refractive Structure-from-Motion on Underwater Images. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Sydney, NSW, Australia, 1–8 December 2013.
31. Yau, T.; Gong, M.; Yang, Y. Underwater Camera Calibration Using Wavelength Triangulation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013.
32. Telem, G.; Filin, S. Photogrammetric modeling of underwater environments. *ISPRS J. Photogramm. Remote Sens.* **2010**, *65*, 433–444. [CrossRef]

33. Chen, X.; Yang, Y.H. Two-View Camera Housing Parameters Calibration for Multi-layer Flat Refractive Interface. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.
34. Dolereit, T.; von Lukas, U.F.; Kuijper, A. Underwater stereo calibration utilizing virtual object points. In Proceedings of the Oceans 2015, Genova, Italy, 18–21 May 2015.
35. Qiu, C.; Wu, Z.; Kong, S.; Yu, J. An Underwater Micro Cable-Driven Pan-Tilt Binocular Vision System with Spherical Refraction Calibration. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–13. [CrossRef]
36. Qi, G.; Shi, Z.; Hu, Y.; Fan, H.; Dong, J. Refraction calibration of housing parameters for a flat-port underwater camera. *Opt. Eng.* **2022**, *61*, 104105. [CrossRef]
37. Ma, Y.; Zhou, Y.; Wang, C.; Wu, Y.; Zou, Y.; Zhang, S. Calibration of an underwater binocular vision system based on the refraction model. *Appl. Opt.* **2022**, *61*, 1675–1686. [CrossRef]
38. Tong, Z.; Gu, L.; Shao, X. Refraction error analysis in stereo vision for system parameters optimization. *Measurement* **2023**, *222*, 113650. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

Article

Visual Localization Method for Unmanned Aerial Vehicles in Urban Scenes Based on Shape and Spatial Relationship Matching of Buildings

Yu Liu [1,2,*], Jing Bai [1] and Fangde Sun [2,3]

[1] School of Artificial Intelligence, Xidian University, Xi'an 710071, China; baijing@mail.xidian.edu.cn
[2] CETC Key Laboratory of Aerospace Information Applications, Shijiazhuang 050081, China; 195001017@csu.edu.cn
[3] Research Institute of Aerospace Technology, Central South University, Changsha 410017, China
* Correspondence: liuyu0213@stu.xidian.edu.cn

Abstract: In urban scenes, buildings are usually dense and exhibit similar shapes. Thus, existing autonomous unmanned aerial vehicle (UAV) localization schemes based on map matching, especially the semantic shape matching (SSM) method, cannot capture the uniqueness of buildings and may result in matching failure. To solve this problem, we propose a new method to locate UAVs via shape and spatial relationship matching (SSRM) of buildings in urban scenes as an alternative to UAV localization via image matching. SSRM first extracts individual buildings from UAV images using the SOLOv2 instance segmentation algorithm. Then, these individual buildings are subsequently matched with vector e-map data (stored in .shp format) based on their shape and spatial relationship to determine their actual latitude and longitude. Control points are generated according to the matched buildings, and finally, the UAV position is determined. SSRM can efficiently realize high-precision UAV localization in urban scenes. Under the verification of actual data, SSRM achieves localization errors of 7.38 m and 11.92 m in downtown and suburb areas, respectively, with better localization performance than the radiation-variation insensitive feature transform (RIFT), channel features of the oriented gradient (CFOG), and SSM algorithms. Moreover, the SSRM algorithm exhibits a smaller localization error in areas with higher building density.

Keywords: UAV; vision-based localization; spatial relationship matching; urban scene

Citation: Liu, Y.; Bai, J.; Sun, F. Visual Localization Method for Unmanned Aerial Vehicles in Urban Scenes Based on Shape and Spatial Relationship Matching of Buildings. *Remote Sens.* **2024**, *16*, 3065. https://doi.org/10.3390/rs16163065

Academic Editors: Zhetao Zhang, Guorui Xiao, Zhixi Nie, Vagner Ferreira and Giuseppe Casula

Received: 22 June 2024
Revised: 9 August 2024
Accepted: 19 August 2024
Published: 20 August 2024

1. Introduction

Owing to the small size and suitable flexibility of unmanned aerial vehicles (UAVs), their application fields are constantly expanding, and they are now widely employed in map surveying and mapping, emergency search and rescue, and military reconnaissance fields. Accurate positioning is a prerequisite for UAVs to realize precise control and mission execution. A global navigation satellite system (GNSS) provides a universal localization method for UAVs. However, GNSS signals readily exhibit interference in complex electromagnetic environments, where the localization accuracy and reliability are substantially affected [1,2]. Therefore, it is necessary to assist UAVs in obtaining localization information by other means.

UAVs generally carry small, low-cost vision sensors when performing tasks; thus, vision-based localization methods are generally used to replace or as a supplement to GNSS [3,4]. In the vision-based localization method, vision sensors carried by UAVs are used to sense the environment, after which the position of the UAVs can be estimated upon image information processing and analysis. Vision-based localization methods do not rely on external signals and provide the advantage of high anti-interference ability [5,6]. For outdoor scenes, the absolute visual localization method based on image matching is the most commonly employed method. The basic principle of vision-based localization is to match real-time UAV images with geo-referenced data, which are usually pre-collected UAV images or satellite remote sensing images. Then, the latitude and longitude corresponding to each pixel of the UAV images can be obtained, and finally, the UAV position can be determined [1,7]. Image matching methods, such as the mutual information (MI) and scale-invariant feature transform (SIFT) methods, generally involve the use of grayscale and texture features to describe image similarity [8,9], leading to failure when there are temporal, imaging perspective, and lighting condition differences between the geo-referenced images and UAV images.

To address the above challenges, a few researchers have proposed the use of map-based matching methods to achieve UAV localization. For example, Nassar [10] considered the shape and area information of individual buildings to match UAV images and a map produced from geo-referenced images, referred to as semantic shape matching (SSM). Map-based matching methods can overcome the differences in color and lighting between UAV images and geo-referenced data [11]. However, the existing map-based matching methods entail the use of geo-referenced maps produced from remote sensing images, which indicates that these maps may not be accurate. Moreover, buildings within a given city are often dense and exhibit similar shapes. Thus, the use of the shape and area of buildings alone cannot capture their uniqueness. Therefore, prior methods can hardly match individual buildings in urban scenes.

According to theory in the field of geographic information science, the most accurate feature of a spatial scene is the spatial relationship [12], and the spatial distribution of buildings can uniquely characterize a given scene. To address the issue of insufficient feature descriptions and the possible failure of existing map-based matching methods, we propose a novel autonomous UAV localization method for urban scenes based on the shape and spatial relationship matching (SSRM) of buildings in urban scenes. The main contributions of our study are as follows:

(1) In SSRM, vector e-map data (stored in .shp format) are used as geo-referenced data instead of pre-collected images or image-based map-related data. The e-map data can comprehensively reflect the individual and spatial relationship characteristics of buildings while also reducing the amount of data prestored on UAVs.

(2) We propose a scene matching method in which the shape information and spatial relationships of buildings are used to match UAV images and geo-referenced data. Compared with existing map-based matching methods, increased consideration is given to the spatial relationships between buildings, thus greatly enhancing the robustness of the matching process.

(3) The effectiveness of the SSRM method is verified via simulation flight data. Moreover, we compare the SSRM method with the radiation-variation insensitive feature transform (RIFT) feature matching algorithm [13], the channel features of oriented gradient (CFOG) template matching algorithm [14], and the SSM map-based algorithm [10]. The consideration of the shape and spatial relationships of buildings ensures the accuracy of scene matching and provides far better localization accuracy.

The remainder of the article is organized as follows: In Section 2, related works are introduced. Our method is described in detail in Section 3. Section 4 presents the datasets and experiments used to validate our method. Section 5 provides the validation results and the comparative experimental results. The factors that may introduce errors are discussed in Section 6. Finally, conclusions are noted, and future research directions are outlined in Section 7.

2. Related Works

Absolute visual localization methods are classified into two categories according to the geo-referenced data used for matching UAV images: image-based matching methods and map-based matching methods.

2.1. Image-Based Matching Methods

Image-based matching methods generally involve the use of high-resolution satellite remote sensing images or pre-collected UAV images with geographic coordinates as geo-referenced images, and UAV images are matched with geo-referenced images. The different image matching algorithms can be subcategorized into template matching methods, feature matching methods, and deep learning-based matching methods.

2.1.1. Template Matching Methods

Template matching methods typically entail the use of robust similarity metrics to evaluate image similarity within a predefined window. Grayscale information within the window is usually employed to compute the similarity metric [15]. In template matching, it is usually assumed that there is a slight difference in appearance between the UAV and geo-referenced images, so the sum of squared differences (SSD) and normalized cross correlation (NCC) are often adopted as standard matching templates [16,17]. However, there are inevitably temporal differences, illumination differences, and imaging perspective differences between UAV images and geo-referenced images, which usually lead to the failure to match UAV images with geo-referenced images [1]. The MI method is more robust to radiometric differences and can resist nonlinear grayscale distortion, and this method aims to calculate the mutual information value in the search window between the UAV images and geo-referenced images [18].

In addition to using grayscale information to calculate the template similarity metric, the geometric structure and morphological features can be used to characterize the similarity between images. For projecting and quantizing histograms of oriented gradients (PQ-HOG) algorithms [19], oriented gradients are used as the basic similarity metric. Moreover, the phase congruency (PC) exhibits greater resistance to illumination and contrast variations, so the histogram of oriented phase congruency (HOPC) method utilizes phase congruency intensity and orientation information to construct similarity descriptors [20]. On the basis of the HOPC method, the CFOG method extracts local descriptors at each pixel to form a pixelwise feature representation, and fast Fourier transform (FFT) is used to define a fast similarity measure in the frequency domain for improving the computational efficiency [14]. The CFOG method has been successfully applied in multiple commercial software programs and has been demonstrated to achieve higher matching accuracy than does the HOPC method [21].

When land cover change occurs between the UAV and geo-referenced image, the above methods may fail. In addition, template matching methods usually require iteration of the geo-referenced image to calculate the similarity within different windows, after which the window with the greatest similarity can be selected as the matching result, thus requiring intensive computations.

2.1.2. Feature Matching Methods

Feature matching methods do not search the whole image but extract representative feature points, so the number of computations is greatly reduced [22]. However, seasonal or color differences between the UAV and geo-referenced images usually lead to the failure of classical feature operators, such as the SIFT, ORB, and SURF operators. Therefore, researchers have proposed image matching schemes that are independent of image color differences. Mantelli et al. proposed the abBRIEF operator, which aims to use the CIE color space to match UAV and geo-referenced images [23]. Ye et al. constructed a feature detector, namely, the minimum moment of phase congruency (MMPC)-Lap method, to robustly determine luminance and contrast variations [20]. Li et al. proposed the RIFT method, using PC for feature point detection and the maximum index map (MIM) for feature description, which makes it unsensitive to nonlinear radiation distortions [13].

In noncooperative areas, high-resolution satellite remote sensing images are usually used as geo-referenced images, and the scale and color differences between UAV and geo-referenced images are generally obvious. In addition, when there are imaging perspective differences between the UAV and geo-referenced images, the complex three-dimensional structure within a city can lead to different areas of exposure for buildings, and a land object can exhibit varying appearances. Therefore, feature matching methods typically yield more mismatched points and a poorer performance.

2.1.3. Deep Learning-Based Matching Methods

Compared with template matching and feature matching methods, deep learning-based methods provide superior learning and representation capabilities for deep features of images and provide obvious advantages in scene adaptation and robustness. Initially, deep learning-based methods were used to extract more representative feature points, which were then combined with traditional similarity metrics to increase the image alignment accuracy [24,25]. Thereafter, research gradually shifted to end-to-end image matching. Danial et al. proposed a self-supervised image matching framework, SuperPoint, which first utilized feature points of simple geometric shapes (e.g., triangles, rectangles) in a synthetic dataset to train a pre-trained detector, namely, MagicPoint. The homographic adaptation module was subsequently used to help relate the feature points across different perspectives and scales to increase the feature point redetection rate and the cross-domain utility [26]. Sarlin et al. proposed a network, namely, SuperGLUE, that can simultaneously perform feature matching and nonmatching point filtering. In SuperGLUE, a graph neural network (GNN) is used to predict the loss function of matching, solve the differentiable optimal transport problem, and ultimately realize feature matching. Moreover, SuperGLUE uses a flexible content aggregation mechanism based on the attention mechanism, which enables it to perceive potential 3D scenes and perform feature matching [27]. Inspired by the SuperGLUE architecture, Sun et al. proposed a new detector-free local image feature matching method, namely, LoFAR. In the LoFAR method, a transformer with self- and cross-attention layers is employed to obtain the feature descriptors of two images. Dense matching is achieved at the coarse-grained level, and dense matching is subsequently refined at the subpixel level [28].

The SuperPoint, SuperGLUE, and LoFAR methods are widely used in simultaneous localization and mapping (SLAM), which also indicates that they require that the matched images exhibit the same size. In outdoor image matching, UAV images and geo-referenced data are always of different sizes, and iteration of the whole geo-referenced image for matching with the UAV image is needed. Furthermore, deep learning-based methods require numerous samples for training, and many geo-referenced images must be stored on UAVs in advance, especially for large flight areas.

2.2. Map-Based Matching Methods

Instead of using pre-collected or satellite remote sensing images as geo-referenced data, maps reflecting land cover are used in map-based matching methods.

UAV images may exhibit different colors due to weather and temporal differences, but the land cover type rarely changes. Therefore, land cover data can be used as geo-referenced data. Masselli et al. employed land cover classification results from Google Earth images as geo-referenced data [29]; during flight, the collected images were classified to land cover and matched with the aforementioned geo-referenced map. Choi proposed a building ratio map (BRM) method, in which the building layer from pre-collected remote sensing images is applied as geo-referenced data. In the BRM method, the ratio of buildings within circles with different radii is determined to produce feature vectors, which are then used to estimate the similarity between UAV images and the building layer for retrieving candidate UAV flight areas [11]. Hao et al. abstracted buildings as point clouds, which could be further regarded as geo-referenced data to match UAV images via a point cloud matching algorithm [30]. Wang et al. reported that the shadow map is a stable feature for scene matching navigation and proposed a shadow-based matching method for the localization of low-altitude flight UAVs [31].

Currently, only a few studies have focused on map-based matching, and there is no unanimously recognized scheme. Moreover, existing methods still entail the use of pre-collected remote sensing images as geo-referenced data, which introduces additional errors.

3. Methodology

In the absence of GNSS signals, if the takeoff position of the UAV is known, the inertial navigation system (INS) can provide rough location information for the UAV within a certain period. The flight altitude of the UAV is generally derived from barometric altimeters. Furthermore, the internal parameters of the camera are known, so the spatial resolution of the UAV images can be calculated directly. The magnetometer can provide UAV flight orientation information. Thus, UAV images can be geometrically corrected with flight orientation information [32,33]. When the camera is at the real nadir, the latitude and longitude of the central pixel of an image reflect the UAV position.

If the takeoff position, spatial resolution, and orientation of UAV images are known, our method first extracts the roof outlines of individual buildings from UAV images, after which they are matched with building entities in the e-map based on their shape and spatial relationship. The buildings with the highest matched frequency can be used to generate control points accordingly, which ultimately can be applied to determine the UAV position. The overall flow is shown in Figure 1.

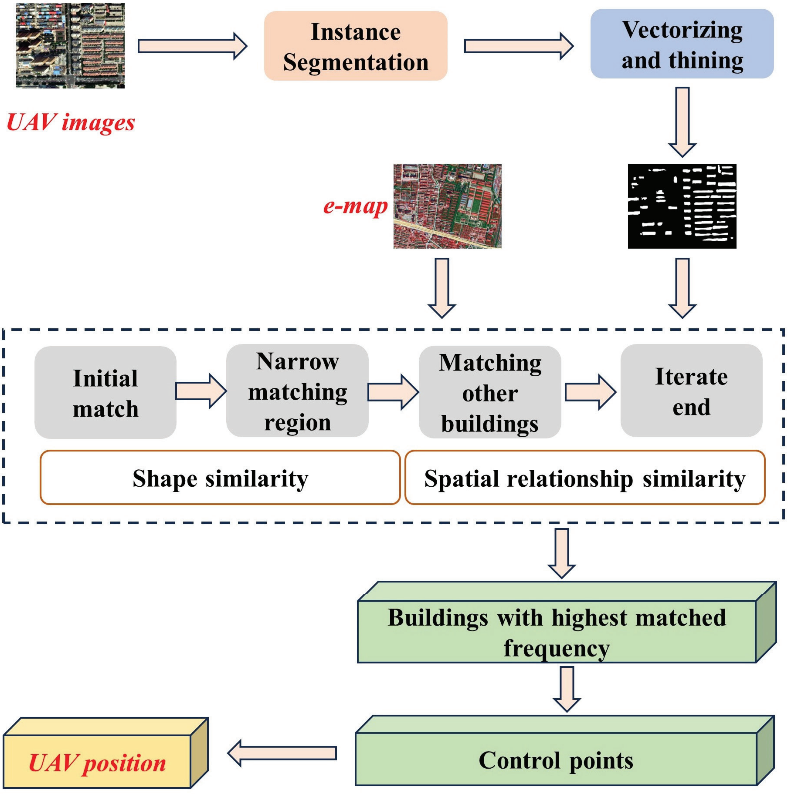


Figure 1. Flowchart of UAV autonomous localization in urban scenarios based on shape and spatial relationship matching of buildings.

If the UAV location at takeoff is also unknown, then as long as the city where the UAV is located is known and a rough location is manually specified for the UAV, SSRM takes more time only in the first matching. And once the approximate position of the UAV at a certain moment is determined, the matching range can be narrowed down.

3.1. Individual Building Extraction

The instance segmentation algorithm not only exhibits the characteristics of semantic segmentation (for pixel-level classification and effective outline extraction) but also exhibits the characteristics of target detection (for distinguishing and locating different individuals of the same category). Therefore, the instance segmentation algorithm is generally selected to extract individual buildings from remote sensing images. Owing to the need for fast image interpretation in UAV localization and considering that the buildings in UAV images are almost always large targets, we use the SOLOv2 model for individual building extraction, thereby relying on its efficiency and favorable instance segmentation ability for large targets [34,35]. The SOLOv2 model, based on the SOLO architecture, uses ResNet101 trained on the ImageNet dataset as the backbone network to extract features. And then, the feature pyramid network (FPN) outputs feature maps of different sizes as inputs for the prediction head, which divides the input feature maps into S × S grids. If the center of the object falls within a grid cell, the grid is responsible for (1) predicting the semantic category by a category branch and (2) segmenting the object instance by a mask branch. Finally, the SOLOv2 model uses matrix non-maximum suppression (NMS) to parallelize the NMS, which renders it advantageous in terms of both its speed and accuracy.

Implementation Details. We use the SOLOv2 model under the MMDetection framework. Multiscale sampling and random flipping are employed for image enhancement during training to adapt to buildings with varying scales at different spatial resolutions. The base learning rate is set to 0.01 with a weight decay of 0.0001 and a momentum of 0.9. The Dice coefficient, which is a metric used to estimate the similarity between the image segmentation results and ground-truth data, is adopted as the loss function of the mask branch. The Dice coefficient represents the degree of overlap between the image segmentation target and the ground truth. The maximum number of iteration epochs is set to 5000 for training.

After extracting the individual buildings from UAV images, the buildings are converted into outline information and stored in the form of image coordinate strings. To reduce the computational burden for subsequent scene matching, the building outlines are thinned via the Douglas–Peucker algorithm [36]. Furthermore, individual buildings with areas smaller than 50 pixels are removed. Considering that the UAV orientation and flight altitude are known, the resolution of the images can be estimated, and the central pixel coordinates, actual area, and aspect ratio of the minimum bounding rectangle for all buildings can be calculated.

3.2. Scene Matching and UAV Position Determination

It is necessary to determine the scene of UAV images before the UAV spatial location can be obtained, which suggests the matching of semantic information interpreted from the UAV images with the e-map to determine which building within the geographic space corresponds to the building in the UAV images. Owing to the complexity of urban scenes and the high frequency of similar buildings, it is unreliable to use only the shape features of buildings for scene matching, such as in SSM [10]. Spatial scenes are most accurately characterized by spatial relationships rather than the shape and size of targets [12]. Therefore, we design a vector scene matching method based on the shape and spatial relationships of individual buildings, and the basic concept of this approach is shown in Figure 2.

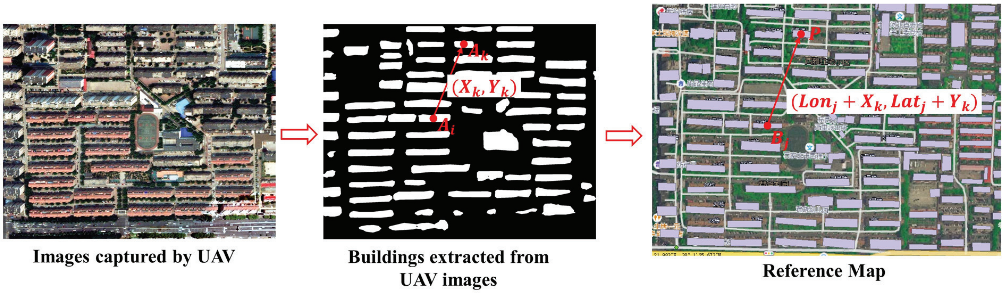


Figure 2. Basic diagram of the SSRM method.

Assuming that the UAV localization uncertainty is R = 500 m, A is the UAV image, and *Map* denotes the geo-referenced map, then the longest side length of each UAV image is r. The process of matching A and *Map* is as follows:

(1) Iterate over building individuals A_i ($1 \leq i \leq N$) extracted from UAV images.

(2) Using the initial position of the UAV as the center and a radius of ($R + r$) m, perform a spatial retrieval on *Map*, and the retrieval result is B. The central geographic coordinates of the buildings B_j ($1 \leq j \leq M$) in B are denoted as (Lon_j, Lat_j).

(3) Compute the Euclidean distances (spatial resolution × pixel distance) of the central pixel (x_i, y_i) of A_i from the four boundaries of the current UAV image. The distance between A_i and the upper boundary is L_t, the distance from the lower boundary is L_b, the distance from the left boundary is L_l, and the distance from the right boundary is L_r.

(4) Iterate over B_j. If the area ratio of B_j to A_i $AreaRatio(B_j, A_i)$ and the minimum bounding rectangle aspect ratio of B_j to A_i $AspectRatio(B_j, A_i)$ satisfy the following relationship:

$$0.5 \leq AreaRatio\left(B_j, A_i\right) \leq 2 \tag{1}$$

$$0.5 \leq AspectRatio\left(B_j, A_i\right) \leq 2 \tag{2}$$

then, B_j matches A, and proceed to the next step. Otherwise, continue iterating over B_j.

(5) With B_j as the center, construct the buffer zone along four directions in B. The buffer distance on the north side is L_t, the buffer distance on the south side is L_b, the buffer distance on the west side is L_l, and the buffer distance on the east side is L_r. The buildings within the buffer zone constitute the final candidate map to be matched.

(6) Calculate the Euclidean vector distance (X_k, Y_k) between the central pixel of each $A_k(k \neq i)$ and the central pixel of A_i and apply the spatial analysis technique to determine whether point $P(Lon_j + X_k, Lat_j + Y_k)$ is located in a building on the candidate map. If not, there is no matching building for A_i, and if P is located in building B_c on the candidate map, we further determine whether the following relationships between A_k and B_c are satisfied:

$$0.5 \leq AreaRatio(B_c, A_k) \leq 2 \tag{3}$$

$$0.5 \leq AspectRatio(B_c, A_k) \leq 2 \tag{4}$$

If the above relationships hold, B_c is considered successfully matched with A_k, and the number of matches is increased by one. Otherwise, A_k is not associated with a matching building.

Finally, the frequency of successfully matched pairs of buildings in A and B is counted, and the top N ($N \leq 15$) pairs of (A_m, B_m) with matched frequencies greater than 3 are adopted as the final successfully matched pairs. Considering the uncertainty in the instance segmentation algorithm when extracting building outlines, we assume that the central coordinates of the successfully matched $A_m(x_m, y_m)$ and $B_m(Lon_m, Lat_m)$ are the ground control point pairs. When the UAV camera is at the real nadir, the geographic coordinates corresponding to the central pixel of the image are the geographic coordinates of the UAV. Therefore, as expressed in Equations (5)–(10), the geographic coordinate of the central pixel of the UAV image can be calculated via a linear regression algorithm, and control points with an error greater than 2 times the standard deviation are excluded in the regression process:

$$b = \frac{\sum_{m=1}^{15}\left(Lat_m - \overline{Lat}\right)(y_m - \overline{y})}{\sum_{m=1}^{15}(y_m - \overline{y})^2} \tag{5}$$

$$a = \overline{Lat} - b\overline{y} \tag{6}$$

$$Lat_{center} = by_{center} + a \tag{7}$$

$$d = \frac{\sum_{m=1}^{15}\left(Lon_m - \overline{Lon}\right)\left(x_m - \overline{x}\right)}{\sum_{m=1}^{15}\left(x_m - \overline{x}\right)^2} \tag{8}$$

$$c = \overline{Lon} - b\overline{x} \tag{9}$$

$$Lon_{center} = dx_{center} + c \tag{10}$$

where a, b, c, and d are the coefficients of the linear equation between the geographic coordinates and the image pixel coordinates; Lat_m and Lon_m are the central geographic coordinates of the successfully matched building in the e-map; $\overline{Lat}$ and $\overline{Lon}$ are the mean values of Lat_m and Lon_m, respectively; x_m and y_m are the central pixel coordinates of the successfully matched buildings in the UAV image; $\overline{x}$ and $\overline{y}$ are the mean values of x_m and y_m, respectively; x_{center} and y_{center} are the central pixel coordinates of the UAV image; and Lat_{center} and Lon_{center} are the geographic coordinates corresponding to the center of the UAV image.

4. Data and Experiments

4.1. Instance Segmentation Dataset

Considering that images acquired during UAV flights exhibit a very high spatial resolution, we used Google Earth images with a spatial resolution of 0.3 m as the building instance segmentation dataset, and the sources included the publicly available BONAI dataset [37] and the self-constructed dataset for the Shijiazhuang area in China. The BONAI dataset contains 3300 images of 1024 pixels by 1024 pixels from six cities in China: Shanghai, Beijing, Harbin, Jinan, Chengdu, and Xi'an. As a supplement, 565 image samples from the Shijiazhuang area were used. The roofs of the buildings in all the images are labeled in detail and stored in Common Objects in Context (COCO) format (a commonly used sample set format proposed by Microsoft). A total of 3000 samples were randomly selected as the training set, 450 samples were selected as the validation set, and 415 samples were selected as the test set. Figure 3 shows examples of our dataset.

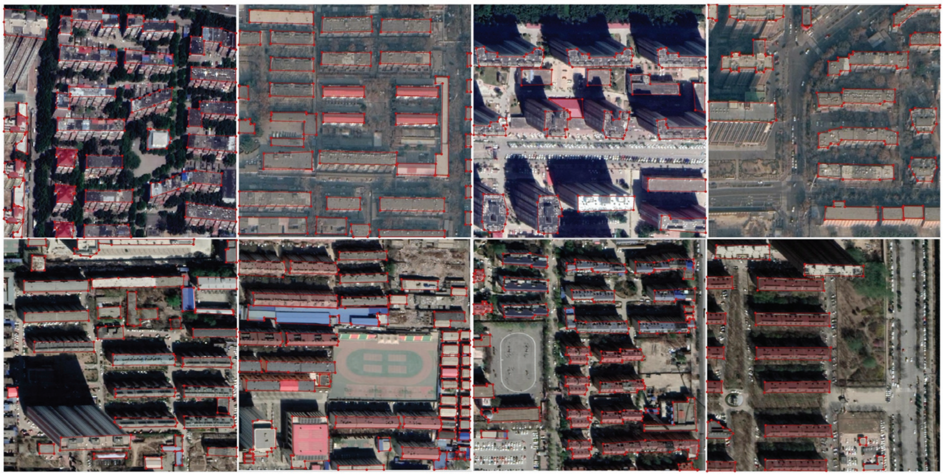


Figure 3. Examples of instance segmentation datasets. The red boxes are the labeled outlines of buildings.

4.2. Geolocalization Dataset

Although the spatial resolution of low altitude UAV images is usually at the centimeter level, image matching methods typically expect similar or identical resolutions between UAV images and geo-referenced images to ensure matching accuracy. Even though the spatial resolution of UAV images is particularly high, the SOLOv2 algorithm cannot guarantee accurate extraction of the outline of the building. In addition, a high spatial resolution also means a large computational volume and low computational efficiency. Therefore, considering the restrictions of UAV flights inside cities, we referred to the method of Nassar [10] to simulate UAV images of about 0.3 m resolution with Google Earth images. We selected Shijiazhuang, China, as the experimental area. We randomly plotted two routes in Google Earth. The first one occurs in the downtown area with an extremely high building density, and the second one occurs in the suburb area with a relatively low building density. A total of 17 points along the first route and 10 points along the second route were then randomly chosen as UAV image acquisition points. Images with a side length of 500 m, a spatial resolution of 0.235 m along the longitudinal direction, and a spatial resolution of 0.3 m along the latitudinal direction centered on these acquisition points were collected via Google Earth. These images exhibited a resolution of 2127 pixels by 1675 pixels. The latitude and longitude of the image acquisition points were recorded as the ground truth.

4.3. E-Map Dataset

The building outline vector data of Shijiazhuang, as shown in Figure 4, were obtained from Gaode Map (an electronic map navigation software of China). The vector data are based on the GCJ-02 coordinate system, which is a coordinate system obtained by offsetting the WGS84 coordinates. Therefore, the vector data were reprojected to the WGS84 geographic coordinate system via the open-source GIS software Geographic Resources Analysis Support System (GRASS 7.0.3). Moreover, GRASS was used to calculate geometric features, such as the area, aspect ratio, and orientation of the smallest bounding rectangle and the central latitude and longitude of each building. The vector data employed in this study covered an area of over 786 km^2 in the Shijiazhuang area, with a total data volume of 21.1 Mb.

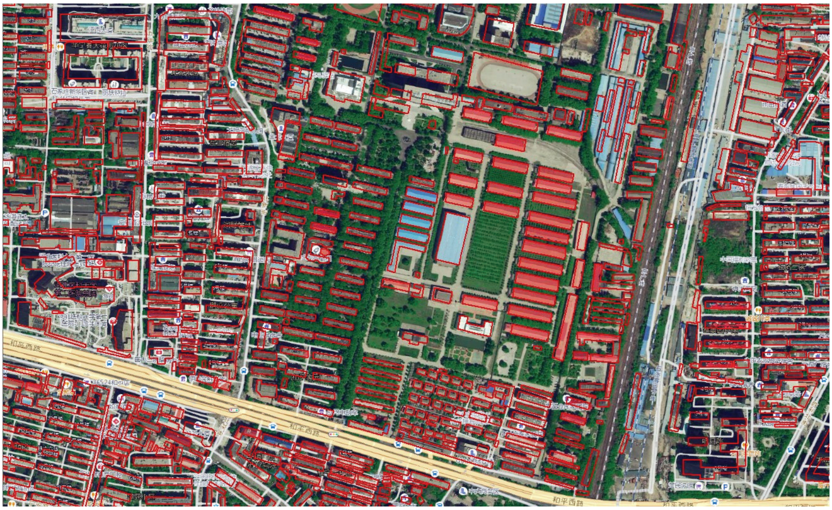


Figure 4. Electronic map vector data of a region in Shijiazhuang city from Gaode Map (an electronic map navigation software of China), superimposed on a satellite remote sensing image.

4.4. Comparison Experiments

The geo-referenced image used in the comparison experiments was a 1 m spatial resolution digital orthophoto image with geographic coordinate information. Notably, the image was refined by the surveying and mapping department. The image acquisition time was 2019, indicating that there were obvious color, imaging perspective, and scale differences between the geo-referenced image and the UAV images, as shown in Figure 5. Assuming that the UAV localization uncertainty is 500 m, an area of 1500 m × 1500 m completely covering the UAV image was randomly cropped within the geo-referenced image as the image patch to be matched. The digital orthophoto image used in the comparison experiments covered an area of 200 km^2, with a total data volume of 572 Mb.

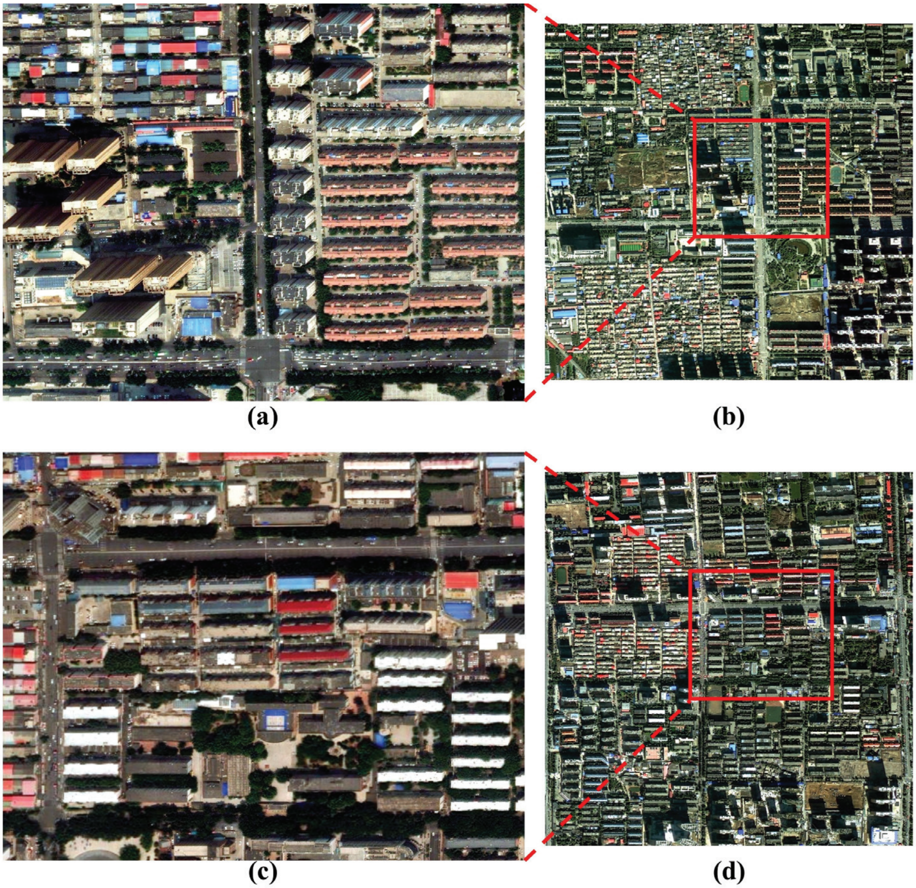


Figure 5. Comparison between the UAV and geo-referenced image in the comparison experiments. (**a**,**c**) show UAV images and (**b**,**d**) show subsets of the geo-referenced image, and the red boxes indicate the positions of the UAV images in the subset.

The RIFT feature matching algorithm is insensitive to nonlinear radiation distortions and has been validated for different types of multimodal image datasets, including optical–optical, infrared–optical and synthetic aperture radar (SAR)–optical datasets [13]. The CFOG method also exhibits better adaptability to image color differences and has been demonstrated as a better algorithm than the MI and HOPC methods [14,21]. Thus, we chose the RIFT and CFOG algorithms as comparison algorithms for the image-based method. Moreover, we chose the SSM map-based method [10] as the ablation comparison algorithm for the SSRM method.

In the RIFT algorithm, both corner and edge points in the PC map are extracted as feature points. Because the PC map is sensitive to the spatial resolution, we resampled the UAV image to the spatial resolution of the geo-referenced image. The log-Gabor convolution sequence was subsequently used to construct the MIM, and the MIM was used to describe feature points via a SIFT-like method. Finally, an exhaustive search method was employed to match the feature points, and the random sample consensus (RANSAC) algorithm was used to filter outliers [13].

The CFOG algorithm was originally used for matching remote sensing images, so it defaults to the presence of geographic projections and coordinates in the images, and the geometric error is relatively small. However, in our experiments, the UAV position uncertainty is 500 m, so we first resampled the UAV image to the spatial resolution of the geo-referenced image and then manually selected eight control point pairs to approximate the transformation relationship between the UAV and geo-referenced images. Afterward, 200 template windows of 140 × 140 pixels in the UAV image were selected to match the geo-referenced image, and the central pixels of the matched windows were regarded as the final matched points [14].

In the SSM algorithm, the geo-reference map is extracted from the e-map with a coverage area of 750 m × 750 m centered on the UAV position. The features of the buildings in the UAV images were matched with those in the geo-reference map one by one. The buildings with the highest sum of the area similarity, minimum bounding rectangle similarity, and orientation similarity were regarded as successfully matched buildings, and duplicate matching pairs were removed [10].

5. Results

5.1. Instance Segmentation Results

We conducted instance segmentation experiments on a PC with an NVIDIA Tesla V100 GPU and 32 GB of video RAM. Figure 6 shows the building extraction results for some of the UAV images. Buildings with clear outlines and larger areas were extracted well. SOLOv2 achieved a value of 67.4% for AP50, which is the average precision at an intersection over union (IoU) threshold of 0.5, and a value of 64.1% for the recall rate for the test set, which is relatively satisfactory. However, there were still a few cases where buildings with high similarity with the background, small areas, or no obvious differences between individuals (e.g., buildings in urban villages) were missed. Moreover, the outlines of some buildings were not very accurate.

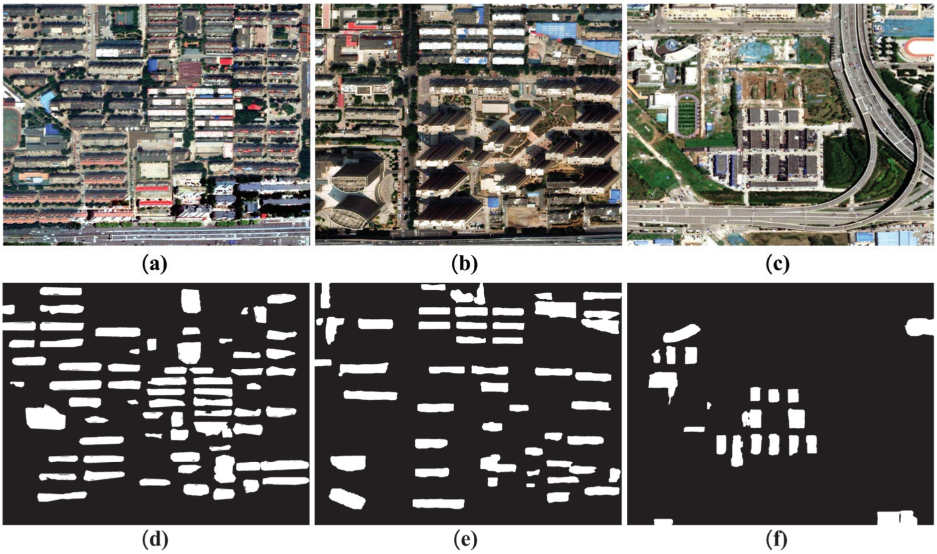


Figure 6. Building extraction results. (**a**–**c**) show UAV images and (**d**–**f**) show building mask maps. (**a**,**b**) show the downtown area and (**c**) shows the suburb area.

In terms of the processing efficiency, the average processing time for a single UAV image was 0.433 s, with an image interpretation time of 0.4 s and a thinning time of 0.033 s for converting building patches into outline information. Rapid extraction of individual buildings is highly beneficial for real-time UAV localization.

5.2. Scene Matching Results

The matching results of the RIFT algorithm are shown in Figures 7 and 8. Figure 7 shows the results with small matching errors. According to the matching results, although there were differences in the lighting and imaging angles between the UAV and geo-referenced images, the phases between the images remained consistent, so the matching effect was relatively satisfactory. Figure 8 shows the results with a poor match performance. Owing to the existence of high-rise buildings, the significant differences in lighting angles and imaging perspectives resulted in the phase deviating between the UAV and geo-referenced images, leading to a poor matching performance. Therefore, the RIFT algorithm is still greatly influenced by the scene.

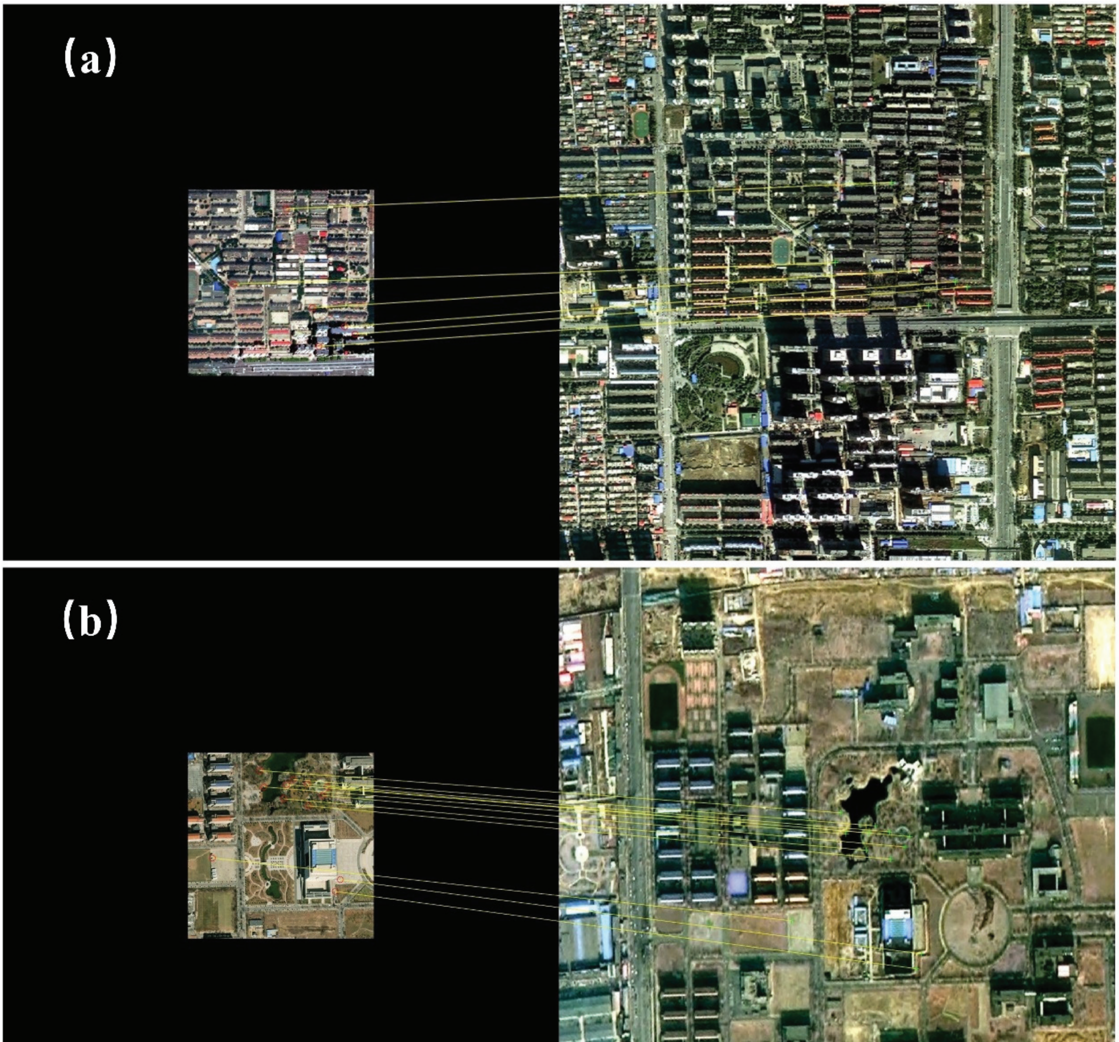


Figure 7. Good matching results of the RIFT algorithm. (**a**) shows the downtown area and (**b**) shows the suburb area.

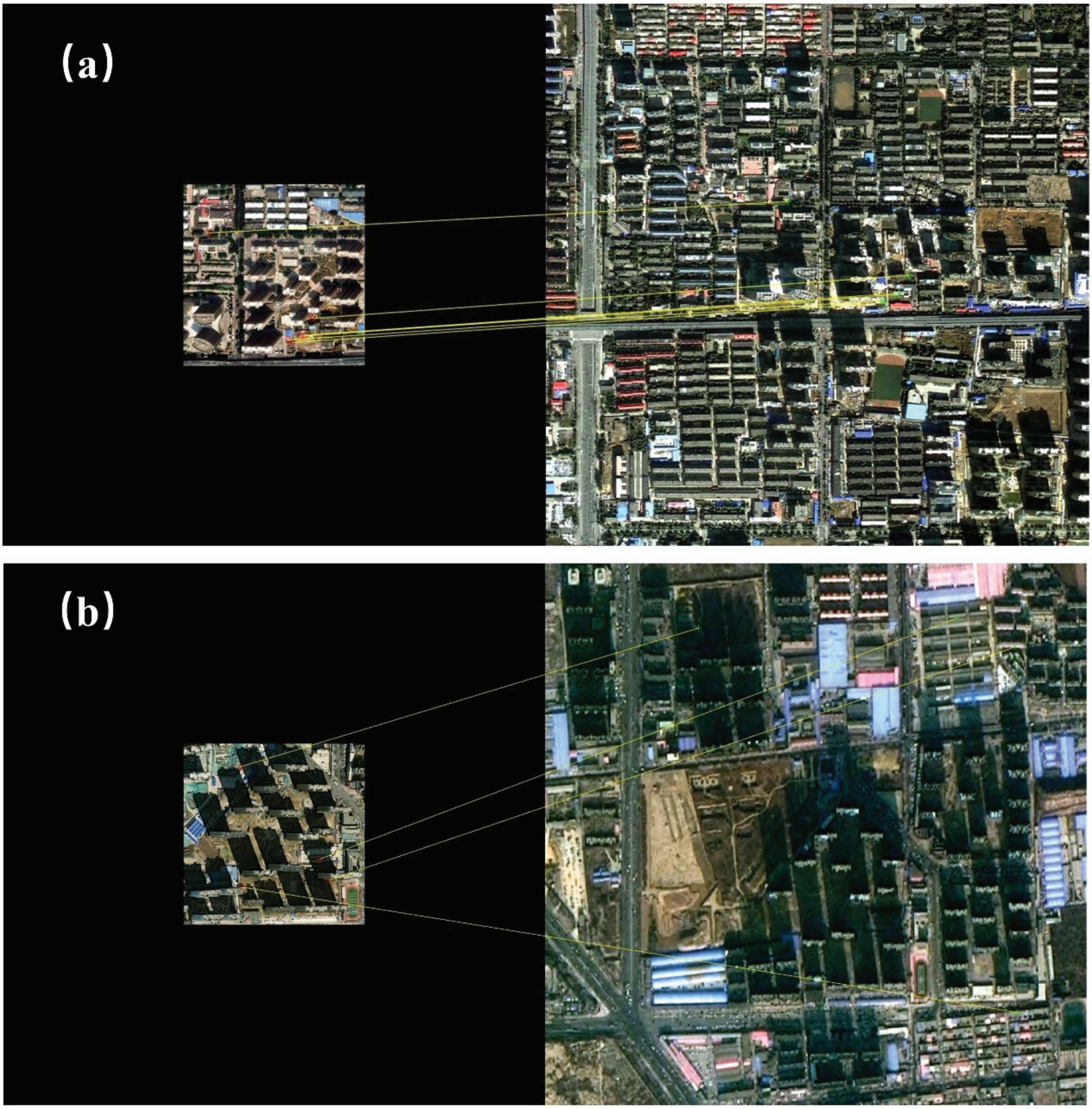


Figure 8. Poor matching results of the RIFT algorithm. (**a**) shows the downtown area and (**b**) shows the suburb area.

Figure 9 shows the matching results of the CFOG algorithm, and almost no feature points were matched correctly. This occurred because the assumption of phase consistency no longer holds when there are obvious lighting angle and imaging perspective differences between the UAV and geo-referenced images.

Figure 10 shows the matching results of the SSM method. Owing to the large number of similar buildings, many individual buildings were mismatched.

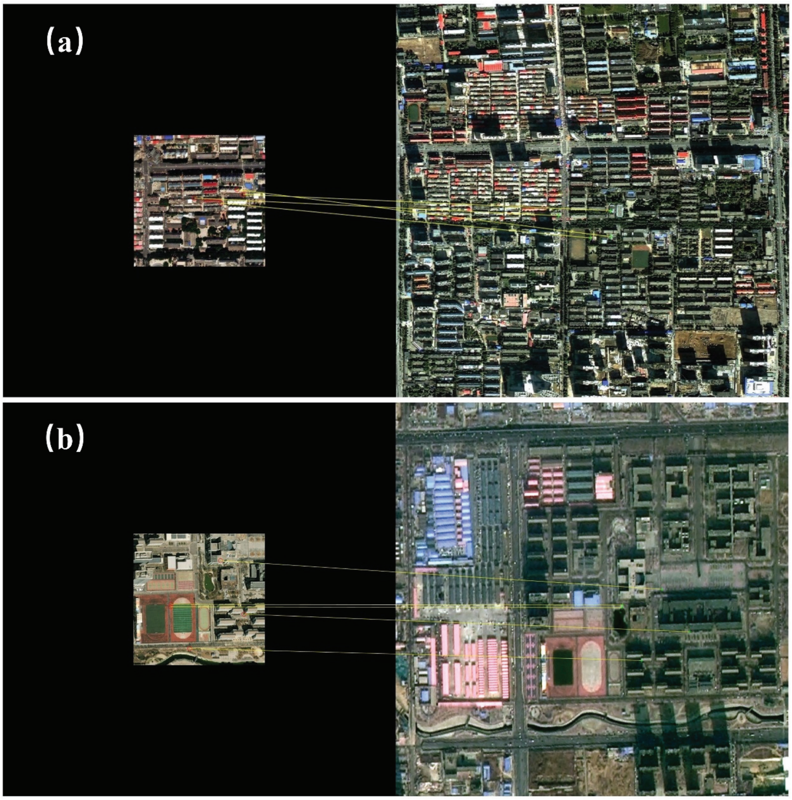


Figure 9. Matching results of the CFOG algorithm. (**a**) shows the downtown area and (**b**) shows the suburb area.

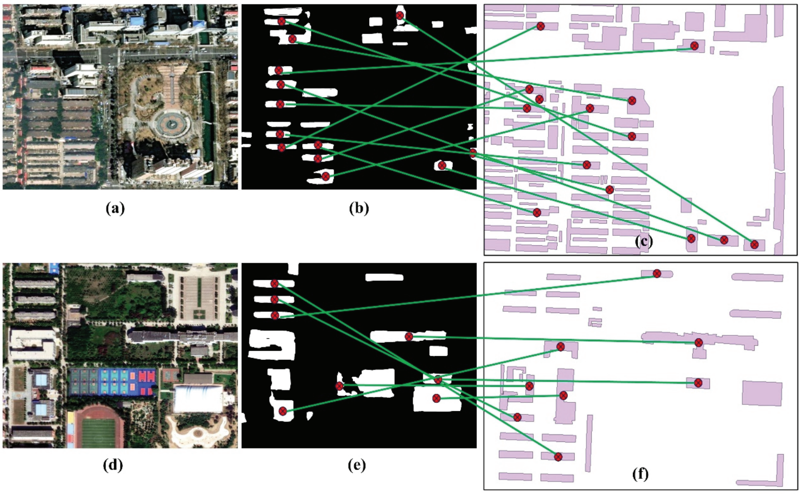


Figure 10. Matching results of the SSM method. (**a**,**d**) show the UAV images, (**b**,**e**) show the building extraction results, and (**c**,**f**) show the e-maps. (**a**) shows the downtown area and (**d**) shows the suburb area. The red points indicate the central points of the buildings, and the points connected by the green lines indicate the successfully matched buildings.

Figure 11 shows the matching results of the SSRM method for three scenes. A comparison of Figure 11a–c revealed that although some buildings in the UAV images were missed or demonstrated less accurate edges, the scene matching process filtered out those buildings and successfully matched those with good extraction results and clear contours. A comparison of Figure 11d–f revealed that, owing to sensor side-view imaging, the tops and bottoms of some tall buildings did not completely overlap, so the tops of buildings at different heights did not fully reflect the true spatial relationships among them. However, the scene matching process could facilitate the elimination of this inconsistency, and only buildings with consistent spatial relationships were matched with the e-map. According to the comparison of Figure 11g–i, owing to the phase difference, there were also inconsistencies in the e-map compared with the UAV images, such as redundant buildings, but the scene matching process was resistant to these e-map inaccuracies and could achieve correct scene matching.

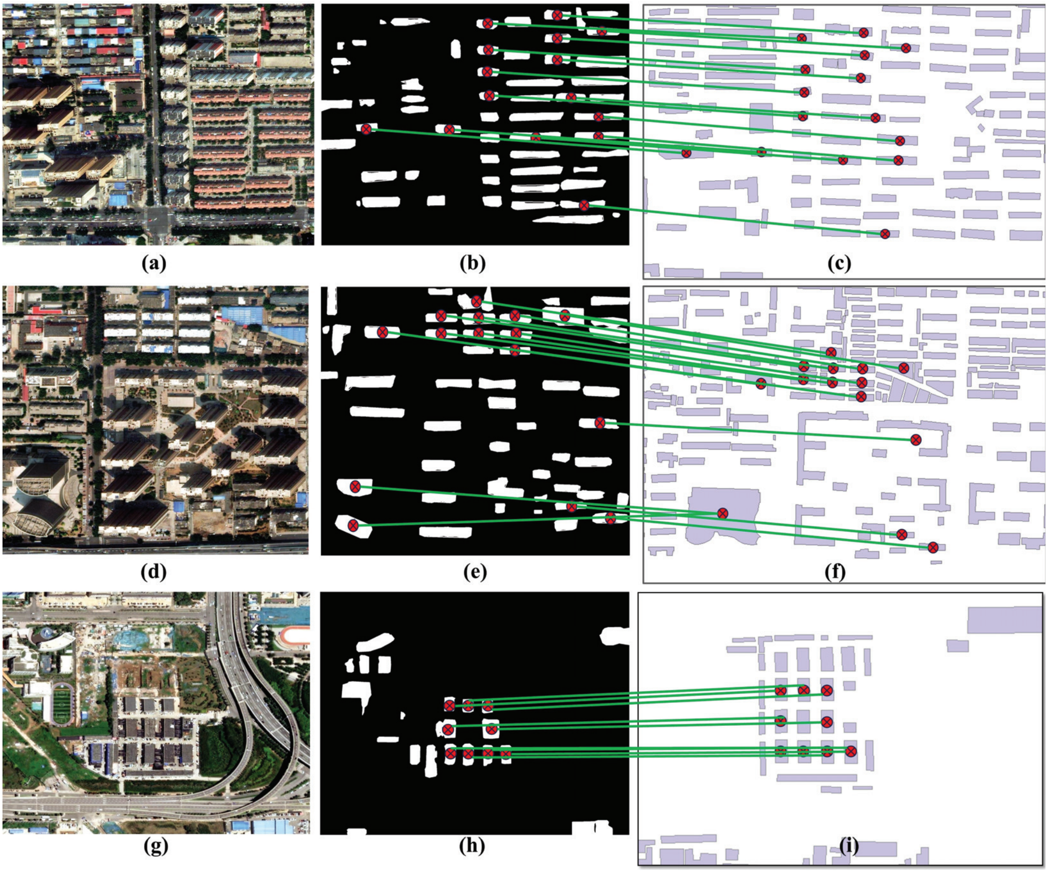


Figure 11. Scene matching results of the SSRM method. (**a**,**d**,**g**) UAV images; (**b**,**e**,**h**) building extraction results from the UAV images; and (**c**,**f**,**i**) e-map data. (**a**,**d**) Downtown area and (**g**) suburb area. The red points indicate the central points of the buildings, and the points connected by the green lines indicate the successfully matched buildings.

5.3. UAV Localization Results

The overall localization performance of each algorithm is detailed in Table 1. The average UAV localization errors of the RIFT algorithm were 38.46 and 250.11 m, with root mean square error (RMSE) values of 105.82 and 196.77 m, for the downtown and suburb datasets, respectively. The average UAV localization errors of the CFOG algorithm were 49.59 and 59.85 m, with RMSE values of 21.45 and 35.08 m, for the downtown and suburb datasets, respectively. The localization results are shown in Figures 12 and 13, and the performance differences among UAV images of different methods are shown in Figure 14. According to Figures 7 and 8, and Table 1, although the RIFT algorithm performed well in matching certain images, its average localization error was still relatively large, indicating that it does not completely overcome the matching difficulties caused by lighting and imaging perspective differences. In addition, the localization error of the RIFT algorithm for the downtown dataset was much smaller than that for the suburb dataset. This occurred because in the suburb dataset, there were significant changes in the land cover type, as well as significant lighting angle and brightness differences between some UAV and geo-referenced images, resulting in significant phase consistency differences between these images and leading to poor matching results. Moreover, the RMSE of the RIFT algorithm was relatively large, indicating that this algorithm is not robust enough and is greatly affected by image scenes. Although the CFOG algorithm could rarely match the feature points correctly, the manually selected control points limited the scope of the template matching process. Therefore, although the average localization accuracy of the CFOG algorithm was low, the localization error and RMSE did not differ much between the urban and suburb datasets.

Table 1. UAV localization results.

Method	Error/m		RMSE/m		Time/s
	Downtown	Suburb	Downtown	Suburb	
RIFT	38.46	250.11	105.82	196.77	18.65
CFOG	49.59	59.85	21.45	35.08	2.01
SSM	43.74	96.44	18.12	35.81	0.748
SSRM	7.38	11.92	4.12	7.57	3.58

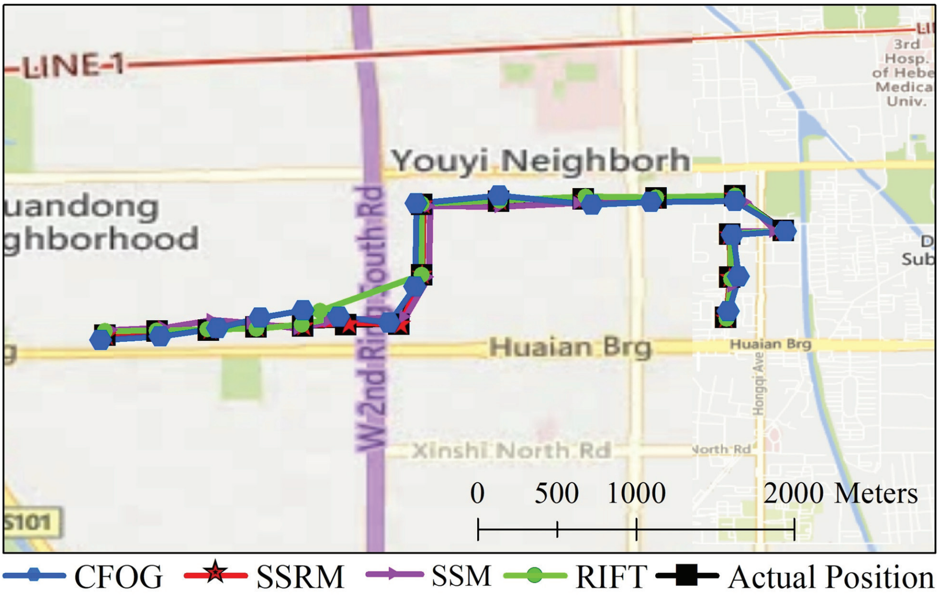


Figure 12. UAV localization results for the downtown dataset. The black line indicates the real UAV flight path, and the red, green, blue, and purple lines indicate the UAV flight paths estimated by SSRM, RIFT, CFOG, and SSM methods, respectively.

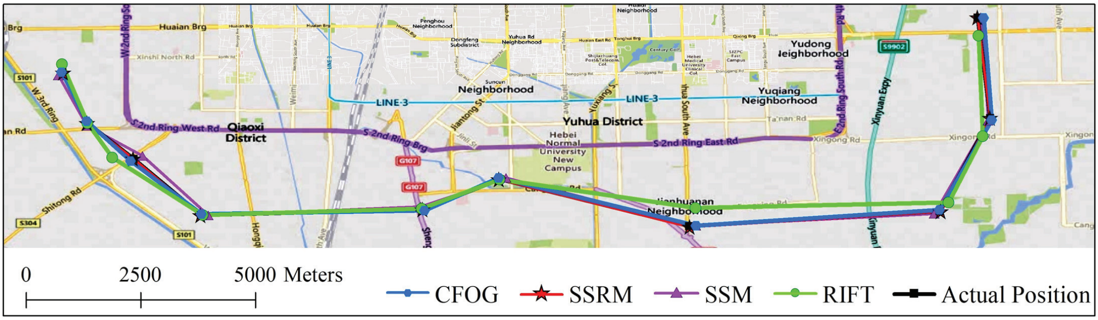


Figure 13. UAV localization results for the suburb dataset. The black line indicates the real UAV flight path, and the red, green, blue, and purple lines indicate the UAV flight paths estimated by the SSRM, RIFT, CFOG, and SSM methods, respectively.

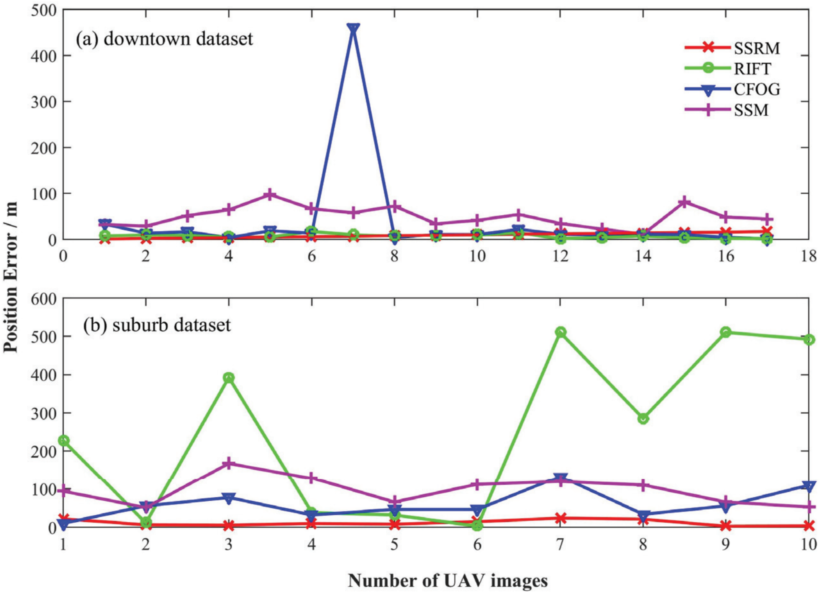


Figure 14. Comparison of the UAV localization errors of the different algorithms.

The SSM method achieved localization errors of 43.74 and 96.44 m, with RMSEs of 18.12 and 35.81 m, for the downtown and suburb datasets, respectively (Figures 12 and 13, respectively). Sparse buildings resulted in poorer performance for the SSM algorithm for the suburb dataset. The shape and orientation features of buildings constrained the range of feature matching. However, multiple similar buildings in the urban scenes still limited the matching accuracy.

The SSRM method achieved average localization errors of 7.38 and 11.92 m, with RMSEs of 1.12 and 7.57 m, for the downtown and suburb datasets, respectively, as shown in Figures 12 and 13, respectively. This demonstrated that the SSRM method significantly reduced the localization error compared with the SSM method, and the denser the buildings are, the higher the localization accuracy. In addition, the SSRM method is robust enough for the different scenes. The SSRM method completely avoids the matching problems caused by land cover change, image illumination, and imaging perspective differences between the UAV and geo-referenced images.

The CFOG algorithm is characterized by high computational efficiency, and its average matching time is 2.01 s. Owing to the complexity of feature point extraction and description, the RIFT algorithm achieves high computational complexity, and its average matching time is 18.65 s (including UAV image resampling). The average time of the SSM algorithm is 0.748 s, with the extraction of buildings from the UAV images requiring 0.4 s and the matching process lasting 0.3 s. The average time of the SSRM method is 3.58 s, of which the average time for building extraction from the UAV images is 0.40 s, the thinning operation time is 0.03 s, and the average time for scene matching based on SSRM is 3.15 s. The computational efficiency of the SSRM method is lower than that of the CFOG and SSM algorithms but significantly greater than that of the RIFT algorithm.

6. Discussion

The building layer of the Gaode map originates from the surveying department and exhibits a small geometric error. Although some buildings in the map may not be consistent with the UAV images due to rapid urban renewal, it does not affect the scene matching process and does not result in UAV localization errors.

The factor that influences the accuracy of the SSRM method is the accuracy of the control points, i.e., the accuracy of the central position of the successfully matched building. The central position of the building outlines is determined by the building extraction process. The deep learning-based building extraction method mainly leads to omission and commission errors and inaccurate outlines. Some buildings with similar features relative to the surrounding background, such as roads, could be missed. The omission of a building will result in it not being used as a ground control point, which does not affect the UAV localization accuracy when there are sufficient control points. Ground targets with similar features to those of buildings may be detected incorrectly. Misdetected buildings exhibit fewer matches in the scene matching process and are not regarded as control points. Thus, they do not affect the UAV localization accuracy.

The building outlines in the e-map represent the bottoms of the buildings, whereas the building outlines extracted from the UAV images represent the building roofs. Because there is an imaging angle in the UAV images, the roof outlines of the buildings do not coincide with the bottom outlines, as shown in Figure 6b, which leads to errors in the control points. For a fixed imaging angle, the higher the building is, the greater the position deviation between the top and bottom outlines in the UAV images and the larger the control point error. When the top outline does not overlap the bottom outline completely, the building is not considered to calculate the control points.

7. Conclusions

In this article, a novel UAV autonomous visual localization method, namely, the SSRM method, is proposed for urban scenes as an alternative to traditional methods based on image matching. The SSRM method first extracts individual buildings from the UAV images, and then uses the shape and spatial relationships of the buildings to match the UAV images with the vector e-map. According to the matching results, control points are generated using the center of the matched buildings, and they are applied to determine the UAV position. The SSRM method can address the impact of lighting, scale, and imaging perspective differences in the image matching process. The SSRM algorithm achieves a smaller localization error in areas with denser buildings. Moreover, the SSRM method requires much less data to be prestored on UAVs than does the image matching method, which is more advantageous in large-area scenarios.

To illustrate the effectiveness of the SSRM method, we apply it to simulated UAV images from Google Earth. In addition, the RIFT, CFOG, and SSM algorithms are utilized for absolute visual localization of UAVs in comparison experiments. The results show that the SSRM method better realizes scene localization and UAV position determination. The obvious difference in performance between the downtown and suburb datasets shows that the RIFT algorithm is not robust enough and is greatly affected by image scenes.

The RIFT algorithm may still fail when there are obvious land cover changes or lighting angle and brightness differences between the UAV and geo-referenced images, because the assumption of phase consistency may not hold. The CFOG algorithm still cannot overcome the problems caused by lighting and imaging differences. The SSM algorithm, although extremely efficient computationally, results in incorrect matching due to the high similarity of buildings, further yielding a low localization performance. The results show that the SSRM method achieves better localization accuracy and is more robust.

Considering that the shape and spatial relationships of buildings are used for scene matching in the SSRM method, it may fail when dense buildings do not occur in UAV images for a long period. In addition, the SSRM method requires known orientation and scale information of the UAV images, and a possible future research direction is a scene matching method given an unknown orientation and scale of the UAV images.

Author Contributions: Conceptualization, Y.L.; methodology, Y.L. and F.S.; comparison experiments, Y.L. and F.S.; writing—original draft preparation, Y.L.; writing—review and editing, J.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the National Natural Science Foundation of China under grants U22B2011 and 62276206, in part by the Aeronautical Science Foundation of China under grant 2023Z071081001.

Data Availability Statement: The original data presented in the study are openly available at https://pan.baidu.com/s/137N9YXoqbWQJ1VLsxPRm6g?pwd=wpaj (permanent).

Acknowledgments: The SOLOv2 instance segmentation datasets were supported by Zhengqiang Guo.

Conflicts of Interest: Authors Yu Liu and Fangde Sun were employed by the company CETC Key Laboratory of Aerospace Information Applications. The remaining author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interest.

References

1. Couturier, A.; Akhloufi, M.A. A review on absolute visual localization for UAV. *Robot. Auton. Syst.* **2021**, *135*, 103666.
2. Kinnari, J.; Verdoja, F.; Kyrki, V. GNSS-Denied Geolocalization of UAVs by Visual Matching of Onboard Camera Images with Orthophotos. In Proceedings of the 20th International Conference on Advanced Robotics, ICAR 2021, Manhattan, NY, USA, 6–10 December 2021; pp. 555–562.
3. Gui, J.; Gu, D.; Wang, S.; Hu, H. A review of visual inertial odometry from filtering and optimisation perspectives. *Adv. Robot.* **2015**, *29*, 1289–1301.
4. Liu, Y.; Bai, J.; Wang, G.; Wu, X.; Sun, F.; Guo, Z.; Geng, H. UAV Localization in Low-Altitude GNSS-Denied Environments Based on POI and Store Signage Text Matching in UAV Images. *Drones* **2023**, *7*, 451. [CrossRef]
5. Li, W. Research on UAV Localization Method based on Image Registration and System Design. Master Degree, Univeristy of Electronic Science and Technology, Chengdu, China, 2023.
6. Boiteau, S.; Vanegas, F.; Gonzalez, F. Framework for Autonomous UAV Navigation and Target Detection in Global-Navigation-Satellite-System-Denied and Visually Degraded Environments. *Remote Sens.* **2024**, *16*, 471. [CrossRef]
7. Zhao, C.; Zhou, Y.; Lin, Z.; Hu, J.; Pan, Q. Review of scene matching visual navigation for unmanned aerial vehicles. *Sci. Sin. Inf.* **2019**, *49*, 507–519.
8. Chen, J.; Chen, C.; Chen, Y. Fast algorithm for robust template matching with M-estimators. *IEEE Trans. Signal Process.* **2003**, *51*, 230–243.
9. Xu, Y.; Pan, L.; Du, C.; Li, J.; Jing, N.; Wu, J. Vision-based UAVs Aerial Image Localization: A Survey. In Proceedings of the 2nd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery, Seattle, WA, USA, 6 November 2018; pp. 8–18.
10. Nassar, A.; Amer, K.; ElHakim, R.; ElHelw, M. A Deep CNN-Based Framework For Enhanced Aerial Imagery Registration with Applications to UAV Geolocalization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1594–1604.
11. Choi, J.; Myung, H. BRM Localization: UAV localization in GNSS-denied environments based on matching of numerical map and UAV images. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 4537–4544.
12. Buck, A.R.; Keller, J.M.; Skubic, M. A modified genetic algorithm for matching building sets with the histograms of forces. In Proceedings of the IEEE Congress on Evolutionary Computation, Brisbane, Australia, 10–15 June 2012; pp. 1–7.
13. Li, J.; Hu, Q.; Ai, M. RIFT: Multi-Modal Image Matching Based on Radiation-Variation Insensitive Feature Transform. *IEEE Trans. Image Process.* **2020**, *29*, 3296–3310.

14. Ye, Y.; Bruzzone, L.; Shan, J.; Bovolo, F.; Zhu, Q. Fast and Robust Matching for Multimodal Remote Sensing Image Registration. *IEEE Trans. Geosci. Remote* **2019**, *57*, 9059–9070.
15. Zhang, X.; Leng, C.; Hong, Y.; Pei, Z.; Cheng, I.; Basu, A. Multimodal Remote Sensing Image Registration Methods and Advancements: A Survey. *Remote Sens.* **2021**, *13*, 5128. [CrossRef]
16. Lewis, J.P. Fast Template Matching. In *Vision Interface 95; Canadian Image Processing and Pattern Recognition Society, Quebec, Canada, 15–19 May 1995*; Canadian Image Processing and Pattern Recognition Society: Toronto, ON, Canada; Volume 95, pp. 120–123.
17. Wan, X.; Liu, J.; Yan, H.; Morgan, G.L.K. Illumination-invariant image matching for autonomous UAV localisation based on optical sensing. *ISPRS J. Photogramm. Remote Sens.* **2016**, *119*, 198–213.
18. Patel, B.; Barfoot, T.D.; Schoellig, A.P. Visual Localization with Google Earth Images for Robust Global Pose Estimation of UAVs. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 6491–6497.
19. Sibiryakov, A. Fast and high-performance template matching method. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011; pp. 1417–1424.
20. Ye, Y.; Shan, J.; Hao, S.; Bruzzone, L.; Qin, Y. A local phase based invariant feature for remote sensing image matching. *ISPRS J. Photogramm. Remote Sens.* **2018**, *142*, 205–221.
21. Fan, Z.; Zhang, L.; Liu, Y.; Wang, Q.; Zlatanova, S. Exploiting High Geopositioning Accuracy of SAR Data to Obtain Accurate Geometric Orientation of Optical Satellite Images. *Remote Sens.* **2021**, *13*, 3535. [CrossRef]
22. Haigang, S.; Chang, L.; Zhe, G.; Zhengjie, J.; Chuan, X. Overview of multi-modal remote sensing image matching methods. *Acta Geod. Cartogr. Sin.* **2022**, *51*, 1848–1861.
23. Mantelli, M.; Pittol, D.; Neuland, R.; Ribacki, A.; Maffei, R.; Jorge, V.; Prestes, E.; Kolberg, M. A novel measurement model based on abBRIEF for global localization of a UAV over satellite images. *Robot. Auton. Syst.* **2019**, *112*, 304–319.
24. Dusmanu, M.; Rocco, I.; Pajdla, T.; Pollefeys, M.; Sivic, J.; Torii, A.; Sattler, T. D2-Net: A Trainable CNN for Joint Detection and Description of Local Features. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 8084–8093.
25. Kumar, B.V.; Carneiro, G.; Reid, I. Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 5385–5394.
26. DeTone, D.; Malisiewicz, T.; Rabinovich, A. SuperPoint: Self-Supervised Interest Point Detection and Description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 224–236.
27. Sarlin, P.; DeTone, D.; Malisiewicz, T.; Rabinovich, A. SuperGlue: Learning Feature Matching with Graph Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 4937–4946.
28. Sun, J.; Shen, Z.; Wang, Y.; Bao, H.; Zhou, X. LoFTR: Detector-Free Local Feature Matching with Transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 8918–8927.
29. Masselli, A.; Hanten, R.; Zell, A. Localization of Unmanned Aerial Vehicles Using Terrain Classification from Aerial Images. *Intell. Auton. Syst.* **2016**, *13*, 831–842.
30. Yun, H.; Ziyang, M.; Jiawen, A.; Yuanqing, W. Localization method by aerial image matching in urban environment based on semantic segmentation. *J. Huazhong Univerisity Sci. Technol. (Nat. Sci. Ed.)* **2022**, *11*, 79–84.
31. Wang, H.; Cheng, Y.; Liu, N.; Zhao, Y.; Chan, H.C. An illumination-invariant shadow-based scene matching navigation approach in low-altitude flight. *Remote Sens.* **2022**, *14*, 3869. [CrossRef]
32. Shan, M.; Wang, F.; Lin, F.; Gao, Z.; Tang, Y.Z.; Chen, B.M. Google map aided visual navigation for UAVs in GPS-denied environment. In Proceedings of the IEEE Conference on Robotics and Biomimetics, Zhuhai, China, 6–9 December 2015; pp. 114–119.
33. Yol, A.; Delabarre, B.; Dame, A.; Dartois, J.; Marchand, E. Vision-based absolute localization for unmanned aerial vehicles. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Chicago, IL, USA, 14–18 September 2014; pp. 3429–3434.
34. Sun, L.; Sun, Y.; Yuan, S.; Ai, M. A survey of instance segmentation research based on deep learning. *CAAI Trans. Intell. Syst.* **2022**, *17*, 16.
35. Wang, X.; Zhang, R.; Kong, T.; Li, L.; Shen, C. SOLOv2: Dynamic and Fast Instance Segmentation. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 17721–17732.
36. Douglas, D.H.; Peucher, T.K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Int. J. Geogr. Inf. Sci.* **1973**, *2*, 112–122.
37. Wang, J.; Meng, L.; Li, W.; Yang, W.; Yu, L.; Xia, G.S. Learning to Extract Building Footprints From Off-Nadir Aerial Images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 1294–1301. [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

Closed-Form Method for Unified Far-Field and Near-Field Localization Based on TDOA and FDOA Measurements

Weishuang Gong [1], Xuan Song [2,*], Chunyu Zhu [1], Qi Wang [1] and Yachao Li [3]

[1] Hangzhou Institute of Technology, Xidian University, Hangzhou 311231, China; wsgong@stu.xidian.edu.cn (W.G.); zhuchunyu@xidian.edu.cn (C.Z.); wangqi@xidian.edu.cn (Q.W.)
[2] School of Astronautics, Northwestern Polytechnical University, Xi'an 710072, China
[3] National Key Laboratory of Radar Signal Processing, Xidian University, Xi'an 710071, China; ycli@mail.xidian.edu.cn
* Correspondence: xuansong95@nwpu.edu.cn

Abstract: When the near-field and far-field information of a target is uncertain, it is necessary to choose a suitable localization method. The modified polar representation (MPR) method integrates the two scenarios and achieves a unified localization with direction of arrival (DOA) estimation in the far field and position estimation in the near field. Previous studies have only proposed solutions for stationary environments and have not considered the motion factor. Therefore, this paper proposes a new unified positioning algorithm using multi-sensor time difference of arrival (TDOA) and frequency difference of arrival (FDOA) measurements without prior target source information. The method represents the position of the target source using MPR and describes the localization problem as a weighted least squares (WLS) problem with two constraints. We first obtain the initial estimates by WLS without considering the constraints and then investigate a two-step error correction method based on the constraints. The first step corrects the initial estimate using the Taylor series expansion technique, and the second step corrects the DOA estimate in the previous step using the direct error compensation technique based on the properties of the second constraint. Simulation experiments show that the method is effective for the unified positioning of moving targets and can achieve the Cramer–Rao lower bound (CRLB).

Keywords: passive localization; unified model; TDOA; FDOA; weighted least squares

Citation: Gong, W.; Song, X.; Zhu, C.; Wang, Q.; Li, Y. Closed-Form Method for Unified Far-Field and Near-Field Localization Based on TDOA and FDOA Measurements. *Remote Sens.* **2024**, *16*, 3047. https://doi.org/10.3390/rs16163047

Academic Editor: Andrzej Stateczny

Received: 22 June 2024
Revised: 3 August 2024
Accepted: 14 August 2024
Published: 19 August 2024

1. Introduction

Passive sensor positioning technology does not actively emit signals and is characterized by strong survivability and long-range capabilities. It has been widely used in sensor networks, radar, sonar, navigation, wireless communications, and other fields [1–5]. When the sensor and the target source remain stationary, a commonly used positioning method is to use the relevant parameters received by the sensor network, such as the time of arrival (TOA) [6–8], time difference of arrival (TDOA) [9–12], angle of arrival (AOA) [13–15], and received signal strength (RSS) [16,17], to establish a system of nonlinear measurement equations to estimate the source position. When there are motion characteristics of the sensor or target, a common positioning system is the combination of multiple types of measurements from sensors, such as the TDOA and frequency difference of arrival (FDOA) [18–21], to obtain the position and velocity of the target at the same time.

The premise for estimating the target position based on sensor measurements is to assume that the target is not too far away from the sensor [22], as the signal wavefront is curved in this condition, which helps determine the unique source location. However, the wavefront of the signal is approximately planar under far-field conditions, the target position cannot be derived, and only the direction of arrival (DOA) estimation can be performed. In actual scenarios, the distance of the target source location is usually unknown. If an inappropriate positioning algorithm is selected, the positioning performance will decrease.

To resolve this issue, Ref. [23] initially studied the unified far-field and near-field localization model based on the TDOA. They proposed to use the modified polar representation (MPR) method to represent the location by decoupling the target position into the DOA and inverse distance. Subsequently, the localization problem is transformed into a maximum likelihood estimation (MLE) problem and solved using a Gaussian Newton iterative method, where the initial values of the iterations are provided by the semidefinite relaxation (SDR) method. The method achieves a unified positioning with long-range direction finding and close-range positioning, where the position of the source is obtained by inverse-range recovery. However, the iterative method requires a proper initial value, otherwise divergence may occur. To eliminate the convergence problem, Ref. [22] proposed two closed-form solution methods, SUM and GTRS, in the framework of the MPR model, which can estimate the target parameters directly without initial values and achieve optimal localization performance. On this basis, Ref. [24] proposed the SCO-MPR method, which further reduced the estimation bias. In order to improve the robustness in strong noise environments, Ref. [25] proposed two SDR methods to estimate the source parameters, and both of their estimation performances can be close to the CRLB. Ref. [26] proposed unified localization models based on the AOA and hybrid AOA-TDOA, which are iteratively solved using the initial values of the SDR method. In contrast, Ref. [27,28] proposed AOA-based closed-form solution algorithms and SDR algorithms, respectively, which are not iterative solution algorithms, avoiding the divergence risk in [26]. In addition, considering the difficulty of determining the propagation velocity of a signal in an unknown medium, Ref. [29] proposed an SDR method with high noise immunity and an algebraic solution method with low computational degree, which realizes the unified localization of the unknown signal velocity. Although the above studies achieved significant results in the case of a stationary target, they did not consider the case of a moving target, and motion between the sensor and the target source is common in real application scenarios. For instance, the strategic placement of monitoring stations along the coastline is essential for detecting sea vessels and aerial crafts. Should we fail to anticipate their subsequent actions through the analysis of their velocities, we might miss the preemptive opportunity, potentially exposing us to grave dangers.

This paper investigates uniform localization without prior information about the target for moving target scenarios. Based on available research results, we continue to use the MPR model to express the target position and propose a new algebraic solution approach. The method converts the nonlinear localization equation into a linear equation by introducing auxiliary variables. Then, we obtain a WLS problem with two constraints by analyzing the nonlinear relationship between the elements of the auxiliary vector. Among them, the second constraint only relates to the source DOA information. We first ignore the constraints to $\mathbf{x}^\circ$ obtain the initial estimation by the WLS algorithm and use the constraints to build new equations to improve the initial estimation to obtain the exact solution. Finally, simulation experiments show that the proposed algorithm is effective.

We use bold lowercase letters for vectors and bold uppercase letters for matrices: is the true value of $\mathbf{x}$; $||\mathbf{x}||$ is the l_2 norms of $\mathbf{x}$; $\mathbf{x}(i)$ is the ith element of $\mathbf{x}$; $\mathbf{x}(i:j)$ is the vector consisting of the ith to jth elements of $\mathbf{x}$; $\mathbf{X}(i_1:i_2, j_1:j_2)$ is the matrix consisting of row i_1 to row i_2 and column j_1 to column j_2 of $\mathbf{X}$; $\mathbf{0}_N$ and $\mathbf{1}_N$ are column vectors of 0 and 1 of length N; $\mathbf{O}_{M\times N}$ denotes the $M \times N$ matrix of zero; $\mathbf{I}_N$ is the identity matrix of size N; $diag(x_1, x_2)$ denotes the diagonal matrix with x_1 and x_2 on the diagonal; and $\odot$ denotes the Hadamard product.

2. Measurement Model

We choose three-dimensional space as the localization scene. As shown in Figure 1, the position of the sensor is $\mathbf{m}_i = [x_i^\circ, y_i^\circ, z_i^\circ]^T$ and the speed is $\dot{\mathbf{m}}_i = [\dot{x}_i^\circ, \dot{y}_i^\circ, \dot{z}_i^\circ]^T$, the position of the target is $\mathbf{p}^\circ = [x^\circ, y^\circ, z^\circ]^T$ and the speed is $\dot{\mathbf{p}}^\circ = [\dot{x}^\circ, \dot{y}^\circ, \dot{z}^\circ]^T$, and d_i° is the range between the target source and the sensor, where $i = 1, \cdots, M$. We set $\mathbf{m}_1$ as the reference sensor.

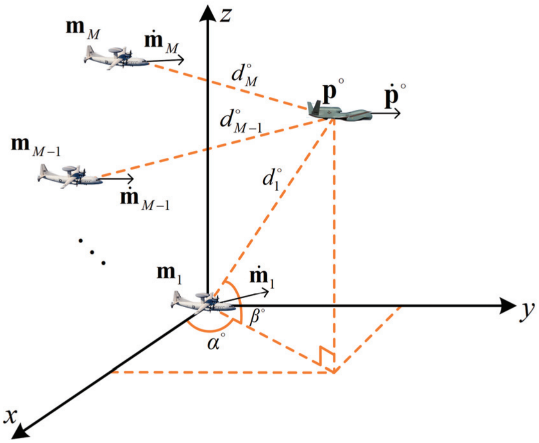


Figure 1. The 3D localization scenarios.

Next, we derive the equations that relate the target parameters to the TDOA and FDOA. From the TDOA measurements, we obtain

$$d_{i1}^\circ = d_i^\circ - d_1^\circ, \tag{1}$$

where d_{i1}° is the range difference (RD), which is the product of the TDOA and the signal propagation velocity, and d_i° can be expressed using the coordinates of the target as

$$d_i^\circ = \|\mathbf{p}^\circ - \mathbf{m}_i\| = \sqrt{(\mathbf{p}^\circ - \mathbf{m}_i)^T(\mathbf{p}^\circ - \mathbf{m}_i)} \tag{2}$$

To obtain the FDOA measurement formula, we take the derivative of both sides of (1) with respect to time, and the outcome is

$$\dot{d}_{i1}^\circ = \dot{d}_i^\circ - \dot{d}_1^\circ, \tag{3}$$

where $\dot{d}_{i1}^\circ$ is the range difference rate (RDR), which can be calculated from FDOA measurements [18], and $\dot{d}_i^\circ$ denotes the range rate, which is derived from Equation (2) for time:

$$d_i^\circ = \frac{(\dot{\mathbf{p}}^\circ - \dot{\mathbf{m}}_i)^T(\mathbf{p}^\circ - \mathbf{m}_i)}{d_i^\circ}. \tag{4}$$

Considering the error due to noise in the real situation, we make the measurement value the following:

$$[\mathbf{d}^T, \dot{\mathbf{d}}^T]^T = [\mathbf{d}^{\circ T}, \dot{\mathbf{d}}^{\circ T}]^T + [\boldsymbol{\eta}^T, \dot{\boldsymbol{\eta}}^T]^T, \tag{5}$$

where $\mathbf{d} = [d_{21}, d_{31}, \cdots, d_{M1}]^T$ and $\dot{\mathbf{d}} = [\dot{d}_{21}, \dot{d}_{31}, \cdots, \dot{d}_{M1}]^T$. In addition, $\mathbf{d}^\circ = [d_{21}^\circ, d_{31}^\circ, \cdots, d_{M1}^\circ]^T$ is the true RD vector, $\dot{\mathbf{d}}^\circ = [\dot{d}_{21}^\circ, \dot{d}_{31}^\circ, \cdots, \dot{d}_{M1}^\circ]^T$ is the true RDR vector, and $\boldsymbol{\eta} = [\eta_{21}, \eta_{31}, \cdots, \eta_{M1}]^T$ and $\dot{\boldsymbol{\eta}} = [\dot{\eta}_{21}, \dot{\eta}_{31}, \cdots, \dot{\eta}_{M1}]^T$ are the measurement noise vectors.

We assume that $\boldsymbol{\eta}$ and $\dot{\boldsymbol{\eta}}$ obey a Gaussian distribution with zero mean and $\mathrm{cov}([\dot{\boldsymbol{\eta}}^T, \dot{\boldsymbol{\eta}}^T]^T) = \begin{bmatrix} \mathbf{Q}_t & \mathbf{O}_{(M-1)\times(M-1)} \\ \mathbf{O}_{(M-1)\times(M-1)} & \mathbf{Q}_f \end{bmatrix} = \mathbf{Q}$, where $\mathbf{Q}_t$ is the covariance matrix of $\boldsymbol{\eta}$, and $\mathbf{Q}_f$ is the covariance matrix of $\dot{\boldsymbol{\eta}}$.

3. The Proposed Location Method

Since the source range is unknown, inappropriate algorithms may be selected and result in suboptimal positioning performance. To solve this problem, referring to [23], this paper represents the position of the source as MPR coordinates to obtain a unified localization model. We introduce the velocity vector $\dot{\mathbf{p}}^{\circ}$ and denote the vector of the target parameters as $\breve{\mathbf{p}}^{\circ} = [\alpha^{\circ}, \beta^{\circ}, g^{\circ}, \dot{\mathbf{p}}^{\circ T}]^{T}$, where α° is the azimuth angle of the target relative to the sensor $\mathbf{m}_1$, β° is the elevation angle of the target relative to the sensor $\mathbf{m}_1$, and g° is the inverse range between the target and the sensor $\mathbf{m}_1$ and $g^{\circ} = 1/d_1$.

Squaring both sides of Formula (1) and combining it with formula (2), we obtain the TDOA equation [9]:

$$d_{i1}^{\circ}{}^{2} + 2d_{i1}^{\circ}d_{1}^{\circ} = \mathbf{m}_i^T\mathbf{m}_i - \mathbf{m}_1^T\mathbf{m}_1 - 2(\mathbf{m}_i - \mathbf{m}_1)^T\mathbf{p}^{\circ}. \tag{6}$$

From Equation (6), we can see that the TDOA equation is only related to the position $\mathbf{p}^{\circ}$ and not to the velocity $\dot{\mathbf{p}}^{\circ}$, so we let (6) be derived with respect to time to obtain the FDOA equation:

$$\dot{d}_{i1}^{\circ}d_{i1}^{\circ} + \dot{d}_{i1}^{\circ}d_{1}^{\circ} + d_{i1}^{\circ}\dot{d}_{1}^{\circ} = \dot{\mathbf{m}}_i^T\mathbf{m}_i - \dot{\mathbf{m}}_1^T\mathbf{m}_1 - (\dot{\mathbf{m}}_i - \dot{\mathbf{m}}_1)^T\mathbf{p}^{\circ} - (\mathbf{m}_i - \mathbf{m}_1)^T\dot{\mathbf{p}}^{\circ}. \tag{7}$$

To express the TDOA and FDOA equations using the MPR form, we can make $\mathbf{p}^{\circ} = (\mathbf{p}^{\circ} - \mathbf{m}_1) + \mathbf{m}_1$ and divide both sides of (6) and (7) by $2d_1^{\circ}$:

$$d_{i1}^{\circ} = (\mathbf{m}_1 - \mathbf{m}_i)^T\overline{\mathbf{p}}^{\circ} + 0.5(\left|\left|\mathbf{m}_i - \mathbf{m}_1\right|\right|_2 - d_{i1}^{\circ}{}^{2})g^{\circ}, \tag{8}$$

$$\dot{d}_{i1}^{\circ} = (\dot{\mathbf{m}}_1 - \dot{\mathbf{m}}_i)^T\overline{\mathbf{p}}^{\circ} + [(\mathbf{m}_i - \mathbf{m}_1)^T\dot{\mathbf{m}}_i - \dot{d}_{i1}^{\circ}d_{i1}^{\circ}]g^{\circ} - (\mathbf{m}_i - \mathbf{m}_1)^T\dot{\mathbf{p}}^{\circ}g^{\circ} - d_{i1}^{\circ}\dot{d}_{1}^{\circ}g^{\circ}, \tag{9}$$

where $\overline{\mathbf{p}}^{\circ}$ is the unit vector of the $\mathbf{m}_1$ pointing to the target, which has the expression

$$\overline{\mathbf{p}}^{\circ} = (\mathbf{p}^{\circ} - \mathbf{m}_1)/d_1^{\circ} = [\cos\alpha^{\circ}\cos\beta^{\circ}, \sin\alpha^{\circ}\cos\beta^{\circ}, \sin\beta^{\circ}]^T. \tag{10}$$

Then, substituting (5) into (8) and (9) and keeping only the linear error term, we can finally obtain the equations considering noise:

$$d_{i1} - [(\mathbf{m}_1 - \mathbf{m}_i)^T\overline{\mathbf{p}}^{\circ} + 0.5(\left|\left|\mathbf{m}_i - \mathbf{m}_1\right|\right|_2 - d_{i1}^2)g^{\circ}] = \varepsilon_i, \tag{11}$$

$$\dot{d}_{i1} - \left\{(\dot{\mathbf{m}}_1 - \dot{\mathbf{m}}_i)^T\overline{\mathbf{p}}^{\circ} + [(\mathbf{m}_i - \mathbf{m}_1)^T\dot{\mathbf{m}}_i - \dot{d}_{i1}d_{i1}]g^{\circ} - (\mathbf{m}_i - \mathbf{m}_1)^T\dot{\mathbf{p}}^{\circ}g^{\circ} - d_{i1}\dot{d}_{1}^{\circ}g^{\circ}\right\} = \dot{\varepsilon}_i, \tag{12}$$

where $i = 2, 3, \cdots, M$, $\varepsilon_i = \frac{d_i^{\circ}}{d_1^{\circ}}\eta_{i1}$ and $\dot{\varepsilon}_i = \frac{\dot{d}_i^{\circ}}{d_1^{\circ}}\eta_{i1} + \frac{d_i^{\circ}}{d_1^{\circ}}\dot{\eta}_{i1}$ represent noise error terms.

3.1. Unconstrained WLS

To design an estimation method for the nonlinear Equations (11) and (12), we define an auxiliary vector $\boldsymbol{\kappa}_1^{\circ} = [\overline{\mathbf{p}}^{\circ T}, g^{\circ}, (\dot{\mathbf{p}}^{\circ}g^{\circ})^T, \dot{d}_1^{\circ}g^{\circ}]^T$ of size 1×8 to turn the equations into linear equations and write the $M - 1$ sets of measurement Equations (11) and (12) as a matrix:

$$\mathbf{f}_1\text{-}\mathbf{E}_1\boldsymbol{\kappa}_1^{\circ}\text{=}\boldsymbol{\varepsilon}_1, \tag{13}$$

where

$$\mathbf{f}_1\text{=}[\mathbf{d}^T, \dot{\mathbf{d}}^T]^T, \tag{14}$$

$$\mathbf{E}_1 = \begin{bmatrix} (\mathbf{m}_1 - \mathbf{m}_2)^T & 0.5(||\mathbf{m}_2 - \mathbf{m}_1||_2 - d_{21}^2) & \mathbf{0}_3^T & 0 \\ \vdots & \vdots & \vdots & \vdots \\ (\mathbf{m}_1 - \mathbf{m}_M)^T & 0.5(||\mathbf{m}_M - \mathbf{m}_1||_2 - d_{M1}^2) & \mathbf{0}_3^T & 0 \\ (\dot{\mathbf{m}}_1 - \dot{\mathbf{m}}_2)^T & (\mathbf{m}_2 - \mathbf{m}_1)^T \dot{\mathbf{m}}_2 - \dot{d}_{21} d_{21} & (\mathbf{m}_1 - \mathbf{m}_2)^T & -d_{21} \\ \vdots & \vdots & \vdots & \vdots \\ (\dot{\mathbf{m}}_1 - \dot{\mathbf{m}}_M)^T & (\mathbf{m}_M - \mathbf{m}_1)^T \dot{\mathbf{m}}_M - \dot{d}_{M1} d_{M1} & (\mathbf{m}_1 - \mathbf{m}_M)^T & -d_{M1} \end{bmatrix}, \tag{15}$$

$$\boldsymbol{\varepsilon}_1 = [\varepsilon_1, \cdots, \varepsilon_M, \dot{\varepsilon}_1, \cdots, \dot{\varepsilon}_M]^T. \tag{16}$$

Although we now have the linear matrix, Equation (13), the elements of the auxiliary vector have a nonlinear relationship with the target parameters, which can affect the estimation accuracy.

The derivation of the constraints is given below. Multiplying both sides of Equation (4) by g° gives the first constraint:

$$\dot{d}_1^\circ g^\circ = \frac{(\dot{\mathbf{p}}^\circ - \dot{\mathbf{m}}_1)^T (\mathbf{p}^\circ - \mathbf{m}_1)}{d_1^\circ} g^\circ = (\dot{\mathbf{p}}^\circ g^\circ - \dot{\mathbf{m}}_1 g^\circ)^T \overline{\mathbf{p}}^\circ. \tag{17}$$

Moreover, according to the definition of $\overline{\mathbf{p}}^\circ$, we can obtain the second constraint:

$$\overline{\mathbf{p}}^{\circ T} \overline{\mathbf{p}}^\circ = 1. \tag{18}$$

Thus, the localization problem is transformed into a WLS problem with two constraints:

$$\begin{array}{ll} \min\limits_{\psi_1} & (\mathbf{f}_1 - \mathbf{E}_1 \boldsymbol{\kappa}_1^\circ)^T \mathbf{W}_1 (\mathbf{f}_1 - \mathbf{E}_1 \boldsymbol{\kappa}_1^\circ) \\ s.t. & \dot{d}_1^\circ g^\circ = (\dot{\mathbf{p}}^\circ g^\circ - \dot{\mathbf{m}}_1 g^\circ)^T \overline{\mathbf{p}}^\circ \\ & \overline{\mathbf{p}}^{\circ T} \overline{\mathbf{p}}^\circ = 1 \end{array}, \tag{19}$$

where $\mathbf{W}_1$ is the weighting matrix with the expression

$$\mathbf{W}_1 = E[\boldsymbol{\varepsilon}_1 \boldsymbol{\varepsilon}_1^T]^{-1} = (\mathbf{B}_1 \mathbf{Q} \mathbf{B}_1^T)^{-1}, \tag{20}$$

and

$$\mathbf{B}_1 = \begin{bmatrix} \mathbf{B} & O_{(M-1)\times(M-1)} \\ \dot{\mathbf{B}} & \mathbf{B} \end{bmatrix}, \tag{21}$$

$$\begin{cases} \mathbf{B} = diag([d_2^\circ / d_1^\circ, \cdots, d_M^\circ / d_1^\circ]) \\ \dot{\mathbf{B}} = diag([\dot{d}_2^\circ / d_1^\circ, \cdots, \dot{d}_M^\circ / d_1^\circ]) \end{cases}. \tag{22}$$

Through (2) and (22), we can know that $\mathbf{W}_1$ depends on the true parameters of the source, which are unknown. To solve this problem, we can make $\mathbf{W}_1 = \mathbf{Q}^{-1}$ [18] and ignore the two constraints to solve for the unconstrained WLS estimate of $\boldsymbol{\kappa}_1^\circ$:

$$\boldsymbol{\kappa}_1 = (\mathbf{E}_1^T \mathbf{W}_1 \mathbf{E}_1)^{-1} \mathbf{E}_1^T \mathbf{W}_1 \mathbf{f}_1. \tag{23}$$

From (23), we can obtain the parameters needed for $\mathbf{W}_1$. Then, we continue to solve for $\boldsymbol{\kappa}_1$ by substituting $\mathbf{W}_1$ into (23), and so on several times to obtain a more accurate $\boldsymbol{\kappa}_1$.

We assume that the measurement error in matrix $\mathbf{E}_1$ is negligible, and then the covariance matrix of $\boldsymbol{\kappa}_1$ is

$$\text{cov}(\boldsymbol{\kappa}_\mathbf{1}) = (\mathbf{E}_1^T \mathbf{W}_1 \mathbf{E}_1)^{-1}. \tag{24}$$

3.2. Optimization Algorithm Based on Constraints

3.2.1. Considering the First Constraint

This subsection uses the first constraint $\dot{d}_1^\circ g^\circ = (\dot{\mathbf{p}}^\circ g^\circ - \dot{\mathbf{m}}_1 g^\circ)^T \overline{\mathbf{p}}^\circ$ to reduce the estimation error of the previous step. We define a new auxiliary vector $\hat{\boldsymbol{\kappa}}_1 = [\hat{\overline{\mathbf{p}}}^T, \hat{g}, (\dot{\mathbf{p}}\hat{g})^T]^T$ of size 1 × 7 and express the auxiliary as the previous step estimate minus the error:

$$\hat{\boldsymbol{\kappa}}_1 = \boldsymbol{\kappa}_1(1:7) - \Delta\boldsymbol{\kappa}_1, \tag{25}$$

where $\Delta\boldsymbol{\kappa}_1 = [\Delta\overline{\mathbf{p}}^T, \Delta g, (\Delta\dot{\mathbf{p}}\mathbf{g})^T]^T$ is the estimation error term.

By expanding the first constraint in a first-order Taylor expansion [30], we obtain

$$\dot{d}_1^\circ \mathrm{g}^\circ \approx \dot{d}_1 \mathrm{g} + \overline{\mathbf{p}}^T \Delta\dot{\mathbf{p}}\mathbf{g} - \dot{\mathbf{m}}_1^T \overline{\mathbf{p}} \Delta \mathrm{g} + (\dot{\mathbf{p}}\mathrm{g} - \dot{\mathbf{m}}_1 \mathrm{g})^T \Delta\overline{\mathbf{p}}. \tag{26}$$

Then, substituting (25) and (26) into (11) and (12) and combining the $M-1$ sets of measurement equations, we obtain the matrix equation

$$\mathbf{a} - \mathbf{E}_2 \Delta\boldsymbol{\kappa}_1 = \boldsymbol{\varepsilon}_1, \tag{27}$$

where

$$\mathbf{E}_2 = [\mathbf{E}_{21}^T, \mathbf{E}_{22}^T]^T, \mathbf{a} = [\mathbf{a}_1^T, \mathbf{a}_2^T]^T, \tag{28}$$

and the respective elements in the ith row of $\mathbf{a}_1$ and $\mathbf{a}_2$ are

$$\mathbf{a}_1(i) = d_{i1} - (\mathbf{m}_1 - \mathbf{m}_i)^T \overline{\mathbf{p}} - 0.5(\left|\left|\mathbf{m}_i - \mathbf{m}_1\right|\right|_2 - d_{i1}^2) g, \tag{29}$$

$$\mathbf{a}_2(i) = \dot{d}_{i1} - (\dot{\mathbf{m}}_1 - \dot{\mathbf{m}}_i)^T \overline{\mathbf{p}} - [(\mathbf{m}_i - \mathbf{m}_1)^T \dot{\mathbf{m}}_i - \dot{d}_{i1} d_{i1}] g + (\mathbf{m}_i - \mathbf{m}_1)^T \dot{\mathbf{p}} g + d_{i1} \dot{d}_1 g, \tag{30}$$

and the respective elements in the ith row of $\mathbf{E}_{21}$ and $\mathbf{E}_{22}$ are

$$\mathbf{E}_{21}(i) = [(\mathbf{m}_1 - \mathbf{m}_i)^T, 0.5(\left|\left|\mathbf{m}_i - \mathbf{m}_1\right|\right|_2 - d_{i1}^2), \mathbf{0}_3^T], \tag{31}$$

$$\mathbf{E}_{22}(i) = \begin{bmatrix} (\dot{\mathbf{m}}_1 - \dot{\mathbf{m}}_i) + d_{i1}(\dot{\mathbf{p}} g - \dot{\mathbf{m}}_1 g) \\ (\mathbf{m}_i - \mathbf{m}_1)^T \dot{\mathbf{m}}_i - \dot{d}_{i1} d_{i1} - d_{i1} \mathbf{m}_1^T \overline{\mathbf{p}} \\ d_{i1} \overline{\mathbf{p}} - (\mathbf{m}_i - \mathbf{m}_1) \end{bmatrix}^T. \tag{32}$$

Using WLS to solve (25), we can obtain the error:

$$\Delta\boldsymbol{\kappa}_\mathbf{1} = (\mathbf{E}_2^T \mathbf{W}_1 \mathbf{E}_2)^{-1} \mathbf{E}_2^T \mathbf{W}_1 \mathbf{b}. \tag{33}$$

Therefore, the final estimate is obtained as

$$\hat{\boldsymbol{\kappa}}_\mathbf{1} = [\hat{\overline{\mathbf{p}}}^T, \hat{g}, (\dot{\mathbf{p}}\hat{g})^T]^T = \boldsymbol{\kappa}_\mathbf{1}(1:7) - \Delta\boldsymbol{\kappa}_\mathbf{1}. \tag{34}$$

Similarly, we assume that the measurement error in matrix $\mathbf{E}_2$ is negligible, and then the covariance matrix of (34) is

$$\mathrm{cov}(\hat{\boldsymbol{\kappa}}_\mathbf{1}) = \mathrm{cov}(\boldsymbol{\kappa}_\mathbf{1}(1:7)) + (\mathbf{E}_2^T \mathbf{W}_1 \mathbf{E}_2)^{-1}. \tag{35}$$

3.2.2. Considering the Second Constraint

In the previous subsection, we ignored the constraint on the unit vector $\overline{\mathbf{p}}^{\circ}$. Since this constraint is only relevant for the DOA, this section uses this constraint to reduce the estimation results of $\hat{\overline{\mathbf{p}}}$ from the previous step.

Defining $\hat{\overline{\mathbf{p}}}$ as the true value plus the error value, we obtain

$$\hat{\overline{\mathbf{p}}} = \overline{\mathbf{p}}^{\circ} + \Delta\overline{\mathbf{p}}, \tag{36}$$

where $\Delta\overline{\mathbf{p}}$ is the error vector.

Substituting (36) into the second constraint and retaining the linear error term yields

$$\hat{\overline{\mathbf{p}}} \odot \hat{\overline{\mathbf{p}}} - \overline{\mathbf{p}}^{\circ} \odot \overline{\mathbf{p}}^{\circ} = 2\hat{\overline{\mathbf{p}}} \odot \Delta\overline{\mathbf{p}}. \tag{37}$$

Since $\hat{\overline{\mathbf{p}}}$ is the first three elements of $\hat{\boldsymbol{\kappa}}_{\mathbf{1}}$, Equation (37) can be rewritten as

$$\hat{\boldsymbol{\kappa}}_1(1:3) \odot \hat{\boldsymbol{\kappa}}_1(1:3) - \boldsymbol{\kappa}_1^{\circ}(1:3) \odot \boldsymbol{\kappa}_1^{\circ}(1:3) = 2\hat{\boldsymbol{\kappa}}_1(1:3) \odot \Delta\hat{\boldsymbol{\kappa}}_1(1:3), \tag{38}$$

where $\Delta\hat{\boldsymbol{\kappa}}_1(1:3) = \Delta\overline{\mathbf{p}}$.

Through Equation (10), we know that

$$\boldsymbol{\kappa}_1^{\circ}(1)^2 + \boldsymbol{\kappa}_1^{\circ}(2)^2 + \boldsymbol{\kappa}_1^{\circ}(3)^2 = 1. \tag{39}$$

Then, by defining the vector $\boldsymbol{\kappa}_2 = [\overline{\mathbf{p}}^{\circ}(1)^2, \overline{\mathbf{p}}^{\circ}(2)^2, \overline{\mathbf{p}}^{\circ}(1)^2 + \overline{\mathbf{p}}^{\circ}(2)^2]^T$ and combining (38) and (39) to create the matrix, we can obtain the matrix

$$\mathbf{f}_2 - \mathbf{E}_3\boldsymbol{\kappa}_2 = \boldsymbol{\varepsilon}_2, \tag{40}$$

where

$$\mathbf{f}_2 = \begin{bmatrix} \hat{\boldsymbol{\kappa}}_1(1:2) \odot \hat{\boldsymbol{\kappa}}_1(1:2) \\ \hat{\boldsymbol{\kappa}}_1(3)^2 - 1 \end{bmatrix}, \tag{41}$$

$$\mathbf{E}_3 = \begin{bmatrix} \mathbf{I}_2 \\ -\mathbf{1}_2^T \end{bmatrix}, \tag{42}$$

$$\boldsymbol{\varepsilon}_2 = \mathbf{B}_2\Delta\hat{\boldsymbol{\kappa}}_1(1:3), \tag{43}$$

and

$$\mathbf{B}_2 = diag([2\hat{\overline{\mathbf{p}}}(1:3)]). \tag{44}$$

The WLS result of (40) is

$$\boldsymbol{\kappa}_2 = \left(\mathbf{E}_3^T\mathbf{W}_2\mathbf{E}_3\right)^{-1}\mathbf{E}_3^T\mathbf{W}_2\mathbf{f}_2, \tag{45}$$

where $\mathbf{W}_2$ is the weighting matrix and

$$\mathbf{W}_2 = E[\boldsymbol{\varepsilon}_2\boldsymbol{\varepsilon}_2^T]^{-1} = \left(\mathbf{B}_2\mathrm{cov}(\hat{\boldsymbol{\kappa}}_{\mathbf{1}}(1:3, 1:3))^{-1}\mathbf{B}_2^T\right)^{-1}. \tag{46}$$

The calculation of $\mathbf{W}_2$ requires the real parameters of the source, which can be derived with the help of the estimation results of the previous step since the parameter errors of the source obtained in the previous step are already small.

Finally, we can obtain the expression for the moving target parameter as

$$\begin{cases} \breve{\alpha} = \arctan\left(\frac{\sqrt{\kappa_2(2)}}{\sqrt{\kappa_2(1)}}\right) \\ \breve{\beta} = \arctan\left(\frac{\sqrt{1-\kappa_2(1)-\kappa_2(2)}}{\sqrt{\kappa_2(1)+\kappa_2(2)}}\right) \\ \breve{g} = \hat{\kappa}_1(4) \\ \breve{\dot{\mathbf{p}}} = \frac{\hat{\kappa}_1(5{:}7)}{\hat{\kappa}_1(4)} \end{cases}. \tag{47}$$

3.3. *The Algorithmic Steps*

In this subsection, we summarize the process of each step of the proposed method in this paper, as shown in Figure 2.

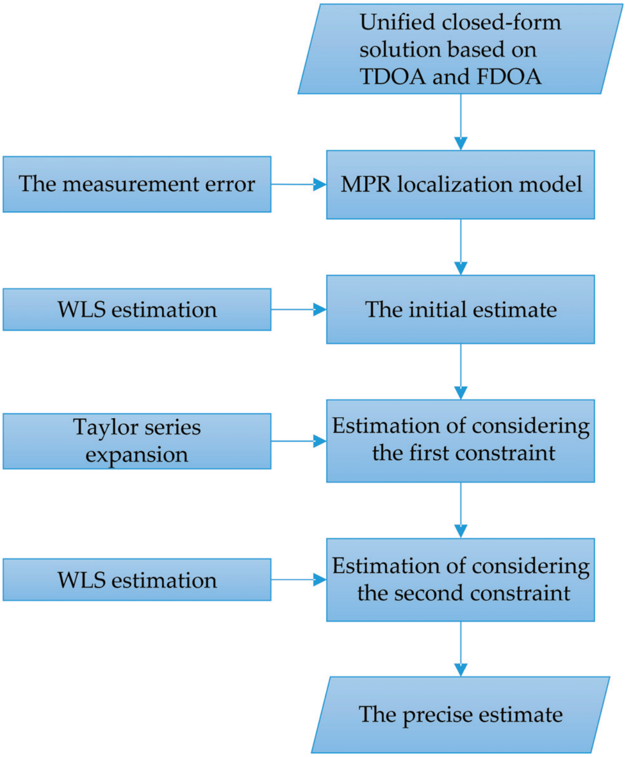

Unified closed-form solution based on TDOA and FDOA
The measurement error
MPR localization model
WLS estimation
The initial estimate
Taylor series expansion
Estimation of considering the first constraint
WLS estimation
Estimation of considering the second constraint
The precise estimate

Figure 2. The algorithmic flowchart.

4. Derivation of the CRLB

In this section, we present a detailed formulation of the CRLB under the model proposed in this paper. The CRLB is a lower bound describing the optimal performance of an unbiased estimator [31], which is defined as

$$\mathrm{CRLB}(\breve{\mathbf{p}}^{\circ}) = \mathrm{FIM}^{-1}(\breve{\mathbf{p}}^{\circ}), \tag{48}$$

where FIM is the Fisher information matrix and

$$\mathrm{FIM}(\breve{\mathbf{p}}^{\circ}) = \frac{\partial \mathbf{F}^{\circ T}}{\partial \breve{\mathbf{p}}^{\circ}} \mathbf{Q}^{-1} \frac{\partial \mathbf{F}^{\circ}}{\partial \breve{\mathbf{p}}^{\circ T}}, \tag{49}$$

where $\mathbf{F}^{\circ} = [\mathbf{d}^{\circ T}, \dot{\mathbf{d}}^{\circ T}]^{T}$.

The partial derivative of $\mathbf{F}^{\circ}$ with respect to $\breve{\mathbf{p}}^{\circ}$ is expressed as

$$\frac{\partial \mathbf{F}^{\circ}}{\partial \breve{\mathbf{p}}^{\circ}} = \begin{bmatrix} \frac{\partial \mathbf{d}^{\circ}}{\partial \alpha^{\circ}} & \frac{\partial \mathbf{d}^{\circ}}{\partial \beta^{\circ}} & \frac{\partial \mathbf{d}^{\circ}}{\partial g^{\circ}} & \frac{\partial \mathbf{d}^{\circ}}{\partial \dot{\mathbf{p}}^{\circ T}} \\ \frac{\partial \dot{\mathbf{d}}^{\circ}}{\partial \alpha^{\circ}} & \frac{\partial \dot{\mathbf{d}}^{\circ}}{\partial \beta^{\circ}} & \frac{\partial \dot{\mathbf{d}}^{\circ}}{\partial g^{\circ}} & \frac{\partial \dot{\mathbf{d}}^{\circ}}{\partial \dot{\mathbf{p}}^{\circ T}} \end{bmatrix}. \tag{50}$$

From (1) and (4), we have

$$\begin{aligned} d_{i1}^{\circ} &= ||\mathbf{p}^{\circ} - \mathbf{m}_i|| - ||\mathbf{p}^{\circ} - \mathbf{m}_1||^{\circ} \\ &= \tfrac{1}{g^{\circ}} \sqrt{1 - 2g^{\circ}(\mathbf{m}_i - \mathbf{m}_1)^T \overline{\mathbf{p}}^{\circ} + g^{\circ 2}(\mathbf{m}_i - \mathbf{m}_1)^T(\mathbf{m}_i - \mathbf{m}_1)} - \tfrac{1}{g^{\circ}} \end{aligned}, \tag{51}$$

$$\begin{aligned} \dot{d}_{i1}^{\circ} &= \tfrac{(\dot{\mathbf{p}}^{\circ} - \dot{\mathbf{m}}_i)^T (\mathbf{p}^{\circ} - \mathbf{m}_i)}{d_i^{\circ}} - \tfrac{(\dot{\mathbf{p}}^{\circ} - \dot{\mathbf{m}}_1)^T (\mathbf{p}^{\circ} - \mathbf{m}_1)}{d_1^{\circ}} \\ &= \tfrac{(\dot{\mathbf{p}}^{\circ} - \dot{\mathbf{m}}_i)^T [\overline{\mathbf{p}}^{\circ} + g^{\circ}(\mathbf{m}_1 - \mathbf{m}_i)]}{g^{\circ} d_i^{\circ}} - (\dot{\mathbf{p}}^{\circ} - \dot{\mathbf{m}}_1)^T \overline{\mathbf{p}}^{\circ} \end{aligned}, \tag{52}$$

where $i = 2, \cdots, M$.

Using (51), we can obtain

$$\frac{\partial \mathbf{d}^{\circ}}{\partial \alpha^{\circ}} = \begin{bmatrix} \frac{(\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_2 - \mathbf{m}_1))^T}{\left|\left|\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_2 - \mathbf{m}_1)\right|\right| g^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \alpha^{\circ}} \\ \vdots \\ \frac{(\overline{\mathbf{u}}^{\circ} - g^{\circ}(\mathbf{m}_M - \mathbf{m}_1))^T}{\left|\left|\overline{\mathbf{u}}^{\circ} - g^{\circ}(\mathbf{m}_M - \mathbf{m}_1)\right|\right| g^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \alpha^{\circ}} \end{bmatrix}, \tag{53}$$

$$\frac{\partial \mathbf{d}^{\circ}}{\partial \beta^{\circ}} = \begin{bmatrix} \frac{(\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_2 - \mathbf{m}_1))^T}{\left|\left|\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_2 - \mathbf{m}_1)\right|\right| g^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \beta^{\circ}} \\ \vdots \\ \frac{(\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_M - \mathbf{m}_1))^T}{\left|\left|\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_M - \mathbf{m}_1)\right|\right| g^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \beta^{\circ}} \end{bmatrix}, \tag{54}$$

$$\frac{\partial \mathbf{d}^{\circ}}{\partial g^{\circ}} = \begin{bmatrix} \frac{(\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_2 - \mathbf{m}_1))^T \cdot \mathbf{p}}{\left|\left|\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_2 - \mathbf{m}_1)\right|\right| g^{\circ}} + \frac{1}{g^{\circ 2}} \\ \vdots \\ \frac{(\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_M - \mathbf{m}_1))^T \cdot \mathbf{p}}{\left|\left|\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_M - \mathbf{m}_1)\right|\right| g^{\circ}} + \frac{1}{g^{\circ 2}} \end{bmatrix}, \tag{55}$$

$$\frac{\partial \mathbf{d}^{\circ}}{\partial \dot{\mathbf{p}}^{\circ}} = \mathbf{O}_{(M-1)\times 3}, \tag{56}$$

where

$$\frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \alpha^{\circ}} = \begin{bmatrix} -\sin\alpha^{\circ}\cos\beta^{\circ} \\ \cos\alpha^{\circ}\cos\beta^{\circ} \\ 0 \end{bmatrix}, \tag{57}$$

$$\frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \beta^{\circ}} = \begin{bmatrix} -\cos\alpha^{\circ}\sin\beta^{\circ} \\ -\sin\alpha^{\circ}\sin\beta^{\circ} \\ \cos\beta^{\circ} \end{bmatrix}, \tag{58}$$

$$\mathbf{p} = -\overline{\mathbf{p}}^{\circ}/g^{\circ}. \tag{59}$$

Using (52), we can obtain

$$\frac{\partial \dot{\mathbf{d}}^{\circ}}{\partial \alpha^{\circ}} = \begin{bmatrix} \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_2)^T - g^{\circ}d_2^{\circ}(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_1)^T}{g^{\circ}d_2^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \theta^{\circ}} - \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_2)^T[\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_2-\mathbf{m}_1)]}{g^{\circ}d_2^{\circ 2}} \cdot \frac{\partial d_2^{\circ}}{\partial \alpha^{\circ}} \\ \vdots \\ \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_M)^T - g^{\circ}d_M^{\circ}(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_1)^T}{g^{\circ}d_M^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \theta^{\circ}} - \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_M)^T[\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_M-\mathbf{m}_1)]}{g^{\circ}d_M^{\circ 2}} \cdot \frac{\partial d_M^{\circ}}{\partial \alpha^{\circ}} \end{bmatrix}, \tag{60}$$

$$\frac{\partial \dot{\mathbf{d}}^{\circ}}{\partial \beta^{\circ}} = \begin{bmatrix} \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_2)^T - g^{\circ}d_2^{\circ}(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_1)^T}{g^{\circ}d_2^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \beta^{\circ}} - \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_2)^T[\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_2-\mathbf{m}_1)]}{g^{\circ}d_2^{\circ 2}} \cdot \frac{\partial d_2^{\circ}}{\partial \beta^{\circ}} \\ \vdots \\ \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_M)^T - g^{\circ}d_M^{\circ}(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_1)^T}{g^{\circ}d_M^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \beta^{\circ}} - \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_M)^T[\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_M-\mathbf{m}_1)]}{g^{\circ}d_M^{\circ 2}} \cdot \frac{\partial d_M^{\circ}}{\partial \beta^{\circ}} \end{bmatrix}, \tag{61}$$

$$\frac{\partial \dot{\mathbf{d}}^{\circ}}{\partial g^{\circ}} = \begin{bmatrix} -\frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_2)^T\overline{\mathbf{p}}^{\circ}}{g^{\circ 2}d_2^{\circ}} - \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_2)^T[\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_2-\mathbf{m}_1)]}{g^{\circ}d_2^{\circ 2}} \cdot \frac{\partial d_2^{\circ}}{\partial g^{\circ}} \\ \vdots \\ -\frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_M)^T\overline{\mathbf{p}}^{\circ}}{g^{\circ 2}d_M^{\circ}} - \frac{(\dot{\mathbf{p}}^{\circ}-\dot{\mathbf{m}}_M)^T[\overline{\mathbf{p}}^{\circ} - g^{\circ}(\mathbf{m}_M-\mathbf{m}_1)]}{g^{\circ}d_M^{\circ 2}} \cdot \frac{\partial d_M^{\circ}}{\partial g^{\circ}} \end{bmatrix}, \tag{62}$$

$$\frac{\partial \dot{\mathbf{d}}^{\circ}}{\partial \dot{\mathbf{p}}^{\circ T}} = \begin{bmatrix} \frac{(1-g^{\circ}d_2^{\circ})\overline{\mathbf{p}}^{\circ T}}{g^{\circ}d_2^{\circ}} - \frac{(\mathbf{m}_2-\mathbf{m}_1)^T}{d_2^{\circ}} \\ \vdots \\ \frac{(1-g^{\circ}d_M^{\circ})\overline{\mathbf{p}}^{\circ T}}{g^{\circ}d_M^{\circ}} - \frac{(\mathbf{m}_M-\mathbf{m}_1)^T}{d_M^{\circ}} \end{bmatrix}, \tag{63}$$

where r_i° is

$$d_i^{\circ} = \frac{1}{g^{\circ}}\sqrt{1 - 2g^{\circ}(\mathbf{m}_i - \mathbf{m}_1)^T\overline{\mathbf{p}}^{\circ} + g^{\circ 2}(\mathbf{m}_i - \mathbf{m}_1)^T(\mathbf{m}_i - \mathbf{m}_1)}, \tag{64}$$

and from (64), we can obtain

$$\frac{\partial d_i^{\circ}}{\partial \alpha^{\circ}} = -\frac{(\mathbf{m}_i - \mathbf{m}_1)^T}{g^{\circ}d_i^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \alpha^{\circ}}, \tag{65}$$

$$\frac{\partial d_i^{\circ}}{\partial \beta^{\circ}} = -\frac{(\mathbf{m}_i - \mathbf{m}_1)^T}{g^{\circ}d_i^{\circ}} \cdot \frac{\partial \overline{\mathbf{p}}^{\circ}}{\partial \beta^{\circ}}, \tag{66}$$

$$\frac{\partial d_i^{\circ}}{\partial g^{\circ}} = \frac{g^{\circ}(\mathbf{m}_i - \mathbf{m}_1)^T\overline{\mathbf{p}}^{\circ} - 1}{g^{\circ 3}d_i^{\circ}}. \tag{67}$$

With the above derivation, we end up with the FIM($\breve{\mathbf{p}}^{\circ}$). Substituting (49) into (48), we obtain the CRLB. The trace of the CRLB matrix is the minimum theoretical variance of the source parameters.

5. Numerical Simulations

In this section, we will verify the effectiveness of the proposed method through simulation. We use seven sensors, and their positions and velocities are shown in Table 1. For simplicity, we place the reference sensor at the origin. The covariance matrices are set to $\mathbf{Q}_t = \sigma^2\mathbf{R}$ and $\mathbf{Q}_f = 0.1\sigma^2\mathbf{R}$, where σ^2 is the variance of the measurement noise and $\mathbf{R}$ is a matrix with diagonal elements of 1 and the rest of the elements of 0.5 [18,21]. To fully analyze the positioning performance, we use two evaluation metrics, mean square error (MSE) and bias, which are calculated as follows:

$$\mathrm{MSE}(\breve{\mathbf{p}}) = \frac{1}{L}\sum_{l=1}^{L} ||\breve{\mathbf{p}}^{\circ} - \breve{\mathbf{p}}_l||^2, \tag{68}$$

$$Bias(\breve{\mathbf{p}}) = \left\| \breve{\mathbf{p}}^{\circ} - \frac{1}{L}\sum_{l=1}^{L} \breve{\mathbf{p}}_l \right\|, \tag{69}$$

where $L = 2000$ is the number of Monte Carlo runs and $\breve{\mathbf{p}}_l$ is the estimation of the true value $\breve{\mathbf{p}}^{\circ}$ in the lth Monte Carlo experiment.

Table 1. Positions and speeds of sensors.

Sensors *no.n*	Position (m)	Velocity (m/s)
1	$\mathbf{s}_1^{\circ} = [0, 0, 0]^T$	$\dot{\mathbf{s}}_1^{\circ} = [30, -20, -20]^T$
2	$\mathbf{s}_2^{\circ} = [456.4, -714.2, 636.5]^T$	$\dot{\mathbf{s}}_2^{\circ} = [-30, 10, 20]^T$
3	$\mathbf{s}_3^{\circ} = [1030.2, -65.8, 826.6]^T$	$\dot{\mathbf{s}}_3^{\circ} = [10, -20, 10]^T$
4	$\mathbf{s}_4^{\circ} = [465.7, 619.8, -789.0]^T$	$\dot{\mathbf{s}}_4^{\circ} = [10, 20, 30]^T$
5	$\mathbf{s}_5^{\circ} = [579.8, -255.4, -838.5]^T$	$\dot{\mathbf{s}}_5^{\circ} = [-20, 10, 10]^T$
6	$\mathbf{s}_6^{\circ} = [1201.2, -541.2, 351.7]^T$	$\dot{\mathbf{s}}_6^{\circ} = [30, 20, 20]^T$
7	$\mathbf{s}_7^{\circ} = [-836.6, 1004.8, -802.9]^T$	$\dot{\mathbf{s}}_7^{\circ} = [10, 20, 20]^T$

In this paper, two simulation scenarios are considered to estimate the target parameters based on target range variation and noise power variation, respectively. The algorithm presented in Section 3.2.1 of this paper is called OCWLS, and the algorithm after correcting the estimates in Section 3.2.2 is called TCWLS. Since all near-field localization algorithms in the Cartesian coordinate system have threshold effects, the classical Ho-TSWLS algorithm mentioned in [18] and the improved Ali-TSWLS algorithm mentioned in [21] are chosen as comparison algorithms. In addition, the CRLB is used as a standard for MSE estimation.

5.1. Impact of Source Range on Estimation Error

In this scenario, the target distance is increased from 10 to 200 km. The noise variance is $\sigma^2 = 1$ m^2 and the angle of the source is randomly set to $\alpha^{\circ} = 20$ deg and $\beta^{\circ} = 12$ deg. The variation in the DOA and inverse-range MSE estimations with increasing target distance is shown in Figures 3 and 4. OCWLS and Ali-TSWLS have not achieved the CRLB, and their performance is approximately 2.7 dB above the CRLB. After using the optimized algorithm, we can observe that the localization performance is improved, with TCWLS reaching the CRLB and still maintaining an accurate estimation of the DOA at long range. However, the estimation of the DOA by Ho-TSWLS fails when the range increases to 120 km, and Ali-TSWLS fails when the range increases to 160 km due to the threshold effect. As seen in Figure 4, TCWLS also achieves the CRLB for the inverse-range MSE estimate. But Ho-TSWLS and Ali-TSWLS gradually increase starting from a Range $=$ 60 km. Figure 5 shows the performance comparison of velocity MSE estimation with distance. When the Range $>$ 100 km, Ho-TSWLS starts to deviate from the CRLB, while TCWLS stays close to the CRLB until the Range $=$ 160 km. Ali-TSWLS is the best and stays close to the CRLB.

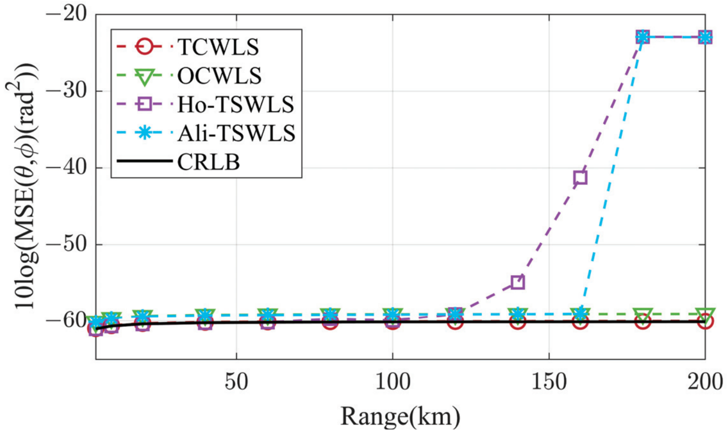


Figure 3. Variation in MSE with distance for DOA estimate.

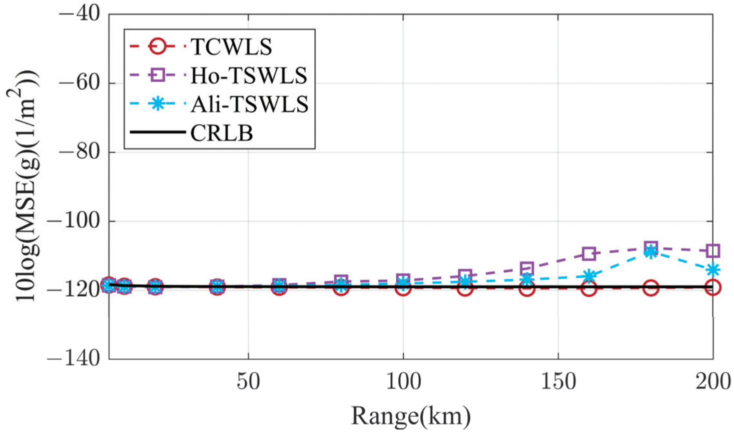


Figure 4. Variation in MSE with distance for inverse-range estimate.

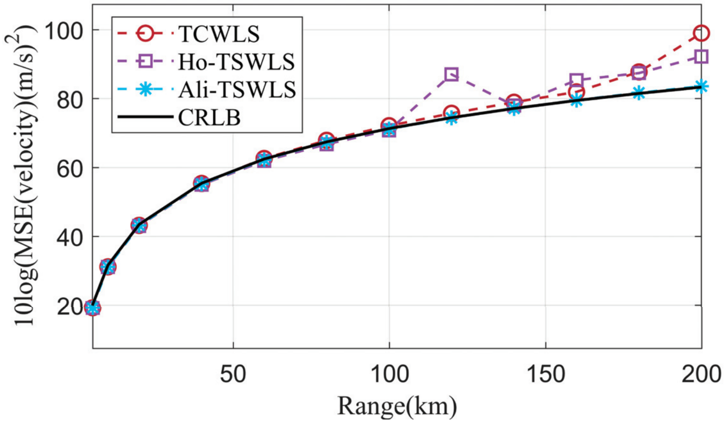


Figure 5. Variation in MSE with distance for velocity estimate.

The variation in estimation bias with distance for the source parameters is shown in Figures 6–8. For DOA estimation, the estimation bias of both OCWLS and TCWLS is much smaller than that of Ho-TSWLS and Ali-TSWLS, and TCWLS performs better than OCWLS, which is in line with the theoretical analysis. For inverse-distance estimation, the bias of TCWLS is much smaller than that of Ho-TSWLS and Ali-TSWLS, and Ali-TSWLS is about 3 dB lower than Ho-TSWLS. For speed estimation, the bias of the TCWLS and Ali-TSWLS methods is reduced by about 17 dB compared to Ho-TSWLS. The bias of TCWLS is lower than that of Ali-TSWLS up to 130 km, and beyond this point, the deviation of Ali-TSWLS becomes even lower. The results show that the proposed algorithm is equally effective for far-field targets, while Ho-TSWLS and Ali-TSWLS fail due to the threshold effect. Although Ali-TSWLS performs better for velocity estimation, it is poor for angle estimation.

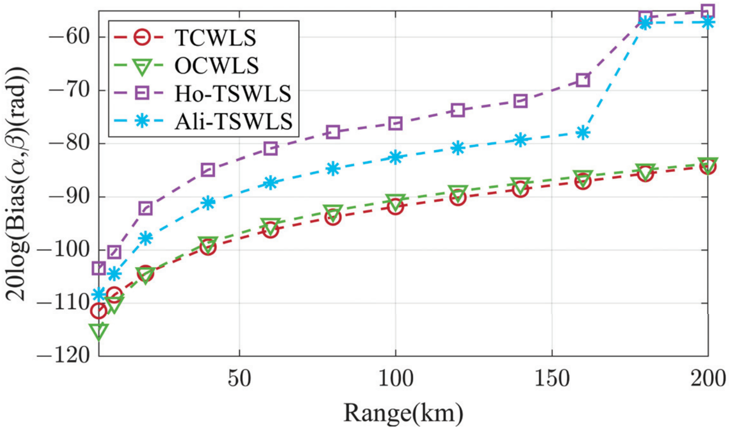


Figure 6. Variation in bias with distance for DOA estimate.

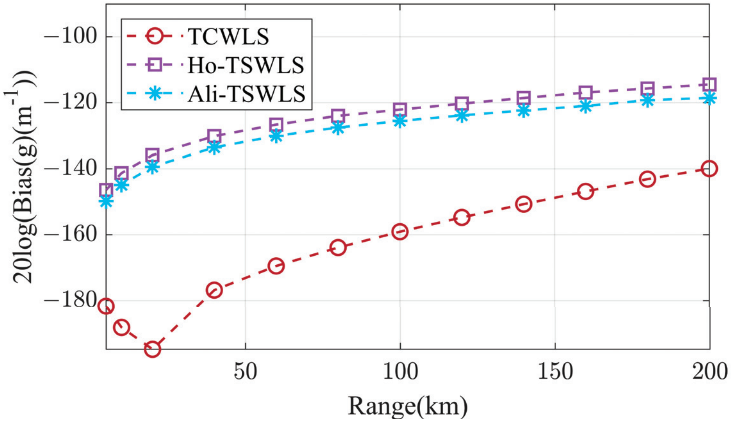


Figure 7. Variation in bias with distance for inverse-range estimate.

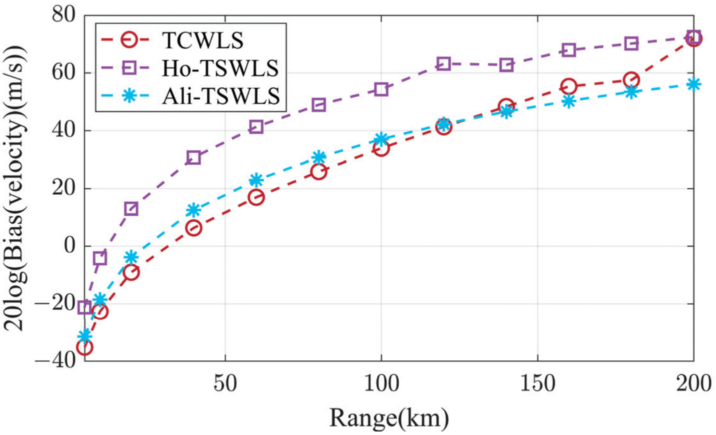


Figure 8. Variation in bias with distance for velocity estimate.

Next, we transform the source parameters from MRP to the Cartesian coordinate system and compare the MSE and deviation of the source position to ensure the validity of the estimation of the target position. Figure 9 shows that before the distance reaches 160 km, the performance of all algorithms is equivalent, and the proposed algorithm begins to have a threshold effect when the Range > 160 km. This is caused by converting the inverse range into distance. A small inverse-range perturbation will cause a huge distance error, thus causing the position MSE to become larger. Although the MSE estimates of Ho-TSWLS and Ali-TSWLS are reasonable, they cannot provide an accurate DOA estimate. Figure 10 shows that TCWLS achieves the lowest deviation performance. We can conclude that TCWLS has the best bias performance in target location estimation, but sometimes the MSE performance may be degraded due to smaller inverse-range perturbations caused by excessively long distance or strong noise.

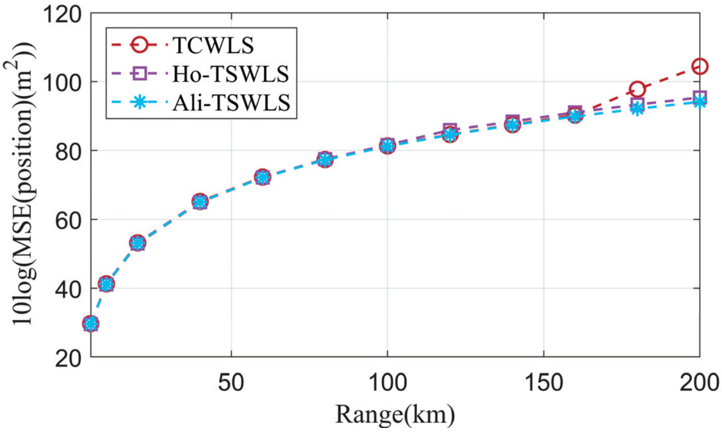


Figure 9. Variation in MSE with distance for position estimate.

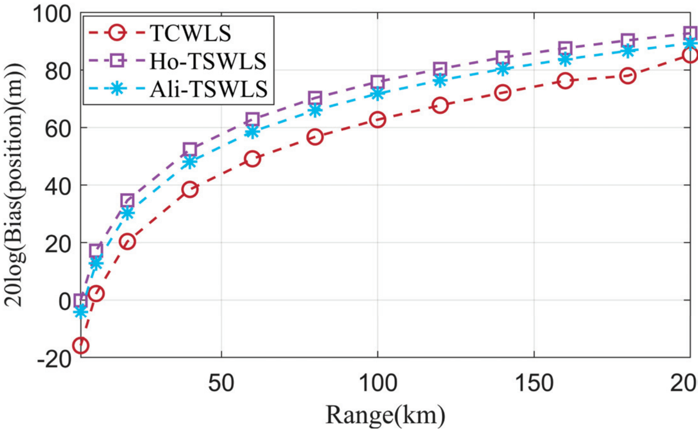


Figure 10. Variation in bias with distance for position estimate.

5.2. Impact of Measurement Noise on Estimation Error

In the previous subsection, it is known that the proposed algorithm applies to far-field sources. In this subsection, we will analyze the effect of noise power on the positioning performance of the proposed algorithm in the near-field case. Noise power increased from -40 dB to 40 dB. The target is at a fixed distance of 15 km, with coordinates $\mathbf{p}^{\circ} = [13787, 5018.2, 3118.7]^{T}$m and speed $\dot{\mathbf{p}}^{\circ} = [25, 20, 40]^{T}$m/s.

Figures 11–13 compare the MSE performance of several methods for target parameter estimation. For target angle estimation, OCWLS and TCWLS are consistently close to the CRLB, and OCWLS is about 2 dB higher than TCWLS. In contrast, Ho-TSWLS already deviates from the CRLB when $10\log(\sigma^2) = 10$, and Ali-TSWLS already deviates from the CRLB when $10\log(\sigma^2) = 20$. Similarly, TCWLS has stable performance for target inverse-range estimation, whereas Ho-TSWLS and Ali-TSWLS start to deteriorate in performance when $10\log(\sigma^2) = 20$. For speed estimation, TCWLS and Ho-TSWLS already deviate from the CRLB when $10\log(\sigma^2) > 20$, while Ali-TSWLS remains close to it until $10\log(\sigma^2) = 25$.

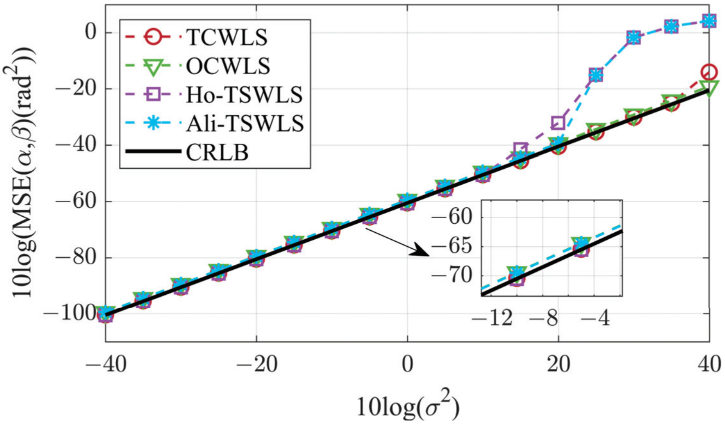


Figure 11. Variation in MSE with noise power for DOA estimate.

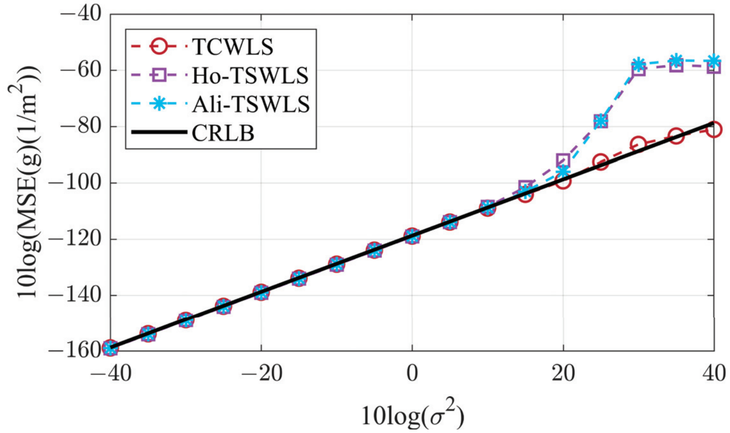


Figure 12. Variation in MSE with noise power for inverse-range estimate.

Figures 14–16 present the bias estimations. For DOA estimates, OCWLS and TCWLS are about 11 dB lower than Ho-TSWLS and about 6 dB lower than Ali-TSWLS. TCWLS has the best bias performance. For velocity estimation, TCWLS is about 19 dB lower than Ho-TSWLS when $10\log(\sigma^2) \leq 25$ and about 4.5 dB lower than Ali-TSWLS when $10\log(\sigma^2) \leq 20$. However, Ali-TSWLS performance is best when $10\log(\sigma^2) \geq 25$.

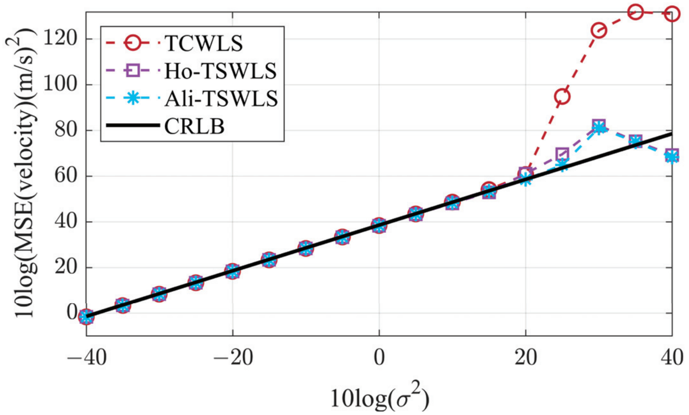


Figure 13. Variation in MSE with noise power for velocity estimate.

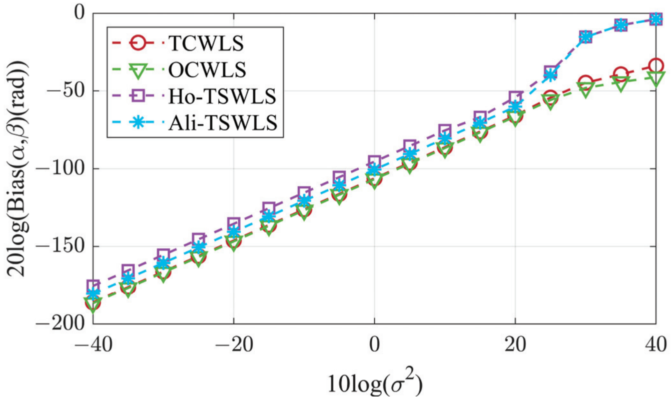


Figure 14. Variation in bias with noise power for DOA estimate.

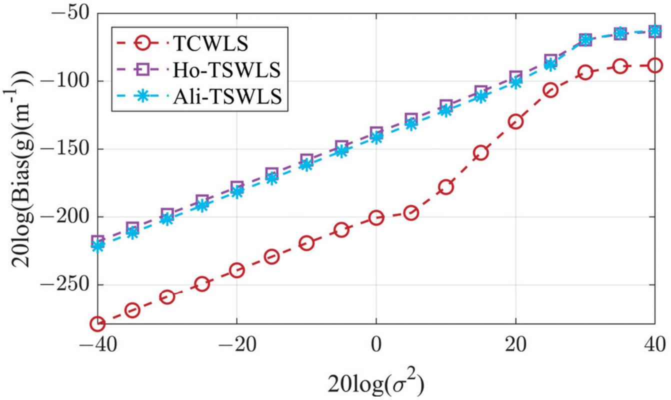


Figure 15. Variation in bias with noise power for inverse-range estimate.

Next, the target parameters are transformed from MRP to a Cartesian coordinate system to compare the MSE and bias of the target location. Figure 17 shows that the performance of all algorithms is consistent when $10\log(\sigma^2) \leq 20$. However, the MSE estimates of the position of Ho-TSWLS and Ali-TSWLS are smaller when $10\log(\sigma^2) > 20$. The reason why the MSE of the proposed algorithm suddenly becomes larger was explained in Section 5.1 and will not be repeated here. Figure 18 compares the bias of all algorithms for position estimation and shows that TCWLS is the best when $10\log(\sigma^2) < 35$.

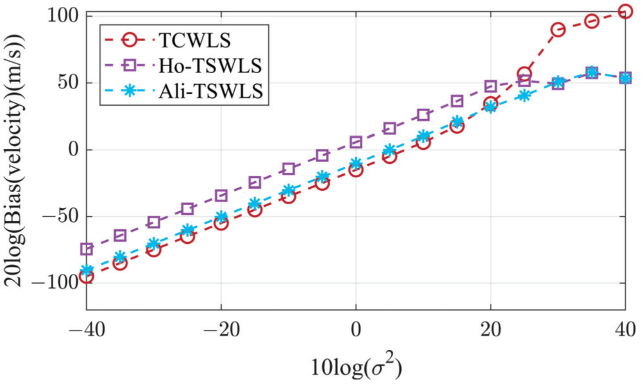


Figure 16. Variation in bias with noise power for velocity estimate.

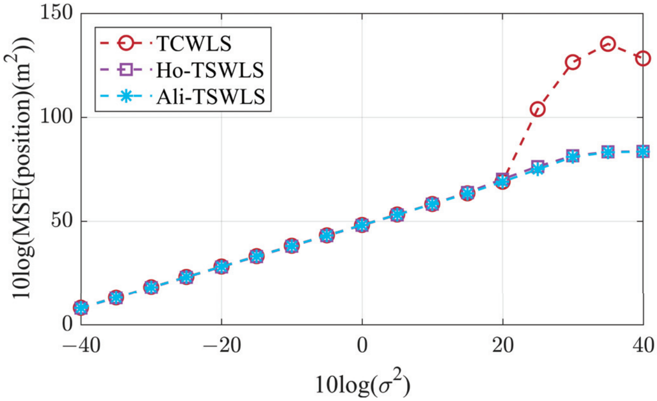


Figure 17. Variation in MSE with noise power for position estimate.

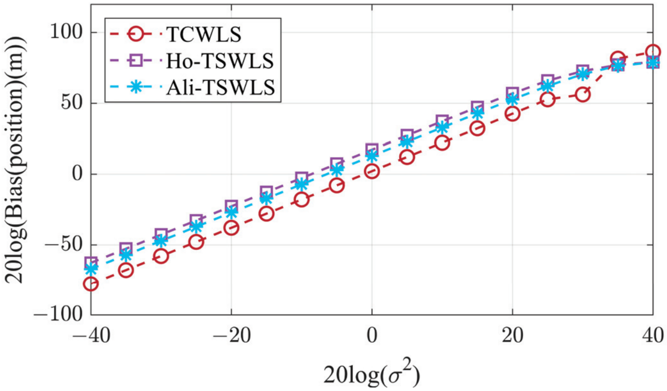


Figure 18. Variation in bias with noise power for position estimate.

5.3. Computation Time

Finally, to compare the complexity of the algorithms, we calculated the average running time of several algorithms. Our simulation processing platform consists of a typical PC harnessing the power of an Intel i7-10700 CPU and a substantial 32.0 GB RAM.

The equipment was sourced from Lenovo, in Beijing, China. It employs MATLAB R2021b as its software environment for executing the simulation processes. The simulation scenario is the one in Section 5.2, and the running time is shown in Table 2. Since several algorithms are closed-form solution algorithms, the running times are in the same order of magnitude, and the proposed algorithm has a slightly higher running time than Ho-TSWLS and Ali-TSWLS for the estimation of a moving target.

Table 2. Average running time.

Method	TCWLS	Ho-TSWLS	Ali-TSWLS
Time (ms)	0.50	0.31	0.27

6. Discussion

This paper proposes a new moving target positioning method in three-dimensional space. The algorithm aims to locate the target without prior information of distance. The positions of the sensors in this paper are picked uniformly and randomly in three-dimensional space without having to maintain a certain configuration to be effectively localized. This can be successfully applied in complex scenarios.

From the simulation results, it can be concluded that under low noise power conditions, TCWLS can achieve the CRLB with a lower bias than the near-field Ho-TSWLS and Ali-TSWLS. However, it should be acknowledged that TCWLS first deviates from the CRLB for the velocity estimate as the noise power increases. This is the direction we need to continue to study in the future. In addition, we need to be aware that small perturbations in the distance inverse estimate can cause large distance and position estimation errors in far-field or strongly noisy power situations. As shown in Equation (46), if the azimuth or elevation angles are at certain specific angles, such as $\alpha^\circ = 90$ deg, this can cause the weighting matrix $\mathbf{W}_2$ to become a singular matrix affected by $\mathbf{B}_2$, ultimately leading to poor positioning performance. This can be avoided by selecting a different sensor as the reference sensor.

TCWLS can also be used in two-dimensional space by simplifying the equations. Furthermore, it can also be used with slight modifications if sensor parameter errors need to be considered.

7. Conclusions

We propose a new positioning algorithm that does not require prior distance knowledge of the moving target. For a near-field target, we can accurately estimate the location and velocity of the target. For a far-field target, we can accurately estimate the DOA of the target. The method first builds a unified model using MPR, then creates an optimization problem with two constraints, and finally obtains an exact closed-form solution via WLS and an optimization algorithm. Simulation experiments validate that the proposed unified localization approach is superior to the traditional near-field moving target localization method. Our future work will focus on improving the robustness of the proposed algorithm in strong noise environments.

Author Contributions: Conceptualization, W.G. and X.S.; methodology, W.G. and X.S.; software, W.G. and X.S.; validation, W.G. and X.S.; formal analysis, W.G. and C.Z.; investigation, C.Z. and Q.W.; resources, W.G.; data curation, W.G. and C.Z.; writing—original draft preparation, W.G. and X.S.; writing—review and editing, Q.W. and Y.L.; visualization, X.S.; supervision, Q.W.; project administration, Q.W.; funding acquisition, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant numbers 62171337, 62201434, 62101396, and 62301391, and supported by the Postdoctoral Fellowship Program of CPSF under Grant Number GZC20233512.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors would like to thank all editors and reviewers for their valuable comments and suggestions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Weinstein, E. Optimal source localization and tracking from passive array measurements. *IEEE Trans. Acoust. Speech Signal Process.* **1982**, *30*, 69–76. [CrossRef]
2. Rappaport, T.S.; Reed, J.H.; Woerner, B.D. Position location using wireless communications on highways of the future. *IEEE Commun. Mag.* **1996**, *34*, 33–41. [CrossRef]
3. Flückiger, M.; Neild, A.; Nelson, B.J. Optimization of receiver arrangements for passive emitter localization methods. *Ultrasonics* **2012**, *52*, 447–455. [CrossRef]
4. Indelman, V.; Gurfil, P.; Rivlin, E. Real-Time Vision-Aided Localization and Navigation Based on Three-View Geometry. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 2239–2259. [CrossRef]
5. Le, T.-K.; Ono, N. Closed-Form and Near Closed-Form Solutions for TDOA-Based Joint Source and Sensor Localization. *IEEE Trans. Signal Process.* **2017**, *65*, 1207–1221. [CrossRef]
6. Chan, Y.T.; Tsui, W.Y.; So, H.C.; Ching, P.C. Time-of-Arrival Based Localization Under NLOS Conditions. *IEEE Trans. Veh. Technol.* **2006**, *55*, 17–24. [CrossRef]
7. Junyang, S.; Molisch, A.F.; Salmi, J. Accurate Passive Location Estimation Using TOA Measurements. *IEEE Trans. Wireless Commun.* **2012**, *11*, 2182–2192. [CrossRef]
8. Nguyen, N.H.; Dogancay, K. Optimal Geometry Analysis for Multistatic TOA Localization. *IEEE Trans. Signal Process.* **2016**, *64*, 4180–4193. [CrossRef]
9. Chan, Y.T.; Ho, K.C. A simple and efficient estimator for hyperbolic location. *IEEE Trans. Signal Process.* **1994**, *42*, 1905–1915. [CrossRef]
10. Xu, E.; Ding, Z.; Dasgupta, S. Source Localization in Wireless Sensor Networks From Signal Time-of-Arrival Measurements. *IEEE Trans. Signal Process.* **2011**, *59*, 2887–2897. [CrossRef]
11. Ho, K.C. Bias Reduction for an Explicit Solution of Source Localization Using TDOA. *IEEE Trans. Signal Process.* **2012**, *60*, 2101–2114. [CrossRef]
12. Malanowski, M.; Kulpa, K. Two_Methods_for_Target_Localization_in_Multistatic_Passive_Radar. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 572–580. [CrossRef]
13. Dogancay, K. Emitter localization using clustering-based bearing association. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 525–536. [CrossRef]
14. Wang, Y.; Ho, K.C. An Asymptotically Efficient Estimator in Closed-Form for 3-D AOA Localization Using a Sensor Network. *IEEE Trans. Wireless Commun.* **2015**, *14*, 6524–6535. [CrossRef]
15. Zheng, Y.; Sheng, M.; Liu, J.; Li, J. Exploiting AoA Estimation Accuracy for Indoor Localization: A Weighted AoA-Based Approach. *IEEE Wireless Commun. Lett.* **2019**, *8*, 65–68. [CrossRef]
16. Wang, G.; Yang, K. A New Approach to Sensor Node Localization Using RSS Measurements in Wireless Sensor Networks. *IEEE Trans. Wireless Commun.* **2011**, *10*, 1389–1395. [CrossRef]
17. Feng, C.; Au, W.S.A.; Valaee, S.; Tan, Z. Received-Signal-Strength-Based Indoor Positioning Using Compressive Sensing. *IEEE Trans. Mob. Comput.* **2012**, *11*, 1983–1993. [CrossRef]
18. Ho, K.C.; Xu, W. An Accurate Algebraic Solution for Moving Source Location Using TDOA and FDOA Measurements. *IEEE Trans. Signal Process.* **2004**, *52*, 2453–2463. [CrossRef]
19. Yu, H.; Huang, G.; Gao, J.; Liu, B. An Efficient Constrained Weighted Least Squares Algorithm for Moving Source Location Using TDOA and FDOA Measurements. *IEEE Trans. Wireless Commun.* **2012**, *11*, 44–47. [CrossRef]
20. Qu, X.; Xie, L.; Tan, W. Iterative Constrained Weighted Least Squares Source Localization Using TDOA and FDOA Measurements. *IEEE Trans. Signal Process.* **2017**, *65*, 3990–4003. [CrossRef]
21. Noroozi, A.; Oveis, A.H.; Hosseini, S.M.; Sebt, M.A. Improved Algebraic Solution for Source Localization from TDOA and FDOA Measurements. *IEEE Wireless Commun. Lett.* **2018**, *7*, 352–355. [CrossRef]
22. Sun, Y.; Ho, K.C.; Wan, Q. Solution and Analysis of TDOA Localization of a Near or Distant Source in Closed Form. *IEEE Trans. Signal Process.* **2019**, *67*, 320–335. [CrossRef]
23. Wang, Y.; Ho, K.C. TDOA Positioning Irrespective of Source Range. *IEEE Trans. Signal Process.* **2017**, *65*, 1447–1460. [CrossRef]
24. Sun, Y.; Ho, K.C.; Wang, G.; Chen, H.; Yang, Y.; Chen, L.; Wan, Q. Computationally Attractive and Location Robust Estimator for IoT Device Positioning. *IEEE Internet Things J.* **2022**, *9*, 10891–10907. [CrossRef]
25. Wang, G.; Ho, K.C. Convex Relaxation Methods for Unified Near-Field and Far-Field TDOA-Based Localization. *IEEE Trans. Wireless Commun.* **2019**, *18*, 2346–2360. [CrossRef]
26. Wang, Y.; Ho, K.C. Unified Near-Field and Far-Field Localization for AOA and Hybrid AOA-TDOA Positionings. *IEEE Trans. Wireless Commun.* **2018**, *17*, 1242–1254. [CrossRef]
27. Sun, Y.; Ho, K.C.; Wan, Q. Eigenspace Solution for AOA Localization in Modified Polar Representation. *IEEE Trans. Signal Process.* **2020**, *68*, 2256–2271. [CrossRef]

28. Chen, X.; Wang, G.; Ho, K.C. Semidefinite relaxation method for unified near-Field and far-Field localization by AOA. *Signal Process.* **2021**, *181*, 107916. [CrossRef]
29. Wang, G.; Xiao, Y.; Ho, K.C.; Huang, L. Unified Near-Field and Far-Field TDOA Source Localization Without the Knowledge of Signal Propagation Speed. *IEEE Trans. Commun.* **2024**, *72*, 2166–2181. [CrossRef]
30. Zhang, F.; Sun, Y.; Zou, J.; Zhang, D.; Wan, Q. Closed-Form Localization Method for Moving Target in Passive Multistatic Radar Network. *IEEE Sensors J.* **2020**, *20*, 980–990. [CrossRef]
31. Kay, S.M. *Fundamentals of Statistical Signal Processing: Estimation Theory*, 1st ed.; Prentice-Hall: Englewood Cliffs, NJ, USA, 1993.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

Cruise Speed Model Based on Self-Attention Mechanism for Autonomous Underwater Vehicle Navigation

Xiaokai Mu [1,2], Yuanhang Yi [3], Zhongben Zhu [3,*], Lili Zhu [3], Zhuo Wang [1] and Hongde Qin [1]

[1] Key Laboratory of Autonomous Marine Vehicle Technology, Harbin Engineering University, Harbin 150001, China; muxiaokai@hrbeu.edu.cn (X.M.); wangzhuo@hrbeu.edu.cn (Z.W.); qinhongde@hrbeu.edu.cn (H.Q.)
[2] Qingdao Innovation and Development Center, Harbin Engineering University, Qingdao 266000, China
[3] Qingdao Innovation and Development Base, Harbin Engineering University, Qingdao 266000, China; yiyuanhang@hrbeu.edu.cn (Y.Y.); lilizhu@hrbeu.edu.cn (L.Z.)
* Correspondence: zhuzhongben@hrbeu.edu.cn

Abstract: This study proposes a cruise speed model based on the Self-Attention mechanism for speed estimation in Autonomous Underwater Vehicle (AUV) navigation systems. By utilizing variables such as acceleration, angle, angular velocity, and propeller speed as inputs, the Self-Attention mechanism is constructed using Long Short-Term Memory (LSTM) for handling the above information, enhancing the model's accuracy during persistent bottom-track velocity failures. Additionally, this study introduces the water-track velocity information to enhance the generalization capability of the network and improve its speed estimation accuracy. The sea trial experiment results indicate that compared to traditional methods, this model demonstrates higher accuracy and reliability with both position error and velocity error analysis when the used Pathfinder DVL fails, providing an effective solution for AUV combined navigation systems.

Keywords: AUV; navigation; deep learning; speed estimation; self-attention mechanism

Citation: Mu, X.; Yi, Y.; Zhu, Z.; Zhu, L.; Wang, Z.; Qin, H. Cruise Speed Model Based on Self-Attention Mechanism for Autonomous Underwater Vehicle Navigation. *Remote Sens.* **2024**, *16*, 2580. https://doi.org/10.3390/rs16142580

Academic Editors: Giuseppe Casula, Zhetao Zhang, Guorui Xiao, Zhixi Nie and Vagner Ferreira

Received: 30 May 2024
Revised: 7 July 2024
Accepted: 12 July 2024
Published: 14 July 2024

1. Introduction

Recently, interest in marine resources has grown considerably, resulting in increased marine development activities. Autonomous Underwater Vehicles (AUVs) are crucial for tasks such as seabed resource exploration, submarine pipeline maintenance, and marine data collection [1,2]. Therefore, obtaining precise navigation and positioning technology for AUVs is crucial to ensuring successful and timely task completion, owing to the highly complex marine environment. In contrast to land robots [3] and aerial robots [4], AUVs do not receive GPS signals underwater, posing a challenge for traditional satellite-dependent navigation techniques in this environment. Emerging technologies have been increasingly employed recently for successful underwater localization and navigation. The primary underwater navigation and localization techniques are categorized into four main groups: acoustic navigation [5–7], geophysical navigation [8–10], Simultaneous Localization and Mapping (SLAM), and inertial navigation and dead reckoning [11,12]. Acoustic waves are the most effective method for transmitting information underwater, making acoustic navigation the primary method for underwater target navigation and localization. Nonetheless, the acoustic beacons must be placed in advance, as acoustic navigation is ineffective in an unknown environment. Geophysical navigation can be divided into three primary groups based on the requisite geophysical parameters: terrain-matching navigation, marine geomagnetic navigation, and gravity navigation. However, geophysical navigation is limited by the requirement to obtain geophysical parameters in advance. Conversely, SLAM enables AUVs to create maps of their surroundings and determine their position within that environment. However, SLAM requires external environmental information measured by additional sensors and high computation capacity.

Inertial navigation is an autonomous system known for not relying on external information or emitting energy externally. The Inertial Navigation System (INS) uses triaxial gyroscopes and accelerometers to measure angular rate and acceleration. Then, the attitude, velocity, and position information of the AUV is calculated by an integral operation. However, the integration process results in error accumulation in the INS, and over a long navigation period, the position can be shifted considerably. This approach partially mitigates the error accumulation problem by utilizing a Doppler Velocity Logger (DVL) for bottom-track velocity and integrating INS and DVL measurements. Kalman filtering (KF) is a widely applied data fusion method [13], and it can achieve optimal filtering with Gaussian white noise in the system process. The bottom-track velocities measured by the DVL are indispensable in the data fusion algorithm. However, the DVL is sensitive to the complex marine environment, which may cause inaccuracies in velocity measurements. For instance, DVL bottom tracking can be vulnerable to interference and disruptions due to steep seafloor slopes or rifts, AUV attitude, currents, and fish populations [14], as shown in Figure 1. In cases where the DVL produces anomalous values for a brief period, the issue can be resolved by utilizing effective bottom tracking from the previous moment. Nevertheless, this method is inadequate when the DVL outputs anomalous data for an extended period or is inactive, in which case the INS solution error accumulates and the navigation accuracy significantly degrades over time. Consequently, investigating the navigation method when the DVL output remains invalid for an extended period is crucial.

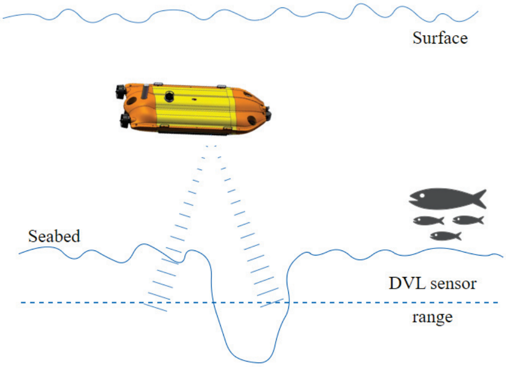


Figure 1. Vulnerability to bottom-track interference.

Some methods have been commonly used in the existing literature to address invalid bottom tracking [15,16]: One approach involves implementing combined navigation by installing additional sensors to replace the DVL in case of failure; however, this method increases costs and system complexity. Conversely, another method replaces the DVL with a mathematical model generating virtual bottom velocity information, solved by way of modeling single- and three-degree-of-freedom dynamics [17]. And Kinsey et al. developed a single-degree-of-freedom nonlinear dynamic model estimator and verified its feasibility [18]. Zhao et al. introduced a mechanism for outlier detection in DVL data and compensated for velocity anomalies using a kinematic model. However, the complexity of AUV models in challenging marine environments makes it difficult to obtain accurate hydrodynamic parameters. Therefore, building precise AUV dynamic models is evidently impractical. Establishing dynamic models with single and three degrees of freedom, validated through sea trials, demonstrated that the speeds calculated using these models closely aligned with those measured by the DVL.

Various machine learning algorithms, such as Support Vector Machines (SVMs) [19], Random Forests (RFs) [20], Extreme Learning Machines (ELMs) [21], and Artificial Neural Networks (ANNs) [22], have been employed in diverse fields owing to the recent widespread application of artificial intelligence technology. In their study, Mu et al. [23]

applied the time-series learning mechanism to AUV navigation and proposed a novel neural network framework using Long Short-Term Memory (LSTM) to process multi-sensor data and determine the position of an AUV during navigation. Another study [19] developed a hybrid predictor by combining partial least squares regression and support vector regression to estimate the bottom velocity of a DVL when faced with DVL failure. Lv et al. employed ELM to establish a model relationship between the AUV's thruster speed, attitude, rudder information, and bottom velocity to compensate for DVL failures. Li et al. proposed a nonlinear autoregressive framework with heteroscedastic inputs (NARX) and adaptive Kalman filtering to predict and fuse DVL outputs. Water-track velocity and flow rate estimation during anomalous DVL bottom velocity were also investigated [24]. Our study presents a deep learning framework incorporating LSTM and Self-Attention to address this issue, considering the current velocity as a variable to estimate the water-track velocity of the DVL. The effectiveness of our approach is validated by comparing the results with the measured data.

This paper proposes a cruise speed model based on the Self-Attention mechanism for estimating AUV speeds in complex marine environments. Utilizing inputs like acceleration, angle, angular velocity, and propeller speed, the model estimates cruise speed via the Self-Attention mechanism. This cruise speed corresponds to the velocities along the three axes of the AUV onboard coordinate system. As a consequence, the model sustains high navigation accuracy even when the bottom-track velocity data are consistently unavailable. The main contributions of this paper are as follows:

(1) To address the continuous failure of bottom-track velocity measurements in complex marine environments, a deep learning-based AUV speed estimation model is constructed to predict and output bottom-track velocities, enhancing AUV navigation accuracy during DVL failures.

(2) LSTM will be used to separately extract time-series data from different data sources, and Self-Attention will be employed to enhance the encoding of time-series data. Water flow rate information is introduced into the network as input to compensate for ocean current information, increasing the model's generalization capability.

(3) The proposed Self-Attention-based cruise speed model's effectiveness on AUVs will be validated through sea trials and simulation data. The results show that the proposed model achieves better navigation accuracy compared to using water-track velocity compensation.

The rest of this paper is organized as follows: Section 2 describes the AUV and equipment specifications used for the field trials. Section 3 derives a model for the application of Kalman filtering in combined AUV navigation. Section 4 details the network model framework and analyzes the results obtained in Section 5. Finally, Section 6 concludes the study.

2. An Introduction of the AUV Platform

Herein, we present the AUV used in our experiment, depicted in Figure 2. The XH R300 employs a double main thrust propulsion system capable of attaining a maximum speed of 5 knots and sustaining continuous travel for up to 10 km. The hydrodynamic characteristics of the XH R300 are notably intricate, necessitating the formulation of a three-degree-of-freedom dynamics model to elucidate its motion. This modeling endeavor is predicated on several key assumptions: first, the AUV is treated as a rigid body; second, the current is assumed to be a two-dimensional flow lacking rotational components; and third, the fluid medium is regarded as uniform and unbounded. The kinetic equations governing the AUV's motion are conventionally expressed as follows:

$$M\dot{v} + C(v)v + D(v)v = f \tag{1}$$

where v denotes the triaxial component of the AUV velocity in the carrier coordinate system, M and C, respectively, denote the inertia matrix and the Coriolis centripetal matrix of the rigid body, $f = [(\tau)_X, (\tau)_Y, (\tau)_N]$ are external forces and moments, $(\tau)_X$ and $(\tau)_Y$ are, respectively, the axial and lateral forces acting on the AUV, and $(\tau)_N$ is the yaw external moment. The expression is as follows:

$$f = \begin{bmatrix} \tau_X \\ \tau_Y \\ \tau_N \end{bmatrix} = \begin{bmatrix} T_{port} + T_{stbd} \\ 0 \\ \left(T_{port} - T_{stbd}\right)B/2 \end{bmatrix} \tag{2}$$

where T_{port} and T_{stbd} are, respectively, the thrust of the port and starboard thrusters, and B is the distance between the thrusters. The three-degree-of-freedom nonlinear dynamics model of the AUV can be described as

$$\begin{aligned} &\tau_X = (m - X_{\dot{u}})\dot{u} - m\left(x_G r^2 + vr\right) + Y_{\dot{v}} vr + \frac{Y_{\dot{r}} + N_{\dot{v}}}{2} r^2 + X_u u + X_{|u|u}|u|u \\ &\tau_Y = (m - Y_{\dot{v}})\dot{v} + (m x_G - Y_{\dot{r}})\dot{r} + (m - X_{\dot{u}})ur + Y_v v + Y_r r + Y_{|v|v}|v|v + Y_{|r|r}|r|r \\ &\tau_N = (m x_G - N_{\dot{v}})\dot{v} - (I_{zz} - N_{\dot{r}})\dot{r} + m x_G ur - Y_{\dot{v}} uv - \frac{Y_{\dot{r}} + N_{\dot{v}}}{2} ur + X_{\dot{u}} uv + N_v v \\ &+ N_r r + N_{|v|v}|v|v + N_{|r|r}|r|r \end{aligned} \tag{3}$$

where $X_{(\cdot)}$, $Y_{(\cdot)}$, and $N_{(\cdot)}$ represent hydrodynamic coefficients. According to Equations (2) and (3), the AUV speed is related to the acceleration, angle, angular velocity, and amount of rudder thrust. The thrust of the servos, in turn, is related to the rotational speed and current obtained through various sensor measurements, which will be used later in this study to estimate the AUV speed. The equipment used to obtain the relevant data is illustrated below.

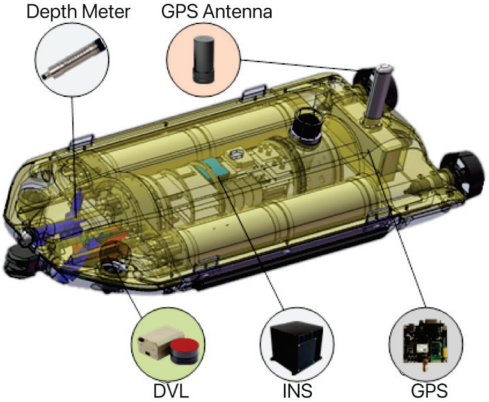


Figure 2. Basic structure of XH R300.

The XH R300 is equipped with a signal cabin, control cabin, power control cabin, and power operation cabin. The primary sensors include a GPS module, Iridium satellite, radio, Wifi, INS, DVL, and depth gauge to obtain AUV position, acceleration, angle, and angular velocity information. Based on functionality, the main control can be divided into a control unit, navigation and positioning unit, guidance and planning unit, perception unit, fault detection unit, and data storage unit. The navigation and positioning unit is crucial for real-time acquisition of AUV pose information and provides the foundational support for the operation of the control unit and guidance and planning unit. The GPS module offers real-time precise latitude and longitude data while the AUV operates on the water surface, as delineated in Table 1. Nevertheless, owing to the rapid attenuation of GPS signals in water, the XH R300 incorporates the INS (detailed in Table 2) that derives the AUV's position, velocity, and triaxial attitude angle by integrating data from the gyroscope, measuring angular rates, and the accelerometer, gauging triaxial accelerations. However, the integration process inevitably results in error accumulation within the INS, impinging upon navigation accuracy. Consequently, the XH R300 is outfitted with a Pathfinder 600 KHz DVL developed by Teledyne, described in Table 3, to rectify these discrepancies. The DVL emits sound waves via a transducer when a phased array is employed, which, upon reaching the seabed, bounce back, enabling velocity estimation relative to the seafloor by analyzing frequency shifts in the received echoes. When GPS signals are unavailable underwater, the disparity between the raw speed of the INS and the speed of the DVL serves as feedback, refining the INS output through an indirect approach.

Table 1. GPS module specifications.

Equipment Type	GPS Module
Single-point positioning accuracy	<1.5 m
Velocimetry accuracy	0.03 m/s
Maximum data update frequency	1 Hz

Table 2. INS specifications.

Equipment Type	INS
Heading accuracy	0.5°
Attitude accuracy	0.02°
Gyro accuracy	0.05°/h (1σ)
Plus meter accuracy	200 µg (1σ)

Table 3. DVL specifications.

Equipment Type	DVL
Maximum height	89 m
Minimum height	0.2 m
Speed range	±9 m/s
Resolution	0.1 cm/s
Pinging frequency	12 Hzmax

However, in deep-sea environments exceeding the operational range of the DVL or encountering steep seabed inclines, the acoustic waves of the DVL may fail to reach or be detected upon seabed contact, rendering the bottom-track data invalid and precluding its integration with INS for high-precision navigation. Although the DVL can also provide water-track velocities, they are notably less precise than bottom-track velocities and fail to meet stringent navigation accuracy requisites. A novel solution addressing these challenges is proposed herein and elaborated upon subsequently. Additionally, a depth gauge ISD4000 developed by Impact Subsea is integrated into the XH R300 for precise depth determination, ensuring accurate depth measurement.

3. AUV Combined Navigation Model

The combined AUV navigation model outlined in this section primarily relies on integrating the INS and DVL systems. Initially, the INS error model is used to formulate the state equations of the integrated navigation system. Subsequently, data gathered by the INS facilitate AUV motion prediction. Observations from the INS and DVL are then incorporated into the model, refining its predictions to align more closely with actual values. This iterative process, conducted in the time domain, culminates in achieving combined AUV navigation. This section details the model construction process and the updated predicted values, which constitute a pivotal aspect of the process.

Model Construction

This study employs two coordinate systems: the navigation and carrier coordinate systems. The navigational coordinate system, denoted as $O - X^n Y^n Z^n$, situates its origin at sea level, with the OX^n axis pointing northward, the OY^n axis eastward, and the OZ^n axis directed toward the geocentric North-East Earth (NEU) geographical coordinates. Conversely, the carrier coordinate system, denoted as $O - X^bY^bZ^b$, positions its origin at the center of gravity of the AUV, with the OX^b axis directed forward, the OY^b axis starboard, and the OZ^b axis downward. The navigational coordinate system undergoes three rotational transformations with respect to the carrier coordinate system: the heading angle α around the OZ^n axis, the pitch angle β around the OY^n axis, and the roll angle γ around the OX^n axis. Typically, instruments are situated in the carrier coordinate system. Therefore, to determine the AUV's absolute position in the navigation coordinate system, the various AUV states are multiplied by its rotation matrix C_b^n, defined as follows:

$$C_b^n = \begin{bmatrix} cos\gamma cos\alpha - sin\gamma sin\beta sin\alpha & -sin\alpha cos\beta & cos\gamma sin\alpha + sin\alpha sin\beta sin\gamma \\ sin\gamma cos\alpha + cos\gamma sin\beta sin\alpha & cos\beta cos\alpha & sin\gamma sin\alpha - cos\gamma sin\beta cos\alpha \\ -cos\beta sin\gamma & sin\beta & cos\gamma cos\beta \end{bmatrix} \tag{4}$$

In the navigation system, the navigation parameter errors of INS are selected as state variables. Due to the generally small error values, the state equation can be considered a first-order linear system, with the speed difference between INS and DVL used as the measurement variable. The error amount is then optimally estimated through standard Kalman filtering to feedback and correct the INS output. Leveraging the INS error model, the state vectors are identified as 15-dimensional error quantities of the INS, including attitude error ϕ_E, ϕ_N, ϕ_U, velocity error $\delta v_E, \delta v_N, \delta v_U$, position error $\delta p_E, \delta p_N, \delta p_U$, gyro zero bias $\varepsilon_x, \varepsilon_y, \varepsilon_z$, and accelerometer zero bias $\nabla_x, \nabla_y, \nabla_z$.

$$X = \left[\phi_E, \phi_N, \phi_U, \delta v_E, \delta v_N, \delta v_U, \delta p_E, \delta p_N, \delta p_U, \varepsilon_x, \varepsilon_y, \varepsilon_z, \nabla_x, \nabla_y, \nabla_z\right]^{\mathrm{T}} \tag{5}$$

The state equation of the system is as follows:

$$\dot{X} = FX + W \tag{6}$$

where F is the state transfer matrix and W is the state noise. The system error propagation equation is shown as follows:

$$\left\{\begin{array}{l}
\dot{\delta} L = -\frac{V_N}{(R_M+h)^2}\delta h + \frac{1}{R_M+h}\delta V_N \\
\dot{\delta}\lambda = \frac{V_E \tan L \sec L}{R_N+h}\delta L - \frac{V_E \sec L}{(R_N+h)^2}\delta h + \frac{\sec L}{R_N+h}\delta V_E \\
\dot{\delta} h = \delta V_U \\
\dot{\delta} V_E = (2\Omega V_N \cos L + \frac{V_E V_N \sec^2 L}{R_N+h} + 2\Omega V_U \sin L)\delta L + \frac{V_E V_U - V_N V_E \tan L}{(R_N+h)^2}\delta h \\
+ \frac{V_N \tan L - V_U}{R_N+h}\delta V_E + (2\Omega \sin L + \frac{V_E \tan L}{R_N+h})\delta V_N - (2\Omega \cos L + \frac{V_E}{R_N+h})\delta V_U \\
+ f_N \varphi_U - f_U \varphi_N + \nabla_E \\
\dot{\delta}\ V_N = -\ (2\Omega V_E \cos L + \frac{V_E \sec^2 L}{R_N+h})\delta L + (\frac{V_E \tan^2 L}{(R_N+h)^2} + \frac{V_N V_U}{(R_M+h)^2})\delta h \\
-(2\Omega \sin L + \frac{V_E \tan L}{R_N+h})\delta V_E - \frac{V_U}{R_M+h}\delta V_N - \frac{V_N}{R_M+h}\delta V_U \\
+ f_U \varphi_E - f_E \varphi_U + \nabla_N \\
\dot{\delta}\ V_U = -\ 2\Omega V_E \sin L \delta L - \left[\frac{V_E^2}{(R_N+h)^2} + \frac{V_N^2}{(R_M+h)^2}\right]\delta h + (2\Omega \cos L + \frac{V_E}{R_N+h})\delta V_E \\
+ \frac{V_N}{R_M+h}\delta V_N + f_E \varphi_N - f_N \varphi_E + \nabla_U \\
\dot{\varphi}_E = \frac{V_N}{(R_M+h)^2}\delta h - \frac{1}{R_M+h}\delta V_N + (\Omega \sin L + \frac{V_E}{R_N+h}\tan L)\varphi_N \\
-(\Omega \cos L + \frac{V_E}{R_N+h})\varphi_U - \varepsilon_E \\
\dot{\varphi}_N = -\Omega \sin L \delta L + \frac{V_E}{(R_N+h)^2}\delta h + \frac{1}{R_N+h}\delta v_E - (\Omega \sin L + \frac{V_E}{R_N+h}\tan L)\varphi_E \\
- \frac{V_N}{R_M+h}\varphi_U - \varepsilon_N \\
\dot{\varphi}_U = (\Omega \cos L + \frac{V_E}{R_N+h}\sec^2 L)\delta L - \frac{V_E \tan L}{(R_N+h)^2}\delta h + \frac{\tan L}{R_N+h}\delta V_E \\
+(\Omega \cos L + \frac{V_E}{R_N+h})\varphi_E + \frac{V_N}{R_M+h}\varphi_N - \varepsilon_U
\end{array}\right. \tag{7}$$

The measurement equation for combined navigation is as follows:

$$Z = HX + V \tag{8}$$

where H is the measurement matrix and V is the measured noise.

The error equation for the DVL in the navigation coordinate system is as follows:

$$
\begin{aligned}
\tilde{V}{}^{n}_{dvl} &= \tilde{C}{}^{n}_{b} C^{b}_{d} V^{d}_{dvl} \\
&= [I - \psi\times] C^{n}_{b} C^{b}_{d} V^{d}_{dvl} \\
&= V^{n}_{dvl} - \psi \times V^{n}_{dvl} \\
&= V^{n}_{dvl} + V^{n}_{dvl} \times \psi \\
\delta V^{n}_{d} &= \tilde{V}{}^{n}_{dvl} - V^{n}_{dvl} = V^{n}_{dvl} \times \psi
\end{aligned}
\tag{9}
$$

where C^{b}_{d} denotes the rotation matrix from the DVL instrument coordinate system to the carrier coordinate system. We take the difference between the SINS and DVL velocities as the measure and construct the measure model according to the error model of the DVL as follows:

$$
\begin{aligned}
Z &= \tilde{V}{}^{n}_{b} - \tilde{V}{}^{n}_{d} = (V^{n} + \delta V^{n}_{b}) - (V^{n} + \delta V^{n}_{d}) \\
&= \delta V^{n}_{b} - \delta V^{n}_{d} = \delta V^{n}_{b} - V^{n}_{d} \times \psi \\
&= HX + V
\end{aligned}
\tag{10}
$$

The specific form of H is as follows:

$$
H = \begin{bmatrix}
0 & V^{n}_{d}{}_{U} & -V^{n}_{d}{}_{N} & 1 & 0 & 0 & \\
-V^{n}_{d}{}_{U} & 0 & V^{n}_{d}{}_{E} & 0 & 1 & 0 & 0_{3\times 9} \\
V^{n}_{d}{}_{N} & -V^{n}_{d}{}_{E} & 0 & 0 & 0 & 1 &
\end{bmatrix}
\tag{11}
$$

where $V^{n}_{d}{}_{E}$, $V^{n}_{d}{}_{N}$, and $V^{n}_{d}{}_{U}$, respectively, denote the triaxial components of the DVL-measured velocity in the geographic coordinate system. Here, the combined SINS/DVL navigation model construction is completed.

4. Deep Learning Navigation Architecture

The AUV state data, captured as a time series, exhibit significant correlations over time. Previous studies on DVL anomalies often treated sensor data at each moment in isolation, neglecting the time-series correlations. Furthermore, not all data points are equally important in predicting subsequent states. In response to these considerations, this section presents a detailed description of a novel deep learning network architecture, developed after comprehensively examining these two aspects.

4.1. Basic LSTM Principles

Deep learning has recently emerged as a ubiquitous tool across various domains, with researchers continuously introducing new network architectures that demonstrate remarkable performance in practical applications. Among these architectures, Recurrent Neural Networks (RNNs) have found widespread use in tasks involving time-series prediction and natural language processing, owing to their adeptness in handling sequential data. Given that AUV sensor data inherently represent time-series data, RNNs are a natural choice for AUV navigation tasks. However, conventional RNNs struggle to retain long-term dependencies, with information relevance diminishing as it recedes from the current moment. This limitation stems from the BackPropagation Through Time (BPTT) method employed during training, where gradients associated with distant moments gradually vanish, rendering conventional RNNs inadequate to address long-term dependency issues [25].

LSTM [26] networks were introduced to mitigate the challenge of vanishing gradients and effectively model long-term dependencies. LSTM represents a specialized variant of RNNs explicitly designed to tackle gradient instability encountered when training sequences with long time-series spans. By introducing more gating units to control the information flow within the network, the stability of the parameter optimization process is enhanced. The *Tanh* function is used to extract valid information to alleviate the problem of vanishing gradients in the calculation of memory cells and hidden states. The LSTM architecture, depicted in Figure 3, incorporates memory cells and introduces several gating mechanisms to regulate the flow of information within the network. At each time step, the input X_t from the current moment and the hidden state H_{t-1} from the preceding moment are fed into the LSTM gates, which undergo processing via three fully connected layers equipped with sigmoid activation functions to compute the input, forget, and output gate values. This computation proceeds as follows:

$$\begin{aligned} I_t &= \sigma(X_t W_{xi} + H_{t-1} W_{hi} + b_i) \\ F_t &= \sigma\left(X_t W_{xf} + H_{t-1} W_{hf} + b_f\right) \\ O_t &= \sigma(X_t W_{xo} + H_{t-1} W_{ho} + b_o) \end{aligned} \tag{12}$$

where W_{xc}, W_{xf}, W_{xo} and W_{hi}, W_{hf}, W_{ho} are weight parameters and b_i, b_f, b_o are bias parameters. The candidate memory element $\overset{\sim}{C_t}$ is calculated similarly to the gate but using the *tanh* function as the activation function. Its equation at moment t is as follows:

$$\overset{\sim}{C_t} = tanh(X_t W_{xc} + H_{t-1} W_{hc} + b_c) \tag{13}$$

where W_{xc} and W_{hc} are weight parameters and b_c are bias parameters. Subsequently, the memory cells are computed, utilizing the previously derived input and forget gate values to determine the extent to which new data from candidate memory cells are incorporated while retaining relevant past information. This approach effectively mitigates the issue of vanishing gradients and facilitates capturing relationships with long-term dependencies within the time series. The computation of memory cells can be described as follows:

$$C_t = F_t \odot C_{t-1} + I_t \odot \overset{\sim}{C_t} \tag{14}$$

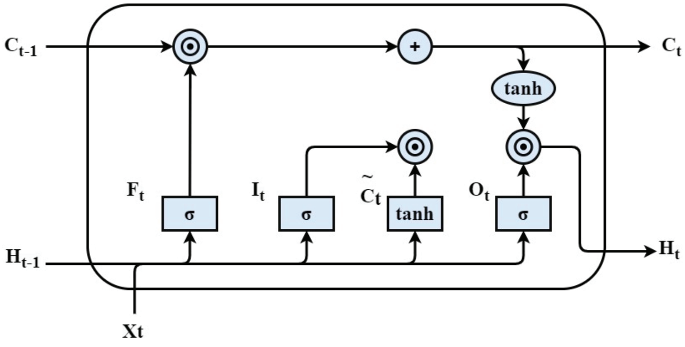


Figure 3. LSTM network structure.

Finally, the hidden state H_t is computed, leveraging the output gate and memory cells. When the output gate is close to 1, it signifies the effective propagation of all memorized information to the prediction phase. Conversely, when the output gate is close to 0, it implies information retention solely within the memory cells without updating the hidden state. This computation unfolds as follows:

$$H_t = O_t \odot tanh(C_t) \tag{15}$$

LSTM has found extensive utility in natural language processing owing to its adeptness in handling long-term dependencies. The proposed model leverages LSTM to process time-series data, with the output of the LSTM layer serving as input to the subsequent attention mechanism layer, as elaborated upon in subsequent sections.

4.2. Self-Attention Mechanism

The Self-Attention mechanism represents a network configuration that comprehensively considers the overall context while prioritizing salient features. In time-series data, the information at any given moment is often interdependent on preceding moments. However, the correlation between data from different moments and the current moment varies. Therefore, during data training, incorporating information from previous moments and emphasizing the most pertinent information is crucial. This is commonly referred to as the Self-Attention mechanism.

The computational process of the Self-Attention mechanism is illustrated in Figure 4. The input H_t is subjected to multiplication by three weight matrices W^Q, W^K, and W^V to derive Q, K, and V, respectively. Subsequently, the resultant Q and K are used to compute the correlation between input vectors α, typically through dot-multiplication. Normalization is then performed using the SoftMax function to obtain A. Finally, A is multiplied by V to yield the output of the Self-Attention mechanism layer.

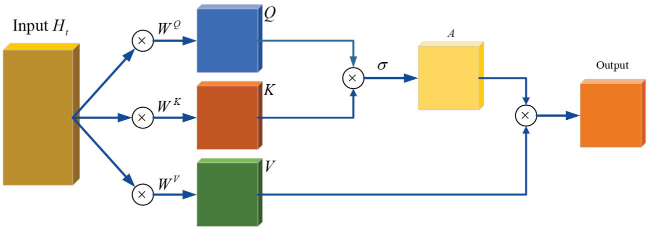


Figure 4. Self-attentive machine architecture.

4.3. The Deep Learning Navigation Framework Based on Self-Attention

In the complex marine environment, the navigation and localization of AUVs predominantly rely on INS and DVL. However, DVL may produce invalid readings under certain conditions, such as encountering a school of fish, resulting in short-term data invalidation. Prolonged DVL invalidity occurs in ultra-deep waters or when encountering steep seabed slopes with no echo returns. While short-term invalidations can be compensated for using kinetic models, relying on such models for extended durations introduces deviations from actual velocities, impeding high-precision navigation and localization.

This section proposes a deep learning navigation framework based on the Self-Attention mechanism to achieve precise navigation over extended periods. The framework adopts an encoder–decoder architecture, organizing sensor data into time-series sequences inputted into the LSTM layer for encoding. Subsequently, time-series data are further refined through the Self-Attention mechanism, followed by decoding through fully connected layers and water-track velocity.

According to the AUV dynamics model outlined in Section 2, the velocity of the AUV correlates with acceleration, angular velocity, angle, thrust [27], and other factors. Acceleration encompasses triaxial acceleration in the instrument coordinate system, while angular velocity includes the triaxial angular velocity of the gyroscope. The angle comprises pitch and roll angles obtained from the INS. Thrust indirectly indicates the speed and current of twin thrusters. Although these data constitute time-series sequences, their sampling frequencies vary among sensors; for example, the collection frequency of the INS is 10, and the collection frequency of the thruster is 2. Although interpolation methods can be used to unify data of different frequencies to a common frequency, models built using this method may cause information increase and loss due to artificial data accumulation or interpolation. Separately processing data of different frequencies can also reduce the data preprocessing process. Additionally, separately processing data from different sources allows the encoder to only encode the data without handling the relationships between data, thus decoupling the network functionally and reducing repetitive work. Hence, data from sensors with different frequencies are inputted into corresponding LSTM layers. As depicted in Figure 5, this framework employs five LSTM layers to receive acceleration, angular velocity, angle, thruster speed, and current information. After extracting and compressing the time-series data of sensors into context vectors through the LSTM layer, the hidden layer serves as the input for further training on data significance at different moments through the Self-Attention mechanism layer. Finally, the Self-Attention mechanism layer output and the DVL-derived bottom-track velocity are fed into the fully connected layer for decoding.

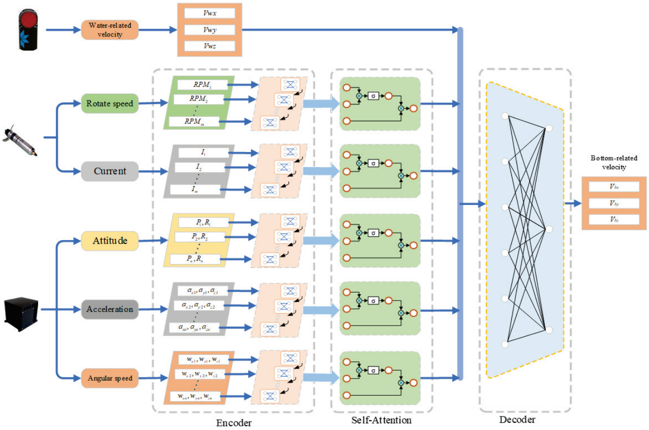


Figure 5. Deep learning navigation framework.

The encoder–decoder architecture decouples the network, reducing redundancy while facilitating input–output sequence correspondence modeling. In the encoder stage, sensor time-series data are compressed into context vectors by LSTM, albeit with inevitable information loss. To address this information loss, a Self-Attention mechanism enhances time-series data encoding, learning correlations between input moments. The input in the decoder stage comprises timing vectors enhanced by the Self-Attention module. Given that the output solely represents the AUV velocity at the current moment without necessitating multiple sequence outputs, a linear layer is employed to map high-dimensional time-series vectors to a low-dimensional sample space, yielding the output of the model. To enhance model generalization during decoding, water-track velocity is encoded by LSTM and combined with timing input, serving as the final input to the linear layer. This enables the model to learn embedded sea current information.

The entire model can be summarized into two categories: First, LSTM and Self-Attention encode the timing information to obtain optimal timing vectors, addressing the long-term dependency problem and extending the inputs to high dimensions to extract effective information in all aspects. Second, the linear layer decoder maps the extracted time-series data to lower dimensions and learns the sea current information and water-track velocities to enhance the generalizability of the model and obtain the optimal output.

4.4. Portfolio Navigation Framework

After constructing the deep learning navigation model based on Self-Attention as described in the previous subsection, the collected INS, DVL, and thruster data are divided into two paths when the DVL operates normally. One input feeds into the combined navigation model for AUV position computation, while the other input trains optimized network parameters for AUV speed estimation. During a short DVL failure, the Pathfinder DVL outputs a valid flag for the bottom-track velocity, where an A flag indicates that the measured bottom-track velocity is valid and any other flag indicates that it is invalid. Consequently, no more fault detection activity is performed, and the combination is navigated by compensating for the speed of the AUV using the water-track velocity. Conversely, during prolonged DVL invalidity, the DVL is determined to be invalid for a long time by calculating the time $t_{invalid}$ since the last valid flag bit. When $t_{invalid}$ is larger than 10 s, the DVL for the bottom-velocity measurement is considered to have been invalid for a long time. At this point, the data from the corresponding sensors are fed into the AUV speed estimation model to predict the current AUV speed, and then the predicted speed is subtracted from the INS speed to obtain the measured value for optimal estimation. During the training and prediction of the AUV speed estimation model, the corresponding sensor data must be saved according to the set time interval. The frequency at which the DVL measures water-track velocity is used and is typically set to 1 s. The specific framework diagram is shown in Figure 6.

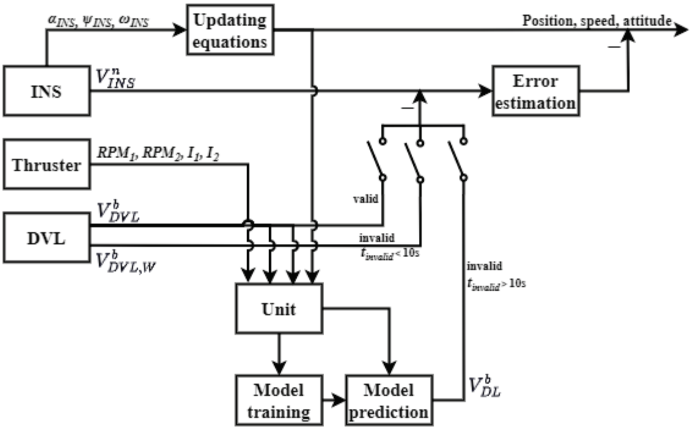


Figure 6. Combined navigation framework during DVL failure.

5. Experimental Results and Discussion

5.1. Test Configuration

To verify the performance of the proposed cruise speed estimation model, a large amount of data was collected at Xuejiadao Wharf, Qingdao, with the XH R300 AUV. The on-site experiment is shown in Figure 7. The data include acceleration, angular velocity, angle, water-track velocity, and bottom-track velocity at each acquisition moment. Additionally, rotational speed and current information are collected for the main thrusts. While bottom-track velocities serve as the target values for training, the remaining data are used as network input to predict velocity. After preprocessing the data, 19,010 pairs of data were selected as the training set and 3140 pairs of data as the validation set. The test set of Experiment 1 contains 4840 pairs of data and the test set of Experiment 2 contains 5340 pairs of data. And the deep learning framework was implemented using Pytorch, with model hyperparameters set as follows: the experiments were executed using NVIDIA GeForce GTX 1050Ti GPU; the time consumption on the training was about 20 ms; the number of nodes in the hidden layer of the LSTM and the output dimension of the Self-Attention layer were both set to 30; when Self-Attention was employed, its head was set to 2, indicating two parallel Self-Attention mechanism layers for extracting timing information from different aspects; the LSTM and Self-Attention dropout layers were set to 0; and the learning rate was set to 0.0001.

Figure 7. On-site experimental diagram.

5.2. *AUV Real Experimental Data Test*

GPS latitude and longitude were not selected as the ground truth values, as the deep-learning speed estimation model references bottom-track velocities. Instead, the trajectory derived from the bottom-track velocities was utilized. AUVs typically navigate in straight lines or execute comb trajectories based on mission requirements; hence, both scenarios were considered in experimental trajectory selection. Figure 8 illustrates a trajectory comparison between different methods, where purple represents trajectories computed using bottom-track velocities, red denotes trajectories computed using water-track velocities, and yellow depicts trajectories computed using velocities estimated by the Deep Learning Model (DLM). In Experiment 1 (Figure 8a), the trajectories exhibit minimal disparity during straight-line navigation. However, during turns, the trajectory derived from the water-track velocity lags, indicating diminished accuracy compared to the trajectory generated using the speed estimation model. Similar observations were noted in Experiment 2 (Figure 8b), particularly during comb trajectory execution.

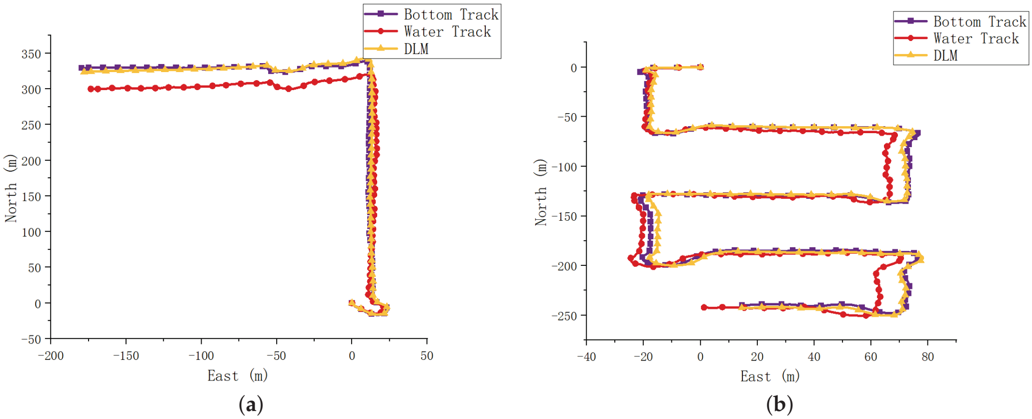


Figure 8. Trajectory comparison chart. (**a**) Experiment 1, (**b**) Experiment 2.

Position error comparison in Figure 9 reveals consistent small errors in trajectories computed from speeds estimated by the deep learning model, irrespective of the experimental scenario. Conversely, trajectory errors computed from water-track velocity occasionally decrease but rapidly accumulate during maneuvers, resulting in noticeable divergence over time. The deep learning model effectively addresses this issue.

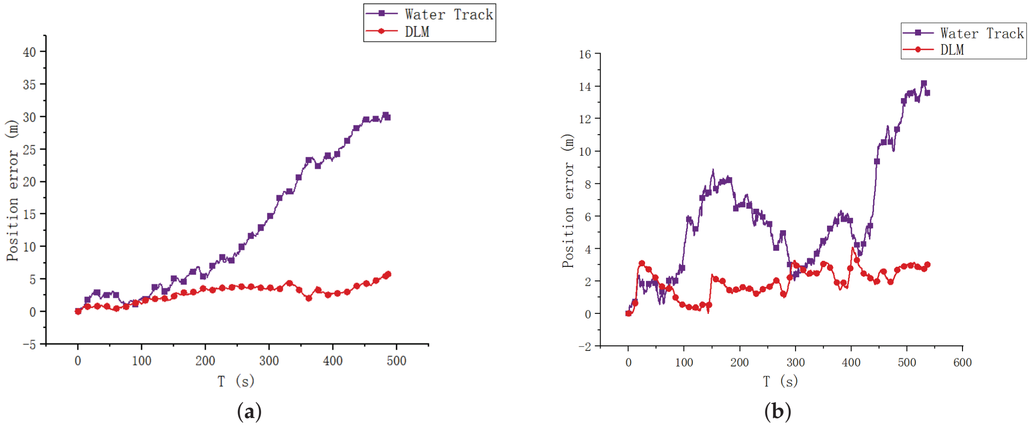


Figure 9. Position error comparison chart. (**a**) Experiment 1. (**b**) Experiment 2.

The estimated velocities of the bottom-track velocity, water-track velocity, and the speed estimation model are shown in Figures 10 and 11, where the green lines are water-track velocities, the red lines are bottom-track velocities, and the blue lines are velocities estimated by the DLM. As can be seen in the figure, the water-track velocity has the largest amplitude, indicating that the measured velocity values are unstable. The reason for this is that the water-track velocity measures the velocity of the AUV relative to the water flow, which has a great deal of variability. However, the bottom-track velocity is more stable and has a smaller amplitude because it measures the velocity of the AUV relative to the seafloor, which is stationary. The velocities estimated by the DLM are as steady as the bottom-track velocity, and they have the same general trend. Consequently, the calculated trajectories are more consistent with those calculated for the bottom-track velocities.

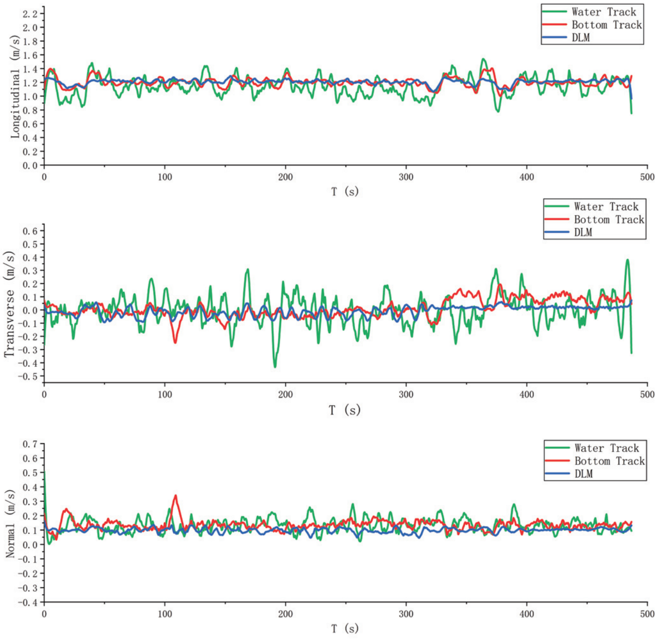


Figure 10. Comparison of velocity for Experiment 1.

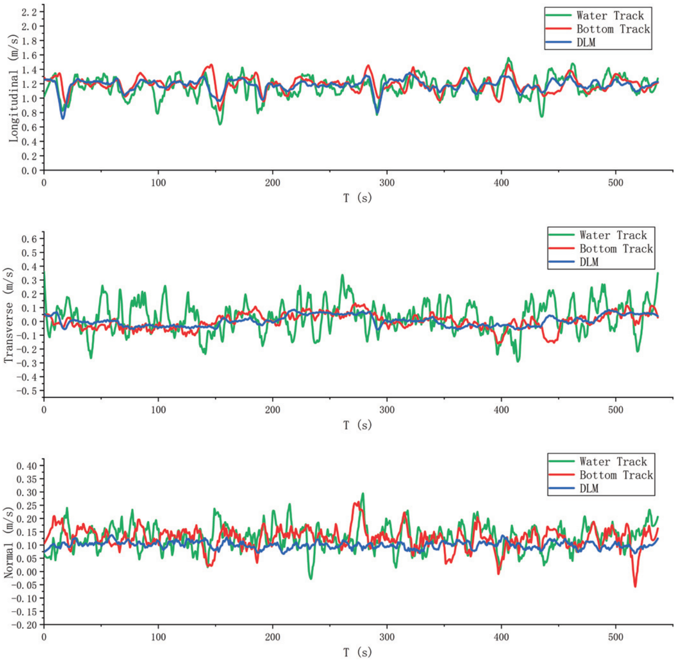
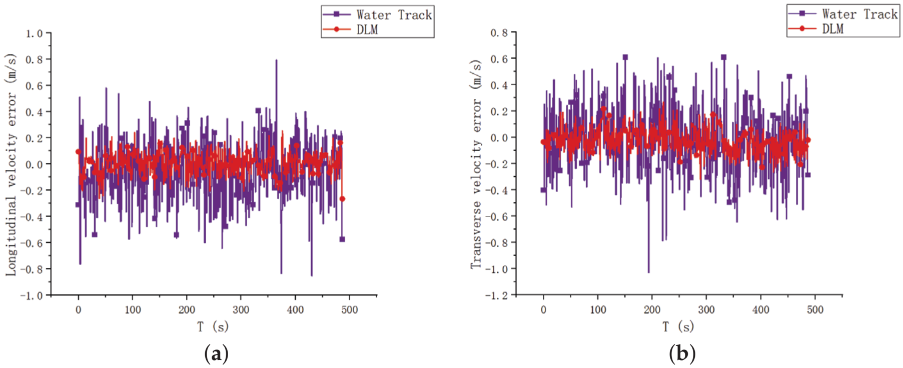


Figure 11. Comparison of velocity for Experiment 2.

Velocity error comparisons are presented in Figures 12 and 13 for straight and comb trajectory cases, respectively, to further elucidate the problem. Fluctuations in the forward and rightward velocity errors are notably higher for the water-track velocity. This discrepancy arises from uncertainties in the water-mass flow velocity, impacting absolute velocity measurements. Conversely, the deep learning model incorporates water-track velocity, extracting flow information and mitigating error fluctuations. Table 4 lists the maximum and average value of forward speed errors, maximum and average value of rightward speed errors, and maximum and average value of position errors obtained from Experiments 1 and 2, indicating consistently smaller error parameters for the deep learning-based speed estimation model compared to the water-track velocity. Consequently, the speed estimation model can effectively improve the accuracy of navigation.



(**a**) (**b**)

Figure 12. Comparison of velocity errors for Experiment 1. (**a**) Longitudinal velocity error. (**b**) Transverse velocity error.

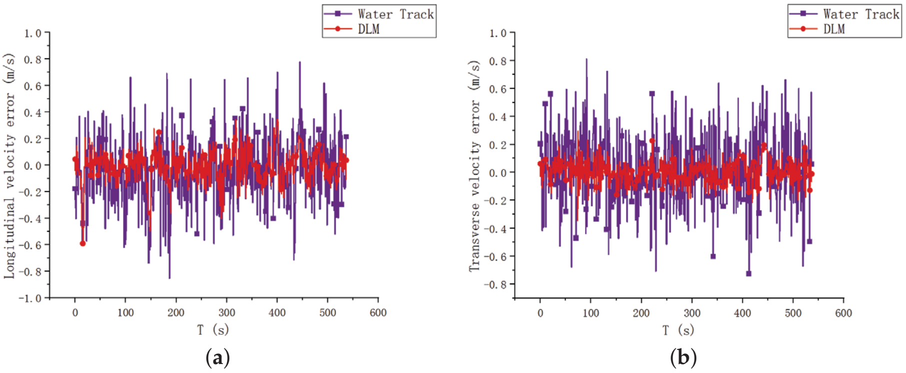


Figure 13. Comparison of velocity errors for Experiment 2. (**a**) Longitudinal velocity error. (**b**) Transverse velocity error.

Table 4. Comparison chart of error parameters.

	Experiment 1		Experiment 2	
	Water Track	DL	Water Track	DL
Maximum value of forward speed error (m/s)	0.853	0.266	0.853	0.591
Average of forward speed error (m/s)	0.192	0.067	0.201	0.088
Maximum value of rightward speed error (m/s)	1.030	0.320	0.810	0.345
Average of rightward speed error (m/s)	0.204	0.073	0.203	0.063
Maximum value of position error (m)	30.442	5.754	14.193	4.074
Average of position error (m)	12.420	2.784	5.889	1.973
Navigation accuracy	5.20%	0.90%	2.13%	0.48%

The experimental results on real data from the XH R300 demonstrate that the deep learning-based speed estimation model significantly enhances the combined navigation effectiveness of the DVL in cases of persistent bottom-track velocity failure. This study developed a deep learning speed estimation model based on LSTM and Self-Attention mechanisms, leveraging the time-series relationships among variables. Compensating the AUV velocity using the proposed model type was validated to incur less error compared to direct compensation with water-track velocity, thereby fulfilling the requirements of high-precision combined navigation. The cruise speed model did not consider large-scale vertical motions, which will be investigated in future work.

6. Conclusions

This study proposes a deep learning model leveraging acceleration, angle, angular velocity, and thruster speed as inputs to estimate AUV speed. LSTM is employed to extract time-series data from these variables, while Self-Attention enhances time-series data encoding to address long-term dependency issues. Water flow rate information, crucially embedded in water-track velocity, is separately encoded and utilized to enhance network generalization. The experimental results based on sea trial data demonstrate that the deep learning-based speed estimation model outperforms direct compensation with water-track velocity, achieving higher speed accuracy and meeting the demand for high-precision combined navigation in persistent DVL failure scenarios, thus enhancing the accuracy of the combined navigation system. Additionally, this research can be extended to scenarios with significant ocean currents, sharp turns, or muddy conditions that severely reduce DVL accuracy, further enhancing the reliability of the integrated navigation system.

Although the proposed method demonstrates superiority in most cases, the accuracy of the speed estimation model may deteriorate with declining bottom-track accuracy, warranting further investigation.

Author Contributions: Conceptualization, X.M. and Y.Y.; methodology, Z.Z.; software, X.M.; validation, Y.Y.; formal analysis, L.Z.; investigation, Z.Z.; resources, H.Q. and Z.W.; data curation, L.Z.; writing—original draft preparation, Y.Y.; writing—review and editing, X.M.; visualization, X.M.; supervision, Z.Z.; project administration, H.Q.; funding acquisition, Z.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key Research and Development Program of China (grant number 2023YFB4707000), the National Natural Science Foundation of China (Grant No. 52025111 and 52301369), and the Postdoctoral Applied Research Project of Qingdao (79002002/006).

Data Availability Statement: The original contributions presented in the study are included in the article material; further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hu, X.; Chen, J.; Zhou, H.; Ren, Z. Development of underwater electric manipulator based on interventional autonomous underwater vehicle (AUV). *J. Zhejiang Univ.-Sci. A* **2024**, *25*, 238–250. [CrossRef]
2. Cepeda, M.F.S.; Machado, M.d.S.F.; Barbosa, F.H.S.; Moreira, D.S.S.; Almansa, M.J.L.; de Souza, M.I.L.; Caprace, J.-D. Exploring Autonomous and Remotely Operated Vehicles in Offshore Structure Inspections. *J. Mar. Sci. Eng.* **2023**, *11*, 2172 [CrossRef]
3. Zhang, Y.; Huang, Z.; Chen, C.; Wu, X.; Xie, S.; Zhou, H.; Gou, Y.; Gu, L.; Ma, M. A Spiral-Propulsion Amphibious Intelligent Robot for Land Garbage Cleaning and Sea Garbage Cleaning. *J. Mar. Sci. Eng.* **2023**, *11*, 1482. [CrossRef]
4. Chiella, A.C.; Machado, H.N.; Teixeira, B.O.; Pereira, G.A. GNSS/LiDAR-Based Navigation of an Aerial Robot in Sparse Forests. *Sensors* **2019**, *19*, 4061. [CrossRef] [PubMed]
5. Koshaev, D.A. AUV Relative Position and Attitude Determination Using Acoustic Beacons. *Gyroscopy Navig.* **2023**, *13*, 262–275. [CrossRef]
6. Wang, J.; Xu, T.; Liu, Y.; Li, M.; Li, L. Augmented Underwater Acoustic Navigation with Systematic Error Modeling Based on Seafloor Datum Network. *Mar. Geod.* **2023**, *46*, 129–148. [CrossRef]
7. Ruben, V.; Friedrich, Z.; Antonio, S. In-Lab Demonstration of an Underwater Acoustic Spiral Source. *Sensors* **2023**, *23*, 4931. [CrossRef] [PubMed]
8. Berdyshev, V.I.; Kiselev, L.V.; Kostousov, V.B. Mapping Problems of Geophysical Fields in Ocean and Extremum Problems of Underwater Objects Navigation. *IFAC-PapersOnLine* **2018**, *51*, 189–194 [CrossRef]
9. Zhu, B.; He, H. Integrated navigation for doppler velocity log aided strapdown inertial navigation system based on robust IMM algorithm. *Optik* **2020**, *217*, 164871. [CrossRef]
10. Zhu, B.; Chang, G.; He, H.; Xu, J. Robust information fusion method in SINS/DVL/AST underwater integrated navigation. *J. Natl. Univ. Def. Technol.* **2020**, *42*, 107–114.
11. Bjørgo, I.S.; Alex, A.; Vahid, H. Path-Following Control of an Underwater Glider Aided by Machine Learning based Dead Reckoning Navigation. *IFAC PapersOnLine* **2023**, *56*, 7006–7013.
12. Sabet, T.M.; Daniali, M.H.; Fathi, A.; Alizadeh, E. A Low-Cost Dead Reckoning Navigation System for an AUV Using a Robust AHRS: Design and Experimental Analysis. *IEEE J. Ocean. Eng.* **2018**,*43*, 927–939. [CrossRef]
13. Geng, K.; Chulin, N. Applications of Multi-height Sensors Data Fusion and Fault-tolerant Kalman Filter in Integrated Navigation System of UAV. *Procedia Comput. Sci.* **2017**, *103*, 231–238. [CrossRef]
14. Xu, X.S.; Pan, Y.F.; Zou, H.J. SINS/DVL integrated navigation system based on adaptive filtering. *J. Huazhong Univ. Sci. Technol.* **2015**, *43*, 95–99.
15. Zhu, J.; Li, A.; Qin, F.; Che, H.; Wang, J. A Novel Hybrid Method Based on Deep Learning for an Integrated Navigation System during DVL Signal Failure. *Electronics* **2022**, *11*, 2980. [CrossRef]
16. Topini, E.; Fanelli, F.; Topini, A.; Pebody, M.; Ridolfi, A.; Phillips, A.B.; Allotta, B. An experimental comparison of Deep Learning strategies for AUV navigation in DVL-denied environments. *Ocean Eng.* **2023**, *274*, 114034. [CrossRef]
17. Tal, A.; Klein, I.; Katz, R. Inertial Navigation System/Doppler Velocity Log (INS/DVL) Fusion with Partial DVL Measurements. *Sensors* **2017**, *17*, 415. [CrossRef] [PubMed]
18. Kinsey, J.C.; Yang, Q.; Howland, J.C. Nonlinear Dynamic Model-Based State Estimators for Underwater Navigation of Remotely Operated Vehicles. *IEEE Trans. Control. Syst. Technol.* **2014**, *22*, 1845–1854. [CrossRef]
19. Zhu, Y.; Cheng, X.; Hu, J.; Zhou, L.; Fu, J. A Novel Hybrid Approach to Deal with DVL Malfunctions for Underwater Integrated Navigation Systems. *Appl. Sci.* **2017**, *7*, 759. [CrossRef]
20. Ma, T.; Li, Y.; Zhao, Y.; Zhang, Q.; Jiang, Y.; Cong, Z.; Zhang, T. Robust bathymetric SLAM algorithm considering invalid loop closures. *Appl. Ocean Res.* **2020**, *102*, 102298. [CrossRef]
21. Lv, P.F.; He, B.; Guo, J.; Shen, Y.; Yan, T.H.; Sha, Q.X. Underwater navigation methodology based on intelligent velocity model for standard AUV. *Ocean Eng.* **2020**, *202*, 107073. [CrossRef]
22. Li, D.; Xu, J.; He, H.; Wu, M. An Underwater Integrated Navigation Algorithm to Deal with DVL Malfunctions Based on Deep Learning. *IEEE Access* **2021**, *9*, 82010–82020. [CrossRef]

23. Mu, X.; He, B.; Zhang, X.; Song, Y.; Shen, Y.; Feng, C. End-to-end navigation for Autonomous Underwater Vehicle with Hybrid Recurrent Neural Networks. *Ocean Eng.* **2019**, *194*, 106602. [CrossRef]
24. Wang, D.; Wang, B.; Huang, H.; Zhang, H. A SINS/DVL navigation method based on hierarchical water velocity estimation. *Meas. Sci. Technol.* **2024**, *35*, 015116. [CrossRef]
25. Lim, B.; Zohren, S.; Roberts, J.S. Population-based Global Optimisation Methods for Learning Long-term Dependencies with RNNs. *arXiv* **2019**, arXiv:1905.09691.
26. Guo, J.; Zhang, X.; Liang, K.; Zhang, G. Memory-Enhanced Knowledge Reasoning with Reinforcement Learning. *Appl. Sci.* **2024**, *14*, 3133. [CrossRef]
27. Sukma, A.J.; Widodo, M.K. Thrust and efficiency enhancement scheme of the fin propulsion of the biomimetic Autonomous Underwater Vehicle model in low-speed flow regime. *Ocean Eng.* **2022**, *243*, 110090.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

A Comprehensive Signal Quality Assessment for BDS/Galileo/GPS Satellites and Signals

Yijun Tian [1], Guorui Xiao [2,*], Rui Guo [1], Dongqing Zhao [2], Lu Zhang [1], Jie Xin [1], Jinglei Guo [1], Yuechao Han [1], Xuefan Du [1], Donghan He [1] and Zheng Qin [1]

[1] Beijing Satellite Navigation Center, Beijing 100094, China; tianyijun56@163.com (Y.T.); shimbarsalon@163.com (R.G.); 13366793306@189.cn (L.Z.); x15340137578@163.com (J.X.); dktgjl@126.com (J.G.); hanyuechaom@163.com (Y.H.); d1007152154@foxmail.com (X.D.); 18255431825@163.com (D.H.); 15810225727@163.com (Z.Q.)

[2] Institute of Geospatial Information, Information Engineering University, Zhengzhou 450001, China; dongqing.zhao@hotmail.com

* Correspondence: xgr@whu.edu.cn

Abstract: With the modernization of global navigation satellite systems (GNSS), especially the rapid development of the BeiDou Navigation Satellite System (BDS), more observations of satellites and signals have become available. Using data of the globally distributed MGEX stations, a systematic and comprehensive evaluation of signal characteristics for BDS-3, BDS-2, GPS, and Galileo is conducted in terms of carrier-to-noise ratio (C/N0), code noise, and multipath in the contribution. First, a comprehensive signal quality assessment method for BDS/Galileo/GPS satellites and signals is proposed, including *C/N0* modeling and MP modeling. For BDS, the BDS-3 satellites apparently have higher signal power than the BDS-2 satellites at the same frequency such as B1I and B3I, and the signal B2a of BDS-3 is superior to other signals in regard to signal power, which is comparable with the superior Galileo E5 signals and GPS L5. Among all the signals, the observation accuracy of E5 is the highest regardless of receiver types, and next highest are BDS-3 B2a and GPS L5. Due to not being affected by the systematic code errors of BDS-2, the observations of BDS-3 satellites contain smaller multipath errors than that of BDS-2 satellites. As for the multipath suppression performance, the BDS-3 signal B2a, GPS L5, and Galileo E5 and E5b perform better than the other signals, which may be related to their wide signal bandwidths.

Keywords: BDS-3; signal quality assessment

Citation: Tian, Y.; Xiao, G.; Guo, R.; Zhao, D.; Zhang, L.; Xin, J.; Guo, J.; Han, Y.; Du, X.; He, D.; et al. A Comprehensive Signal Quality Assessment for BDS/Galileo/GPS Satellites and Signals. *Remote Sens.* **2024**, *16*, 2277. https://doi.org/10.3390/rs16132277

Academic Editor: Baocheng Zhang

Received: 11 May 2024
Revised: 11 June 2024
Accepted: 18 June 2024
Published: 21 June 2024

1. Introduction

As of May 2024, the data of 30 BDS-3 satellites in orbit are available, including 3 GEO satellites, 3 IGSO satellites, and 24 MEO satellites [1–4], and currently, BDS constellation is composed of BDS-2 and BDS-3 [5,6]. Furthermore, four BDS-3 experimental satellites (BDS-3s) are available, including two IGSO and two MEO satellites [7,8]. BDS-3 added two new signals, B1C and B2a, on the basis of compatibility with BDS-2 B1I and B3I signals [9–13]. Table 1 shows the characteristics of the current BDS-3 public signals. Apparently, the two new signals, B1C and B2a, have higher ranging accuracy, wider bandwidth, and better interoperability. In the signal design, to provide better compatibility and interoperability, the B1C frequency is selected as 1575.42MHz, which is used for frequency multiplexing with Galileo E1 and GPS L1, and B2a is selected as 1176.45 MHz, which is compatible with Galileo E5a and GPS L5.

With the continuous improvement of the GNSS navigation and positioning theory, the research on signal quality evaluation of GPS, Galileo, BDS-2, and other systems has been relatively mature. The contents of the signal quality evaluation include the observed signal-to-noise ratio [14–16], pseudorange multipath and noise [17–20], and pseudorange and carrier observation accuracy [21–23]. For the new signal E5 of the Galileo, the research of Zaminpardaz et al. on the four IOV and nine FOC satellites showed that the E5 signal modulated with AltBOC has significantly better signal power and lower noise and multipath biases among the five Galileo signals. The instantaneous ambiguity fixation on the E5 signal can be achieved by correcting the multipath [19]. Similarly, the research by Tian et al. and Simsky et al. indicated that apart from outperforming others in signal powers, the E5 signal has a high level of short delay multipath suppression, which may be related to its advanced signal modulation method [22,23]. As for BDS, the characteristics of the signals B1C, B2a, and B2b were first analyzed by Zhang et al., utilizing the five BDS-3 satellites, and it was found that the elevation-dependent pseudorange biases that appeared in the BDS-2 satellite observations no longer exist in the new-generation BDS-3 satellites [24].

Table 1. BDS-3 open service signal system.

Signal Component	Frequency (MHz)	Bandwidth (MHz)	Code Rate (Mcps)	Modulation	Broadcast Satellite	Compatible Frequency
B1I	1561.09	4.092	2.046	BPSK (2)	All	
B3I	1268.52	20.46	10.23	BPSK (10)	All	
B1C_data	1575.42	32.736	1.023	BOC (1,1)	IGSO MEO	L1, E1
B1C_pilot				QMBOC (6,1,4/33)		
B2a_data	1176.45	20.46	10.23	BPSK (10)	IGSO MEO	L5, E5a
B2a_pilot						
B2b_I	1207.14	—	10.23	QPSK (10)	IGSO MEO	E5b
B2b_Q						

The availability of more BDS-3 satellite observation data has made the evaluation of BDS-3 performance a research hotspot. Fu Zheng et al. studied the receiver pseudorange deviation between BDS-2 and BDS-3 and found that the receiver pseudorange deviation between BDS-3 and BDS-2 varies greatly and is related to the receiver type [25]. By investigating the intrinsic characteristics of receiver-related biases in the combination of BDS-3 and BDS-2, Mi et al. concluded that there exist noticeable differences in the epoch-wise DCB estimates of the same type between BDS-3 and BDS-2 [26]. As for the atomic clock, the drift rate and frequency stability of the clock on the BDS-3 satellite are significantly better than those on BDS-2 [27,28]. Furthermore, when studying the combination positioning of BDS-3 and other systems, the increase in BDS-3 new signals can significantly improve the positioning performance of BDS/GPS/Galileo [29,30]. However, it was found that the inter-system bias (ISB) between BDS-3 and other GNSS systems cannot be ignored. Even ISB exists between BDS-3 and BDS-2 [31,32].

The excellent characteristics of BDS-3 interest us in further evaluating its performance compared to other GNSS systems. In addition, with the completion of BDS-3, the multi-system fusion positioning service with BDS as the core has preliminary conditions. The combination of BDS-3 with other GNSS systems can provide sufficient observations and huge potential for improving the accuracy, reliability, and availability of PNT services [33–35], yet the differences in data quality and random characteristics between observations of different types of satellites and signals make it impossible to achieve ideal results in multi-system fusion positioning. In addition, there are currently many brands of receivers for BDS-3 data processing, which also makes the quality and characteristics of the observation data inconsistent. Therefore, conducting a comprehensive evaluation of BDS-3 is of great significance, as compared to other GNSS systems.

Utilizing data from globally distributed MGEX stations that can receive BDS-3 satellite signals, this paper first provides a brief description of the BDS-3 signal regime, then provides a systematic and comprehensive assessment of its data quality, analyzing and comparing it with other GNSS systems.

2. Data and Methods

In order to study the performances of signals and satellites of BDS-3, quality evaluations on C/N0, code noise, and multipath were conducted using data from the MGEX stations that can process the BDS-3 signals.

2.1. Data

Table 2 shows the classification of the MGEX stations that can process BDS-3 signals from 14 May to 20 May 2019 (DOYs 2019, 134–140). The earlier use of data here is because BDS-3 data are just available at this time, and BDS-2 data are also relatively complete, laying a solid foundation for the subsequent BDS performance analysis. In the experiment, as the stations equipped with the same type of receiver showed high consistency, the observations of the representative stations were used for experimental analysis to compare the characteristics of C/N0, code noise, and the multipath between BDS-3 and the other GNSS systems in different types of receivers. All the selected stations were installed in an open sky environment, and data can be found at the IGS official website (http://www.igs.org/mgex/data-products/#data, accessed on 2 April 2024).

Table 2. Classifications of different receiver types in MGEX stations (G: GPS; E: Galileo; C: BDS).

Receiver Type	Firmware Version	Observation Signal	Selection of Stations
JAVAD	TRE_3 3.7.6	G: L1, L2, L5 E: E1, E5a, E5b, E5 C: B1I, B2I, B3I, B2a	POTS, SGOC, SUTM, ULAB, URUM, WIND, WUH2
	Others	G: L1, L2, L5 E: E1, E5a, E5b, E5 C: B1I, B2I, B3I	9 measuring stations, such as BSHM, KOKV
LEICA	GR10, GR25, GR30, GR50	G: L1, L2, L5 E: E1, E5a, E5b, E5 C: B1I, B2I	25 measuring stations, such as ALIC, EBRE
TRIMBLE	NETR9	G: L1, L2, L5 E: E1, E5a, E5b, E5 C: B1I, B2I, B3I,	48 measuring stations, such as BOR1, CUT0
SEPT	POLARX5	G: L1, L2, L5 E: E1, E5a, E5b, E5, E6 C: B1I, B2I, B3I	40 measuring stations, such as CEBR, HOB2

2.2. Carrier-to-Noise Density Ratio Evaluation Model

The carrier-to-noise density ratio (*C/N0)* values are usually used to indicate the signal quality of GNSS observed data. In the experiment, the *C/N0* values of all epochs corresponding to a certain satellite *s* at the frequency *j* were first extracted, and then the corresponding satellite elevation *EL* on each epoch was calculated.

$$EL = \arcsin\left(e_u\right) \tag{1}$$

where e_u is the transformation coordinate value between the earth-centered fixed coordinate system and the topocentric coordinate system.

Two models with regard to $C/N0$ were established in the contribution so as to evaluate the power of different signals in different GNSS systems.

The first model was for comparing the signal power between different satellite types at the same frequency. In the model, the satellite elevations were used as the basis for grouping the $C/N0$ values, and the $C/N0$ values of a certain satellite at a certain frequency point were divided into groups every 10 degrees. Afterward, an average value of $C/N0$ was calculated within each group. The formula is as follows.

$$C/N0^{s}_{j,ave}(k) = \frac{\sum_{i=1}^{n} C/N0^{s}_{j}(i)_{[k\cdot 10^{\circ},(k+1)\cdot 10^{\circ}]}}{n} \tag{2}$$

where $k = 0 \sim 8$ denotes the k_{th} elevation group, and n is the number of epochs within the elevation group of $\left[k\cdot 10^{\circ},\ (k+1)\cdot 10^{\circ}\right]$. $C/N0^{s}_{j,ave}(k)$ is the average $C/N0$ value of satellite s at the frequency j within the k_{th} elevation interval.

The second model was for comparing the signal power at different signals. Different from the first model, the $C/N0$ values over the same type of satellites at the same frequency were grouped according to their elevations. Then, $C/N0$ values in each elevation interval of 10 degrees were averaged, and STDs of $C/N0$ were calculated to estimate uncertainties.

$$C/N0_{j,ave}(k) = \frac{\sum_{s=1}^{m}\sum_{i=1}^{n} C/N0^{s}_{j}(i)_{[k\cdot 10^{\circ},(k+1)\cdot 10^{\circ}]}}{n} \tag{3}$$

where m is the number of satellites that are the same type, and $C/N0^{s}_{j,ave}(k)$ is the average $C/N0$ value over a certain type of satellite at the frequency j within the k_{th} elevation interval. STDs can be expressed as:

$$C/N0_{j,\mathrm{STD}}(k) = \sqrt{\frac{\sum_{s=1}^{m}\sum_{i=1}^{n} \left(C/N0^{s}_{j}(i)_{[k\cdot 10^{\circ},(k+1)\cdot 10^{\circ}]} - C/N0_{j,ave}(k)\right)^{2}}{n}} \tag{4}$$

2.3. Noise Assessment Model

The BDS, GPS, or Galileo single-system pseudorange and phase-nondifference observation equations are:

$$P^{s}_{r,j} = \rho^{s}_{r} + (dt_{r} - dt^{s}) + T^{s}_{r} + \beta_{j} I^{s}_{r,j} + (d_{r,j} + d^{s}_{,j}) + e^{s}_{r,j} \tag{5}$$

$$\Phi^{s}_{r,j} = \rho^{s}_{r} + (dt_{r} - dt^{s}) + T^{s}_{r} - \beta_{j} I^{s}_{r,j} + \varepsilon^{s}_{r,j} + \lambda_{j}(N^{s}_{r,j} + b_{r,j} + b^{s}_{,j}) \tag{6}$$

$$P^{s}_{r,j} - \Phi^{s}_{r,j} = 2\beta_{j} I^{s}_{r} + (d_{r,j} + d^{s}_{,j}) - \lambda_{j}(N^{s}_{r,j} + b_{r,j} + b^{s}_{,j}) \tag{7}$$

$$\Delta(P^{s}_{r,j} - \Phi^{s}_{r,j}) = \Delta e^{s}_{r,j} \tag{8}$$

where $P^{s}_{r,j}$ is the pseudorange observation, and $\Phi^{s}_{r,j}$ is the carrier phase observation. ρ^{s}_{r} is the geometry term, and dt_{r} and dt^{s} are the receiver clock error and satellite clock error. T^{s}_{r} is the tropospheric delay, and $I^{s}_{r,j}$ is the ionospheric delay. $d_{r,j}$ and $d^{s}_{,j}$ are the hardware delays of the receiver and satellite in pseudorange observations, respectively. $b_{r,j}$ and $b^{s}_{,j}$ represent the initial phase deviation and hardware delay of the receiver and satellite in the carrier phase observation. $N^{s}_{r,j}$ is the ambiguity term in the carrier phase observation. $e^{s}_{r,j}$ and $\varepsilon^{s}_{r,j}$ represent other unmodeled errors, such as observation noise and multipath effects in pseudorange and carrier phase observations, respectively.

Using the pseudorange minus the carrier, the geometric term can be eliminated, and then using the inter-epoch difference, the ambiguity and ionospheric term can be eliminated without circle slips. Apart from this, the phase noise was small and negligible in comparison to the pseudorange noise; thus, the code noise can be evaluated by the above equation.

2.4. Code Multipath Analysis

A GNSS receiver theoretically receives a direct wave signal from the GNSS satellite, but in reality, in addition to the direct wave, there are several indirect waves that reach the receiver's signal-receiving antenna, including reflected waves from ground or ground objects, reflected waves from the satellite, and scattered waves from the atmospheric propagation medium.

Generally, the code multipath can be evaluated by constructing a combination of multipaths using the pseudorange and carrier phase observations, which is expressed as follows.

$$\begin{aligned} MP_j^s &= P^s_{r,j} - \tfrac{f_j^2+f_i^2}{f_j^2-f_i^2}\Phi^s_{r,j} + \tfrac{2f_i^2}{f_j^2-f_i^2}\Phi^s_{r,i} \\ &= P^s_{r,j} - (\tfrac{f_j^2}{f_j^2-f_i^2}\Phi^s_{r,j} - \tfrac{f_i^2}{f_j^2-f_i^2}\Phi^s_{r,j}) - \tfrac{f_i^2}{f_j^2-f_i^2}(\Phi^s_{r,j} - \Phi^s_{r,i}) \end{aligned} \tag{9}$$

The penultimate term of the multipath combination corresponds to the ionosphere-free combination of the carrier phase so as to eliminate the effect of the first-order ionospheric delay, while the last term is the geometry-free combination so that the effect of the geometry term in the combination can be effectively eliminated.

Substituting the observed value equation into the above equation gives the equation:

$$MP_j^s = dM_j - \frac{f_i^2 + f_j^2}{f_j^2 - f_i^2}dm_j + \frac{2f_i^2}{f_j^2 - f_i^2}dm_i - B \tag{10}$$

where the subscripts i and j $(i \neq j)$ denote the frequency, and dM and dm stand for the code multipath error and the carrier phase multipath error. B includes the ambiguity term, as well as the hardware delay deviation, which can be expressed as

$$B = \frac{f_i^2 + f_j^2}{f_j^2 - f_i^2}\lambda_j N_j - \frac{2f_i^2}{f_j^2 - f_i^2}\lambda_i N_i - d_j - d^s \tag{11}$$

Theoretically, the carrier phase multipath error and the systematic errors, such as phase noise, were small, making them negligible with respect to the code multipath. When the carrier phase observation does not appear in the cycle slip, the ambiguity parameter can be treated as a constant. The differential code deviation was also stable in the short term. Therefore, in the multipath combination, it was possible to smooth the irrelevant items by taking the average value. Then, the code multipath in a complete arc can be obtained as

$$MP^s_{j,var} = MP^s_j - < MP^s_j > \tag{12}$$

where the operator $< \cdot >$ indicates smoothing in the time domain.

In addition, considering the fact that the code accuracy of some special signals can reach the centimeter level, it was necessary to try to make the frequency difference between the two signals as large as possible to minimize the effects of phase multipath and phase noise when choosing to build the multipath combination. Therefore, the frequency f_j and f_i were chosen in the paper, as shown in Table 3.

Table 3. Selection of signals in different multipath combinations.

System	Galileo					BDS-2			BDS-3			GPS		
f_j	E1	E5a	E5b	E5	E6	B1I	B2I	B3I	B1I	B3I	B2a	L1	L2	L5
f_i	E5a	E1	E1	E1	E1	B2I	B1I	B1I	B3I	B1I	B1I	L2	L1	L1

Similar to the *C/N0* model, the multipath values on all epochs corresponding to each satellite at the same frequency were grouped according to the satellite elevations, with each 10° elevation as a split. Then, the multipath values within the same elevation interval were averaged.

$$MP^s_{j,ave}(k) = \frac{\sum_{i=1}^{n} MP^s_{j,var}(i)_{)[k\cdot 10^\circ,(k+1)\cdot 10^\circ]}}{n} \tag{13}$$

where $MP^{s}_{j,ave}(k)$ denotes the mean multipath value of satellite s at frequency j in the k_{th} set of elevation intervals.

3. Experiment Results

3.1. Experimental Analysis of Signal Power

Using the model given above, the *C/N0* values of BDS-3 satellites at different signals are analyzed. For ease of presentation, BDS-2 is represented as capital II, the BDS-3 experimental system is represented as capital IIIs, and BDS-3 is represented as III in the following diagrams.

The average *C/N0* values of all satellites for BDS-2, BDS-3s, and BDS-3 at different signals as a function of elevation are depicted in Figure 1, derived from the first model of C/N0. The *C/N0* values of the same types of satellites are represented by lines of the same color, indicating that the difference in signal power between satellites of the same type is very small. In terms of B1I and B3I, the average *C/N0* values of BDS-3 MEO satellites are all about 1–2 dB higher than that of BDS-2 MEO satellites and 3–4 dB higher than that of BDS-2 GEO and IGSO satellites. The lower signal strength of GEO and IGSO satellites is due to their high satellite orbit altitude, resulting in greater signal attenuation.

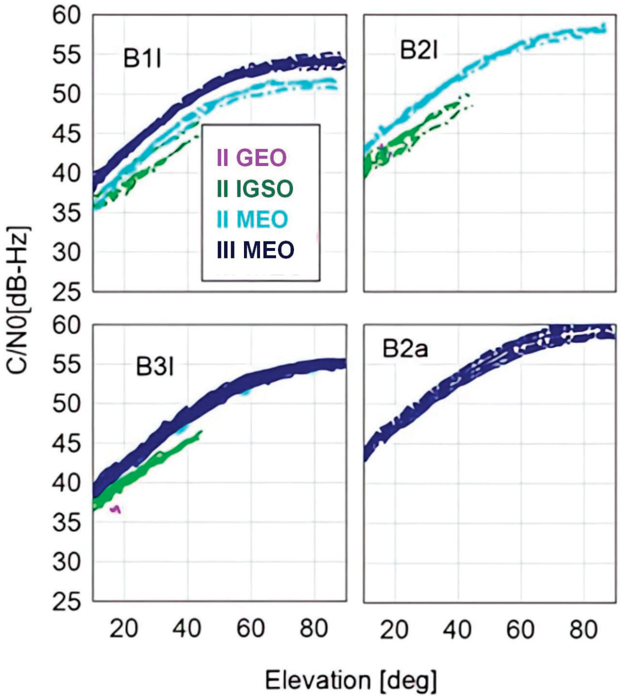


Figure 1. Average *C/N0* values against the elevations of all the satellites for BDS B1I/B2I/B3I/B2a based on observations collected from MGEX stations during DOY 134–140 in 2019. Only the results of the representative receiver type JAVAD are plotted. (II: BDS-2; IIIs: BDS-3 experimental system; III: BDS-3).

Due to the small differences between satellites of the same type, the *C/N0* values of satellites of the same type are calculated together according to the second *C/N0* model to further analyze and compare the signal performances of different GNSS systems, which are shown in Figure 2. In the experiment, it was found that the experimental results obtained by different types of receivers were basically consistent; therefore, only performances of the receiver type JAVAD TRE_3 3.7.6 are described in the contribution.

The left panel of Figure 2 first shows the comparison of signal powers at various frequency points of BDS-2 and BDS-3. Consistent with the above analysis results, the signal powers of BDS-3 satellites at B1I and B3I show a higher level than BDS-2 satellites. Among all the signals of BDS, B2a performs the best, varying from 46 to 60 dB-Hz.

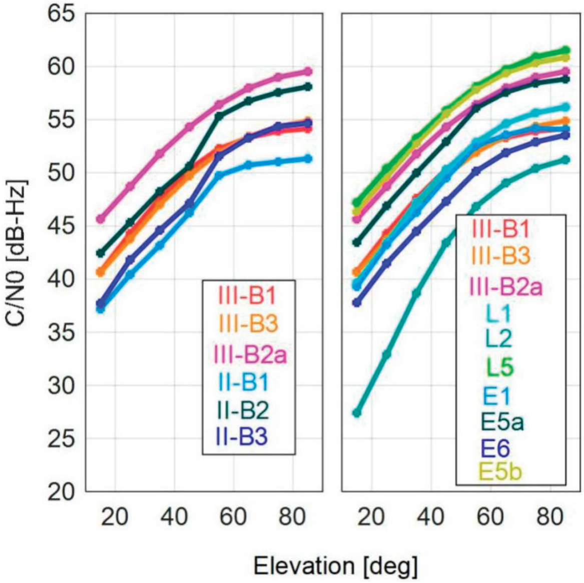


Figure 2. Average C/N0s against the elevations of different signals. The left panel is only for the BDS signals, while the right panel is for all the signals of BDS-3, Galileo, and GPS.

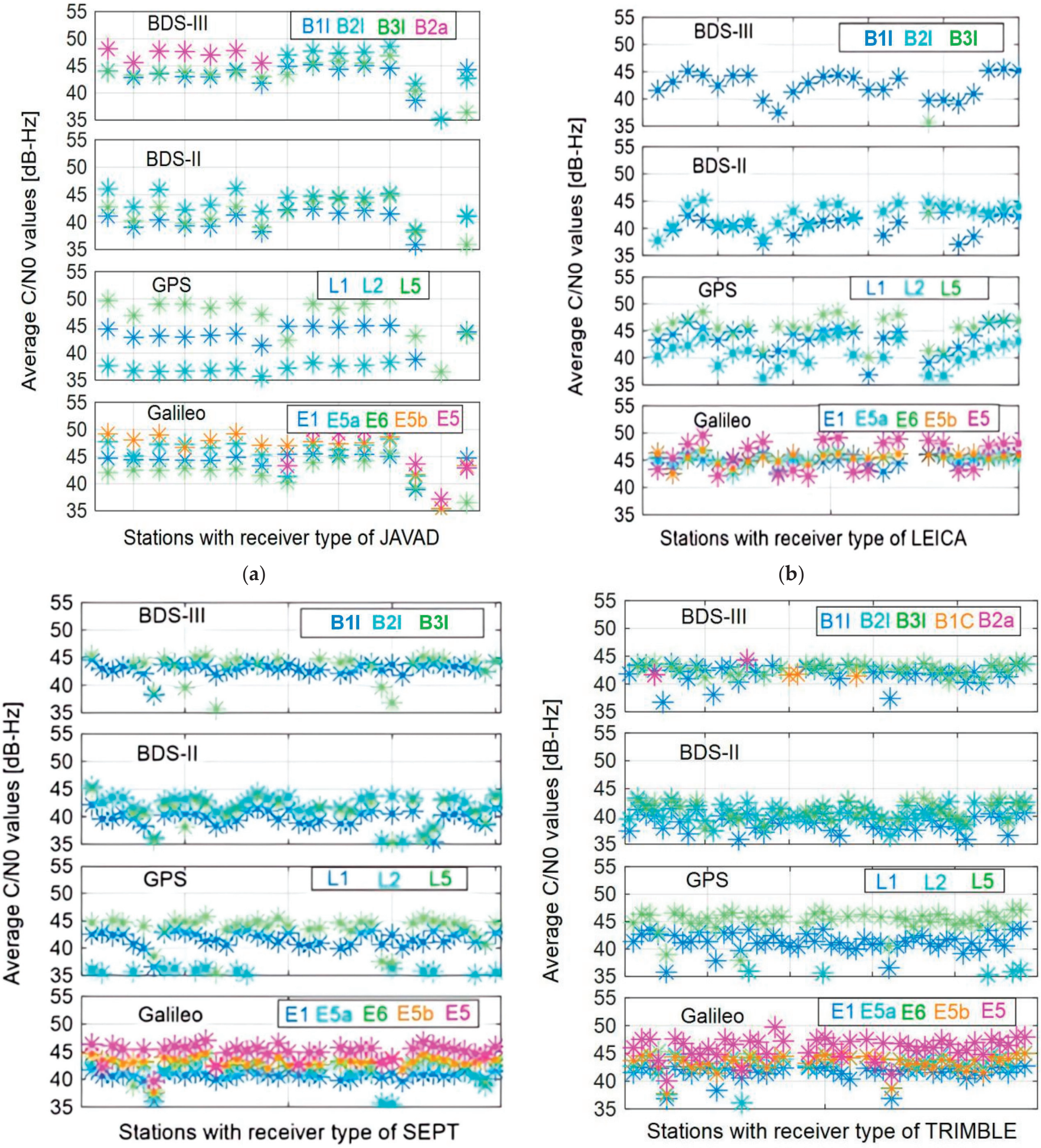
The right panel of Figure 2 shows the C/N0s comparison of various types of signals for BDS-2, BDS-3, GPS, and Galileo. Among all the signals, the GPS L5, BDS-3 B2a, and Galileo E5b perform slightly better than the other signals, and L2 of GPS always has the lowest signal power.

The average *C/N0* values of BDS-2, BDS-3, GPS, and Galileo on different types of receivers at each signal are shown in Figure 3. There is little difference in signal power on different types of receivers. Comparing the *C/N0* averages on the first seven stations of JAVAD, which can receive B2a signals, the B2a signal has significantly higher power than the other signals, regardless of station. Among all the BDS-2 signals, the signal power of B1I is always smaller than that of B2I and B3I on the same station, while the order of signal power for GPS is L5 > L1 > L2. For Galileo, the signal power of E5 is the strongest, although on some stations equipped with LEICA-type receivers, the E5 signal is not the best.

3.2. Experimental Analysis of Pseudorange Noise

Similarly, the experimental data given in Section 2.1 are used to perform the experimental analysis of pseudorange noise according to the Equation (8). CC is represented as the pseudorange noise.

The time series of CC values of BDS-3 C35 at B2a are given in Figure 4. The gray line indicates the satellite elevation. There is a certain correlation between the pseudorange noise and satellite elevation. The higher the satellite elevation, the smaller the noise, and vice versa. In Figure 5, the time series of CC values of four BDS-3 satellites at different signals are depicted. The performances of the four satellites are almost the same, and all of them show elevation-dependent and periodic characteristics. Corresponding to the high signal power, the B2a signal possesses the lowest noise level among the three signals B1I, B3I, and B2a.



(**a**) (**b**)

(**c**) (**d**)

Figure 3. Comparison of *C/N0* values at different signals for different receiver types. The horizontal axis in subfigure (**a**) represents 15 stations with the receiver type JAVAD, while that in subfigure (**b**) represents 24 stations with the receiver type LEICA, that in subfigure (**c**) represents 29 stations with the receiver type SEPT, and that in subfigure (**d**) represents 40 stations with the receiver type TRIMBLE.

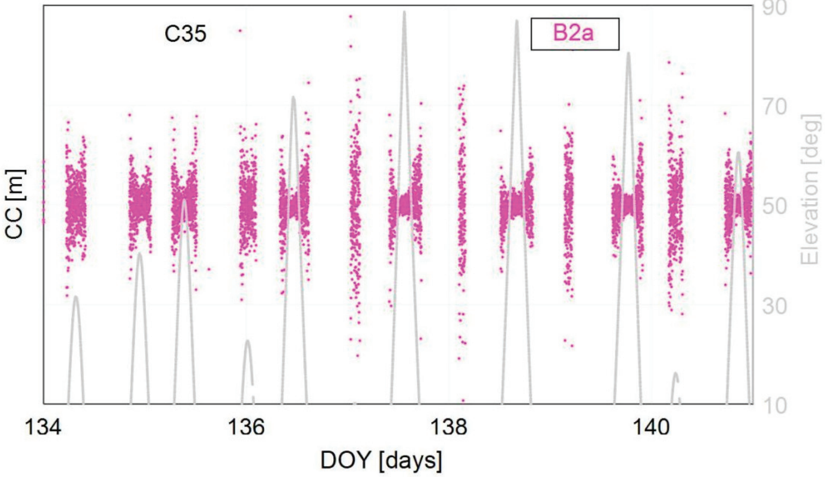


Figure 4. Time series of CC values of the BDS-3 C35 satellite at the B2a signal during DOY 134–140 in 2019. The satellite elevation over time is indicated by the gray line.

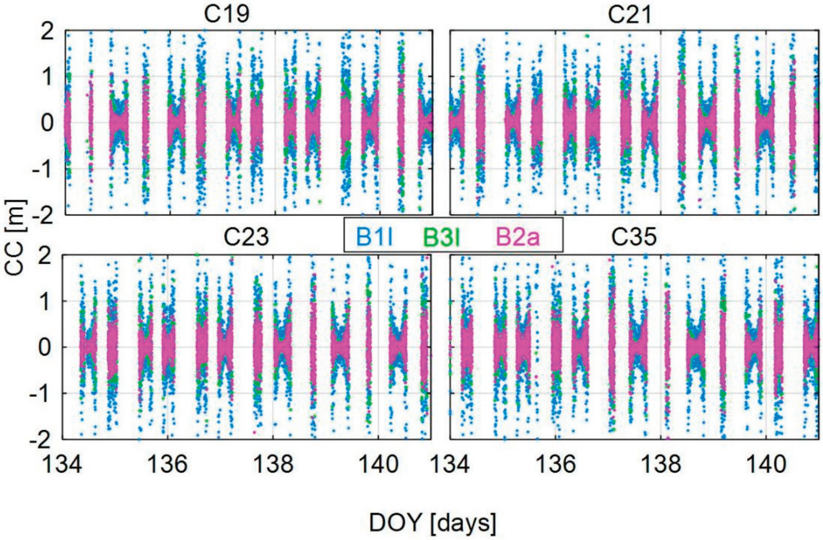


Figure 5. Time series of CC values of the four selected BDS-3 satellites at B1I/B3I/B2a during DOY 134–140 in 2019.

Similarly, the time series of CC values for BDS-2, Galileo, and GPS satellites are given in Figure 6. For BDS-2, the GEO and IGSO satellites have larger noise levels than MEO satellites, which can also be related to their higher orbital altitude. Possessing high signal power, BDS-2 B3I, GPS L5, and Galileo E5b show smaller noise than other signals. It is worth noting that although GPS L2 has a low signal power, its noise level is not significantly greater than the other signals, which indicates that the noise level is not only related to the signal power.

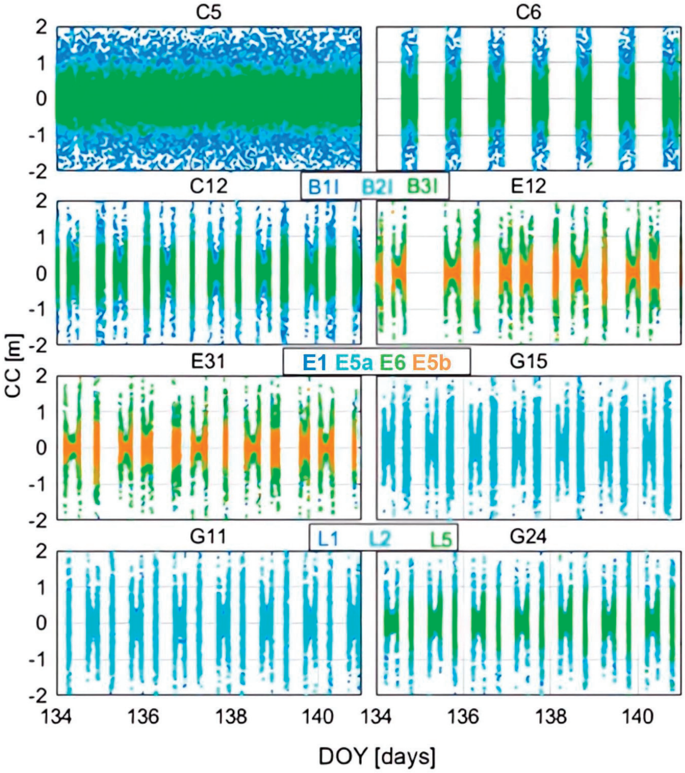


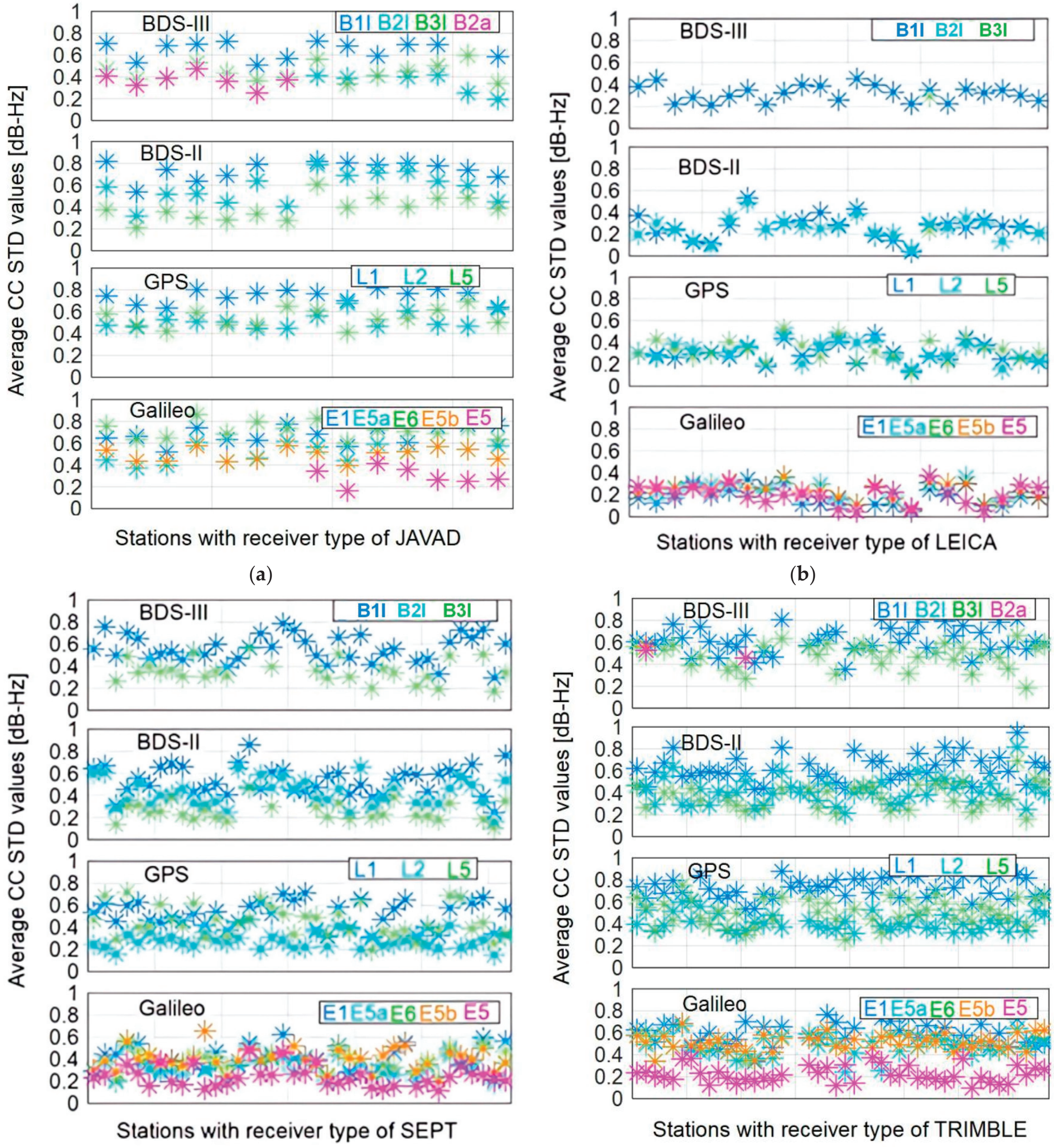
Figure 6. Time series of CC values for BDS-2 (C5, C6, and C12), Galileo (E12 and E31), and GPS (G15, G11, and G24) satellites at different signals during DOY 134–140 in 2019. Legends B1I (light green), B2I (bright green), and B3I (light blue) are for BDS satellites C5, C6, and C12. Legends E1 (light blue), E5a (light green), E6 (bright green), and E5b (light yellow) are for Galileo satellites E12 and E31. Legends L1 (light blue), L2 (light green), L5 (bright green) are for GPS satellites G15, G11 and G24.

Similar to the comparison of *C/N0* on each station in Figure 3, STDs of CC values at different signals on each station are depicted in Figure 7, which are used to reflect the code noise level. Comparisons of different signals on different stations have almost identical characteristics, indicating that the pseudorange noise levels of different signals have little relationship with the receiver type. It is worth noting that the Galileo E5 signal exhibits optimal performance due to its high signal power.

3.3. Experimental Analysis of Multipath

Corresponding to Figures 5 and 6, comparisons of multipath values are presented in Figures 8 and 9, respectively. The BDS-3 satellites perform better than BDS-2, and their multipath values no longer tend to vary with the elevation, which may be related to the better multipath resistance of their new modulated signals. In addition, the on-satellite multipath errors that appeared on BDS-2 satellites are no longer present on the BDS-3 satellites. The BDS-3 B2a, GPS L5, and Galileo E5b have significantly smaller multipath errors than other signals.

In order to more clearly compare the magnitude of multipath values at each signal, the STD values of multipath combinations for each satellite at different signals are shown in Figure 10. The accuracies of BDS-3 B2a, BDS-2 B3I, GPS L5, and Galileo E5b are superior to other signals in their respective systems.



(**a**)

(**b**)

(**c**)

(**d**)

Figure 7. Comparison of average CC STD values in different stations. The horizontal axis in subfigure (**a**) represents 15 stations with receiver type JAVAD, while that in subfigure (**b**) represents 24 stations with receiver type LEICA, that in subfigure (**c**) represents 29 stations with receiver type SEPT, and that in subfigure (**d**) represents 40 stations with receiver type TRIMBLE. In each subgraph, the 4 panels are for BDS-3 B1I/B3I/B2a, BDS-2 B1I/B2I/B3I, GPS L1/L2/L5, and Galileo E1/E5a/E6/E5b/E5, respectively.

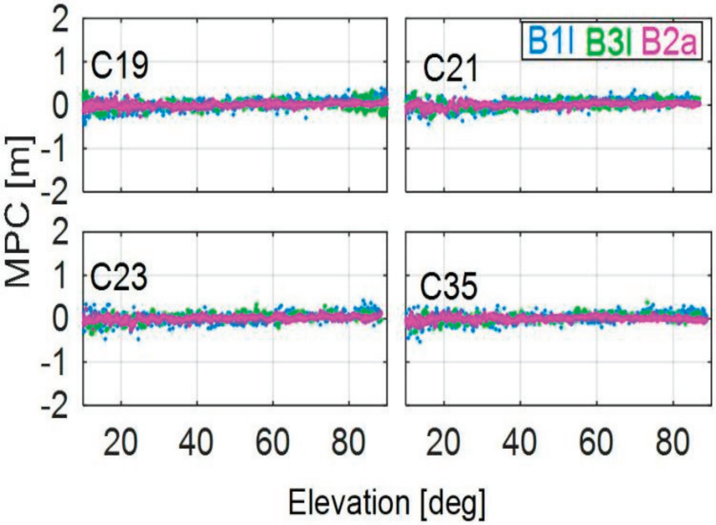


Figure 8. Multipath values against satellite elevation for BDS-3 C19, C21, C23, and C35 satellites during DOY 134–140 in 2019. MPC stands for Multipath Combination.

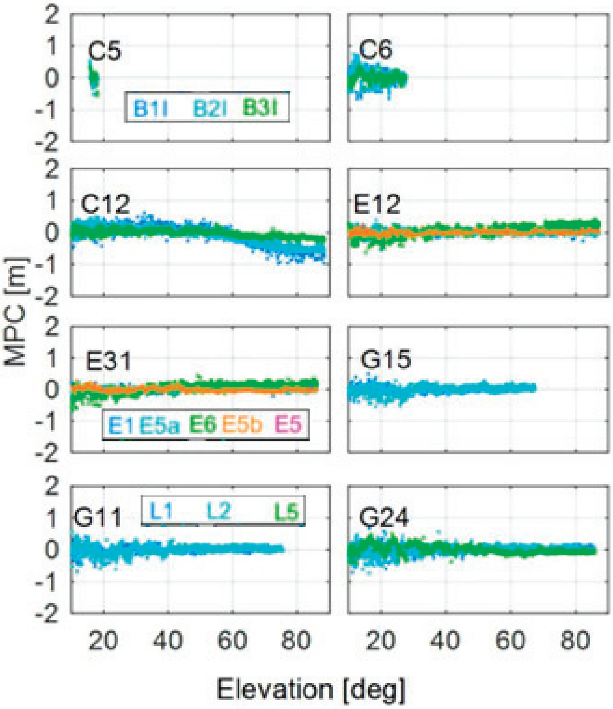


Figure 9. Multipath values against satellite elevation for BDS-2 (C5, C21, and C23), Galileo (E12 and E31), and GPS (G15, G11, and G24) satellites during DOY 134–140 in 2019. Legends B1I (light blue), B2I (light green) and B3I (bright green) are for BDS satellites C5, C6, and C12. Legends E1 (light blue), E5a (light green), E6 (bright green), and E5b (light yellow) are for Galileo satellites E12 and E31. Legends L1 (light blue), L2 (light green) and L5 (bright green) are for GPS satellites G15, G11 and G24.

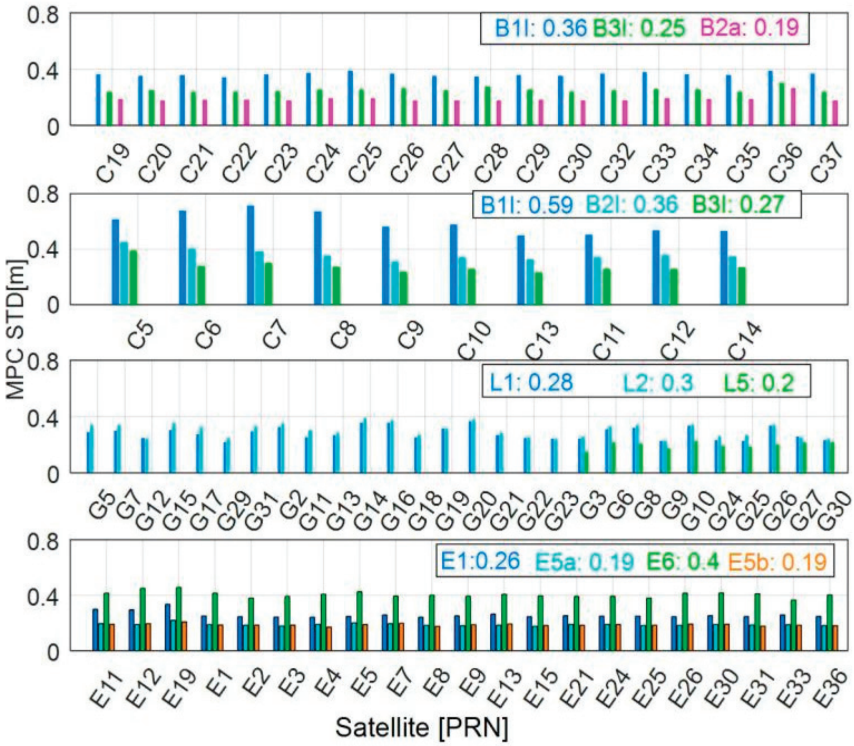
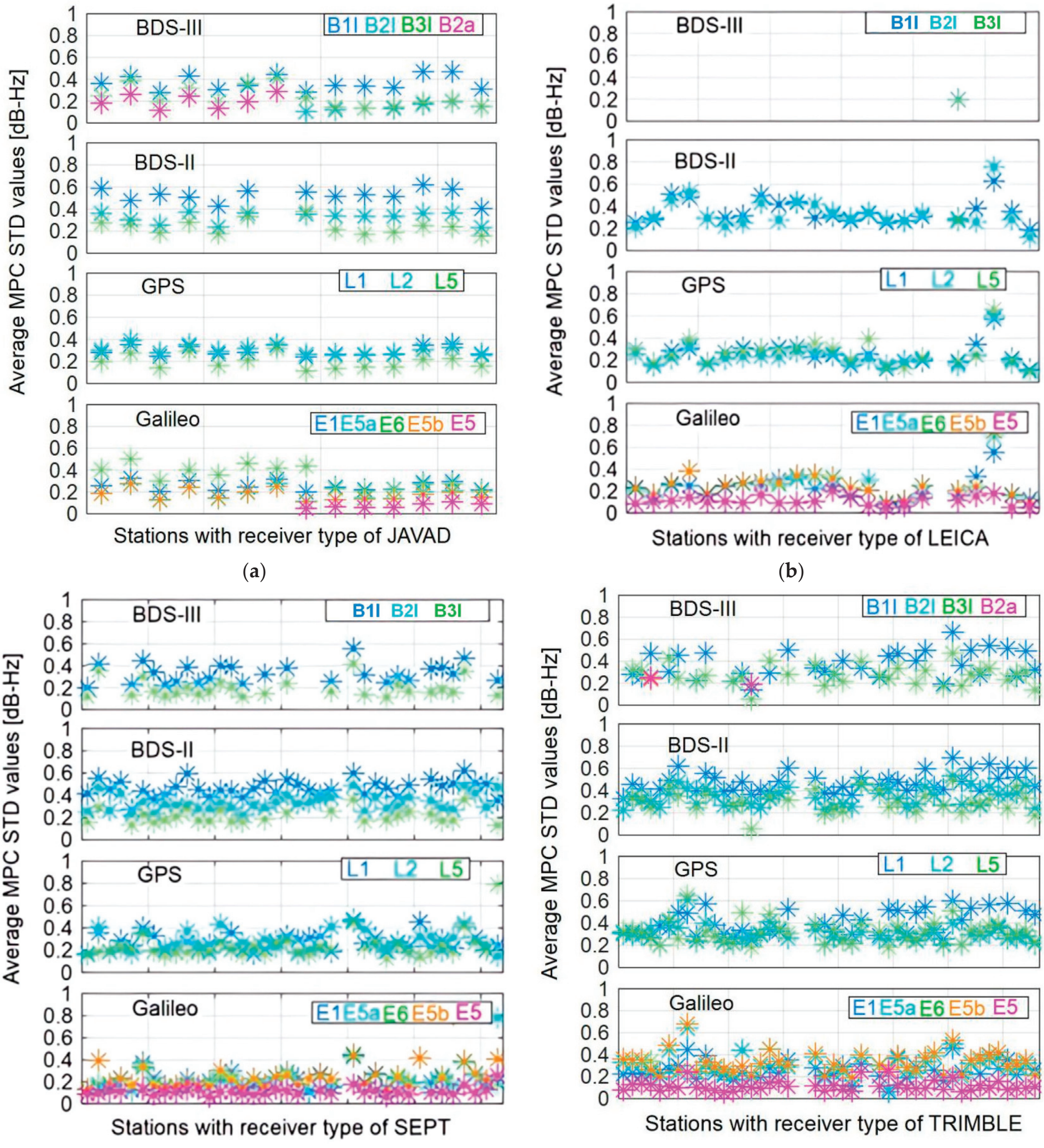


Figure 10. STDs of multipath combinations for different satellites. The four panels from top to bottom are for BDS-3, BDS-2, GPS, and Galileo satellites, respectively.

Similar to the comparison of *C/N0* and CC values for each station, the STD values of the multipath combinations for each type of receiver at different signals are shown in Figure 11. Similar characteristics can be found, namely that the multipath values have little relationship with the receiver type. Similarly, the Galileo E5 signal exhibits superior performance compared to all other signals, regardless of receiver type.



(**a**)

(**b**)

(**c**)

(**d**)

Figure 11. Comparison of average MPC STD values in different stations. The horizontal axis in subfigure (**a**) represents 15 stations with receiver type JAVAD, while that in subfigure (**b**) represents 24 stations with receiver type LEICA, that in subfigure (**c**) represents 29 stations with receiver type SEPT, and that in subfigure (**d**) represents 40 stations with receiver type TRIMBLE. In each subgraph, the 4 panels are for BDS-3 B1I/B3I/B2a, BDS-2 B1I/B2I/B3I, GPS L1/L2/L5, and Galileo E1/E5a/E6/E5b/E5, respectively.

4. Discussion

By using data from different types of receivers and constructing models to evaluate signal power, code noise, and multipath effects, the satellite and signal performance of BDS and other systems such as GPS, Galileo were analyzed and compared.

The receiver type has little influence on *C/N0* values. For BDS, signal powers of BDS-3 satellites are approximate 2–3 dB-Hz higher than those of BDS-2 satellites at the same frequency, which may be related to its advance modulation, and B2a of BDS-3, E5 of Galileo, and L5 of GPS have stronger signal powers.

The pseudorange noise levels of different signals have little relationship with the receiver type. It is worth noting that the Galileo E5 signal exhibits optimal performance due to its high signal power. The observation accuracy of E5 can reach 0.2 m, which is the highest among all the signals, regardless of receiver types, and next highest are BDS-3 B2a and GPS L5.

Similar characteristics can be found, namely that the multipath values have little relationship with the receiver type. Not being affected by the systematic code errors of BDS-2, the observations of BDS-3 satellites contain smaller multipath errors than that of BDS-2 satellites. As for the multipath suppression performance, the BDS-3 signal B2a, GPS L5, and Galileo E5b perform better than the other signals, which may be related to their wide signal bandwidths. The Galileo E5 signal exhibits superior performance compared to all other signals, regardless of receiver type.

In conclusion, the signals B2a, E5, and L5 exhibit advantages over the other signals, which may be attributed to the advanced signal modulation method.

5. Conclusions

A comprehensive evaluation of the signal characteristics of BDS, Galileo, and GPS was conducted using data from four types of receivers of MGEX stations. The following conclusion can be drawn.

In terms of signal power, the BDS-3 satellites have better manifestation than the BDS-2 satellites, and the signal B2a is better than B1I and B3I, while E5 of Galileo and L5 of GPS perform better than the other signals of the respective systems. The pseudorange noise exhibits characteristics of periodicity and variation with satellite elevation. The BDS-3 B2a signal possesses smaller noise than the other signals of BDS. The signal L5 performs better than the other signals of GPS, while the Galileo E5 signal performs best among all the signals of the three systems, regardless of receiver types. The MP error of BDS-3 is on the same level as that of GPS and Galileo, which is apparently smaller than that of BDS-2, and the systematic biases which exist in the pseudorange observations of BDS-2 are absent for BDS-3. Similar to the characteristics of the CC values, the BDS-3 B2a, GPS L5, and Galileo E5b have significantly smaller multipath errors than other signals. In general, the C/N0, CC, and multipath values have little relationship with the receiver type, but they affect each other.

In this paper, although some preliminary conclusions and research results have been achieved, there are still the following studies that need to be further deepened and expanded, due to system development and personal conditions, etc. The paper mainly focuses on the data quality between different types of satellites and different frequency observations of BDS/Galileo/GPS, but it does not conduct a detailed study of the observation stochastic model based on this. With the availability of BDS-3 and more Galileo FOC satellite observations, the establishment of a stochastic model that reflects the differences in the stochastic characteristics of different observations based on the data quality analysis to achieve high-precision-combined multi-GNSS navigation and positioning will become the key of the next research.

Author Contributions: Conceptualization, Y.T. and G.X.; methodology, Y.T. and R.G.; software, Y.T. and D.Z.; validation, L.Z. and J.G.; formal analysis, J.X.; investigation, Y.H.; resources, X.D.; writing—original draft preparation, Y.T.; writing—review and editing, D.H. and Z.Q.; visualization, J.X. and L.Z.; supervision, D.Z.; project administration, G.X.; funding acquisition, R.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (grant nos. 42274045, 42004028 and 41904039), Natural Science Foundation of Henan (grant no. 232300421105) and China Postdoctoral Science Foundation (grant no. 2023M733285).

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request. The data can be found at the IGS official website (http://www.igs.org/mgex/data-products/#data, accessed on 2 April 2024).

Acknowledgments: The authors express great gratitude to IGS and iGMAS for providing the GNSS data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. ILRS BeiDou (COMPASS) Center of Mass Information. Available online: https://ilrs.gsfc.nasa.gov/missions/satellite_missions/current_missions/bm13_general.html (accessed on 12 April 2024).
2. Yang, Y.; Yang, Y.; Hu, X.; Tang, C.; Guo, R.; Zhou, Z.; Xu, J.; Pan, J.; Su, M. BeiDou-3 broadcast clock estimation by integration of observations of regional tracking stations and inter-satellite links. *GPS Solut.* **2021**, *25*, 57. [CrossRef]
3. CSNO. Report on the Development of BeiDou Navigation Satellite System (Version 2.1). Available online: http://www.beidou.gov.cn/xt/gfxz/201805/P020180509588857259014.pdf (accessed on 10 December 2021).
4. Gu, S.; Wang, Y.; Zhao, Q.; Zheng, F.; Gong, X. BDS-3 differential code bias estimation with undifferenced uncombined model based on triple-frequency observation. *J. Geod.* **2020**, *94*, 45. [CrossRef]
5. CSNO. Development of BeiDou Navigation Satellite System (Version 4.0). Available online: http://www.beidou.gov.cn/xt/gfxz/201912/P020191227430565455478.pdf (accessed on 10 December 2021).
6. Li, X.; Li, X.; Liu, G.; Yuan, Y.; Freeshah, M.; Zhang, K.; Zhou, F. BDS multi-frequency PPP ambiguity resolution with new B2a/B2b/B2a + b signals and legacy B1I/B3I signals. *J. Geod.* **2020**, *94*, 107. [CrossRef]
7. Zhang, X.; Li, X.; Lu, C.; Wu, M.; Pan, L. A comprehensive analysis of satellite-induced code bias for BDS-3 satellites and signals. *Adv. Space Res.* **2019**, *63*, 2822–2835. [CrossRef]
8. Zhang, Z.; Li, B.; Nie, L.; Wei, C.; Jia, S.; Jiang, S. Initial assessment of BeiDou-3 global navigation satellite system: Signal quality, RTK and PPP. *GPS Solut.* **2019**, *23*, 1–12. [CrossRef]
9. Lu, M.; Li, W.; Yao, Z. Overview of BDS III new signals. *Navigation* **2019**, *66*, 19–35. [CrossRef]
10. China Satellite Navigation System Administration Office. BDS Satellite Navigation System Space Signal Interface Control Document Open Service Signal B1C (Version 1.0) [EB/OL][R]. 27 December 2017. Available online: http://www.beidou.gov.cn/xt/gfxz/201712/P020171226741342013031.pdf (accessed on 18 May 2022).
11. China Satellite Navigation System Administration Office (CSNO). BDS Satellite Navigation System Space Signal Interface Control Document Open Service Signal B2a (Version 1.0) [EB/OL][R]. 27 December 2017. Available online: http://www.beidou.gov.cn/xt/gfxz/201712/P020171226742357364174.pdf (accessed on 18 May 2022).
12. China Satellite Navigation System Administration Office (CSNO). BDS Satellite Navigation System Space Signal Interface Control Document Open Service Signal B3I (Version 1.0) [EB/OL][R]. 9 February 2018. Available online: http://www.beidou.gov.cn/xt/gfxz/201802/P020180209623601401189.pdf (accessed on 18 May 2022).
13. China Satellite Navigation System Administration Office (CSNO). BDS Satellite Navigation System Space Signal Interface Control Document Open Service Signal B1I (Version 1.0) [EB/OL][R]. 27 February 2019. Available online: http://www.beidou.gov.cn/xt/gfxz/201902/P020190227593621142475.pdf (accessed on 18 May 2022).
14. Gaglione, S.; Angrisano, A.; Castaldo, G.; Freda, P.; Gioia, C.; Innac, A. The first Galileo FOC satellites: From useless to essential. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium, Milan, Italy, 26–31 July 2015; IEEE: Piscataway, NJ, USA, 2015.
15. Montenbruck, O.; Hauschild, A.; Steigenberger, P.; Hugentobler, U.; Teunissen, P.; Nakamura, S. Initial assessment of the compass/BDS-2 regional navigation satellite system. *GPS Solut.* **2013**, *17*, 211–222. [CrossRef]
16. Hauschild, A.; Montenbruck, O.; Sleewaegen, J.; Huisman, L.; Teunissen, P. Characterization of Compass M-1 Signals. *GPS Solut.* **2012**, *16*, 117–126. [CrossRef]
17. Wang, G.; Jong, K.; Zhao, Q.; Hu, Z.; Guo, J. Multipath analysis of code measurements for BDS geostationary satellites. *GPS Solut.* **2015**, *19*, 129–139. [CrossRef]
18. Wanninger, L.; Beer, S. BDS satellite-induced code pseudorange variations: Diagnosis and therapy. *GPS Solut.* **2015**, *19*, 639–648. [CrossRef]
19. Zaminpardaz, S.; Teunissen, P. Analysis of Galileo IOV + FOC signals and E5 RTK performance. *GPS Solut.* **2017**, *21*, 1855–1870. [CrossRef]
20. Pan, L.; Zhang, X.; Liu, J.; Li, X.; Li, X. Analysis and correction of the inter-frequency clock bias for BDS satellites. In *China Satellite Navigation Conference (CSNC) 2016 Proceedings*; Springer: Singapore, 2016; Volume II, pp. 115–128.
21. Yang, Y.; Li, J.; Wang, A.; Xu, J.; He, H.; Guo, H.; Shen, J.; Dai, X. Preliminary assessment of the navigation and positioning performance of BDS regional navigation satellite system. *Sci. China Earth Sci.* **2014**, *57*, 144–152. [CrossRef]

22. Tian, Y.; Sui, L.; Xiao, G.; Zhao, D.; Tian, Y. Analysis of BDS/Galileo/GPS signals and RTK performance. *GPS Solut.* **2019**, *23*, 37. [CrossRef]
23. Simsky, A.; Sleewaegen, J.; Crisci, M. Performance assessment of Galileo ranging signals transmitted by GSTB-v2 satellites. In Proceedings of the 19th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2006), Fort Worth, TX, USA, 26–29 September 2006; pp. 1547–1559.
24. Zhang, X.; Wu, M.; Liu, W.; Li, X.; Yu, S.; Lu, C. Initial assessment of the COMPASS/BDS-3: New-generation navigation signals. *J. Geod.* **2017**, *91*, 1225–1240. [CrossRef]
25. Fu, Z.; Gong, X.; Gu, S.; Lou, Y.; Shi, S. Accounting for biases between BDS-3 and BDS-2 overlapping B1I/B3I signals in BeiDou global ionospheric modeling and DCB determination. *Adv. Space Res.* **2022**, *69*, 3677–3691.
26. Mi, X.; Sheng, C.; ElMowafy, A.; Zhang, B. Characteristics of receiver-related biases between BDS-3 and BDS-2 for five frequencies including inter-system biases, differential code biases, and differential phase biases. *GPS Solut.* **2021**, *25*, 3. [CrossRef]
27. Liang, Y.; Xu, J.; Wu, M.; Li, F. Analysis of the Long-Term Characteristics of BDS On-Orbit Satellite Atomic Clock: Since BDS-3 Was Officially Commissioned. *Remote Sens.* **2022**, *14*, 4535. [CrossRef]
28. Wang, E.; Song, W.; Zhang, Y.; Shi, X.; Wang, Z.; Xu, S.; Shu, W. Evaluation of BDS/GPS Multi-Frequency RTK Positioning Performance under Different Baseline Lengths. *Remote Sens.* **2022**, *14*, 3561. [CrossRef]
29. Yang, Y.; Mao, Y.; Sun, B. Basic performance and future developments of BeiDou global navigation satellite system. *Satell. Navig.* **2020**, *1*, 1. [CrossRef]
30. Shi, J.; Ouyang, C.; Huang, Y.; Peng, W. Assessment of BDS-3 global positioning service: Ephemeris, SPP, PPP, RTK, and new signal. *GPS Solut.* **2020**, *24*, 81. [CrossRef]
31. Gu, S.; Dai, C.; Fang, W.; Zheng, F.; Wang, Y.; Zhang, Q.; Lou, Y.; Niu, X. Multi-GNSS PPP/INS tightly coupled integration with atmospheric augmentation and its application in urban vehicle navigation. *J. Geod.* **2021**, *95*, 64. [CrossRef]
32. Ma, Z.; Cui, J.; Liu, Z.; Su, X.; Xiang, Y.; Xu, Y.; Deng, C.; Hui, M.; Li, Q. Influence of Inter-System Biases on Combined Single-Frequency BDS-2 and BDS-3 Pseudorange Positioning of Different Types of Receivers. *Remote Sens.* **2024**, *16*, 1710. [CrossRef]
33. Dang, X.; Yin, X.; Zhang, Y.; Gao, C.; Wu, J.; Liu, Y. Improved Medium Baseline RTK Positioning Performance Based on BDS/Galileo/GPS Triple-Frequency only Observations. *Remote Sens.* **2023**, *15*, 5198. [CrossRef]
34. Xiao, K.; Sun, F.; Zhu, X.; Zhou, P.; Ma, Y.; Wang, Y. Assessment of overlapping triple-frequency BDS-3/BDS-2/INS tightly coupled integration model in kinematic surveying. *GPS Solut.* **2024**, *28*, 2. [CrossRef]
35. Shu, B.; Tian, Y.; Qu, X.; Li, W.; Huang, G.; Du, Y.; Zhang, Q. Estimation of BDS-2/3 phase observable-specific signal bias aided by double-differenced model: An exploration of fast BDS-2/3 real-time PPP. *GPS Solut.* **2024**, *28*, 88. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

Integration of High-Rate GNSS and Strong Motion Record Based on Sage–Husa Kalman Filter with Adaptive Estimation of Strong Motion Acceleration Noise Uncertainty

Yuanfan Zhang [1], Zhixi Nie [1,*], Zhenjie Wang [1], Guohong Zhang [2] and Xinjian Shan [2]

[1] College of Oceanography and Space Informatics, China University of Petroleum, Qingdao 266580, China; z18010038@s.upc.edu.cn (Y.Z.); sdwzj@upc.edu.cn (Z.W.)

[2] Key Laboratory of Earthquake Dynamics, Institute of Geology, China Earthquake Administration, Beijing 100029, China; zhanggh@ies.ac.cn (G.Z.); xjshan@ies.ac.cn (X.S.)

* Correspondence: niezhixi@upc.edu.cn

Abstract: A strong motion seismometer is a kind of inertial sensor, and it can record middle- to high-frequency ground accelerations. The double-integration from acceleration to displacement amplifies errors caused by tilt, rotation, hysteresis, non-linear instrument response, and noise. This leads to long-period, non-physical baseline drifts in the integrated displacements. GNSS enables the direct observation of the ground displacements, with an accuracy of several millimeters to centimeters and a sample rate of 1 Hz to 50 Hz. Combining GNSS and a strong motion seismometer, one can obtain an accurate displacement series. Typically, a Kalman filter is adopted to integrate GNSS displacements and strong motion accelerations, using the empirical values of noise uncertainty. Considering that there are significantly different errors introduced by the above-mentioned tilt, rotation, hysteresis, and non-linear instrument response at different stations or at different times at the same station, it is inappropriate to employ a fixed noise uncertainty for strong motion accelerations. In this paper, we present a Sage–Husa Kalman filter, where the noise uncertainty of strong motion acceleration is adaptively estimated, to integrate GNSS and strong motion acceleration for obtaining the displacement series. The performance of the proposed method was validated by a shake table simulation experiment and the GNSS/strong motion co-located stations collected during the 2023 Mw 7.8 and Mw 7.6 earthquake doublet in southeast Turkey. The experimental results show that the proposed method enhances the adaptability to the variation of strong motion accelerometer noise level and improves the precision of integrated displacement series. The displacement derived from the proposed method was up to 28% more accurate than those from the Kalman filter in the shake table test, and the correlation coefficient with respect to the references arrived at 0.99. The application to the earthquake event shows that the proposed method can capture seismic waveforms at a promotion of 46% and 23% in the horizontal and vertical directions, respectively, compared with the results of the Kalman filter.

Keywords: high-rate GNSS; strong motion; Sage–Husa Kalman filter; coseismic displacement

Citation: Zhang, Y.; Nie, Z.; Wang, Z.; Zhang, G.; Shan, X. Integration of High-Rate GNSS and Strong Motion Record Based on Sage–Husa Kalman Filter with Adaptive Estimation of Strong Motion Acceleration Noise Uncertainty. *Remote Sens.* **2024**, *16*, 2000. https://doi.org/10.3390/rs16112000

Academic Editor: Michael E. Gorbunov

Received: 6 May 2024
Revised: 25 May 2024
Accepted: 30 May 2024
Published: 1 June 2024

1. Introduction

Traditional seismic wave recording instruments mainly include seismometers and strong motion seismometers, and both have high sampling rates and a good sensitivity. However, during strong earthquakes, the velocity records of near-field seismometers are prone to saturation limits [1]. A strong motion seismometer is a kind of inertial sensor that can record middle- to high-frequency strong ground motion without saturation. However, the low-frequency noises introduced by tilt, rotation, hysteresis, non-linear instrument response, and measurement noise are not easy to solve [2]. These low-frequency noises are additionally amplified when integrating the acceleration records to velocities and displacements and this leads to non-physical baseline drifts. The drift becomes larger and larger with the increase of integration time, especially for the great seismological events [3]. The typical method involves applying high-pass filtering to remove low-frequency errors, but the trend component is weakened. As a result, it is unable to obtain accurate static displacements, and the maximum ground displacement is also greatly reduced. Alternatively, some empirical baseline correction schemes have been proposed where the start time and duration of different stages are usually determined by an empirical acceleration threshold. By taking the zero velocities before start time and after end time as constraints, corrections for the baseline shifts are estimated through piece-wise fitting [4–8]. However, the baseline shift does not always accompany the strongest ground shaking. The threshold-based methods tend to lead to an over- or underestimation of the true baseline shift. Other correction schemes, which are performed by manual calibration, rely on subjective determinations for the choice of correction parameters [9].

The Global Navigation Satellite System (GNSS) enables the direct observation of ground displacements, with an accuracy of several millimeters to centimeters and a sample rate of 1 Hz to 50 Hz [10]. Compared with seismic sensors, high-frequency GNSS can obtain permanent displacement information of the Earth's surface exempt from measuring saturations and baseline drifts [11,12]. Initially, the GNSS displacement retrieval methods depend on relative positioning (RP) [13,14]. However, the RP method only obtained relative displacements regarding a reference station. For a large earthquake, there are also seismic waveforms in the reference station [15]. Compared with the RP, precise point positioning (PPP) can provide absolute seismic waveforms alone [16,17]. Nevertheless, it has limited accuracy owing to unresolved integer cycle ambiguities. In recent years, precise point positioning with ambiguity resolution (PPP-AR) has been developed to improve the positioning accuracy of the PPP method [18,19]. It can provide rival accuracy to that of the RP method by applying precise ephemeris, uncalibrated phase delay (UPD), or fractional cycle bias (FCB) products [20]. However, the limitation of PPP-AR is that a (re)convergence period of tens of minutes is needed. The accuracy of the PPP-derived/PPP-AR-derived coseismic displacement might be decreased when an earthquake happens, by coincidence, during the PPP/PPP-AR (re)convergence period [21]. To overcome the (re)convergence of PPP/PPP-AR, some time-differential algorithms such as the variometric approach (VA) and temporal point positioning (TPP) have been developed for the retrieval of seismic waves [22–29]. By only applying broadcast ephemeris, the VA method can directly obtain seismic velocity waveforms based on epoch-differenced phase measurements. But the integration process from the velocities to the displacements is subject to accumulation errors. The TPP method adopts temporal-differenced phase measurements between a reference epoch and the current epoch, and there is almost no shift in the derived displacement waveforms. It should be noted that the precise ephemeris must be applied in the TPP method. Regardless of the method of GNSS displacement retrievals, such as RP, PPP/PPP-AR, VA, and TPP, GNSS is systematically noisier than seismic sensors mainly due to residual atmosphere interference, multipath and other errors, and it also has a relatively lower sampling rate [30,31].

The integration of GNSS and strong motion seismometer can harness the inherent strengths of both GNSS and accelerometer sensor. This fusion enhances their complementarity, particularly in P-wave arrival distinguishing, rapid magnitude estimation, and earthquake early warning, especially for large earthquake events [32,33]. However, the baseline drifts in the displacements obtained from strong motion records are the main challenge in the combination of GNSS and strong motion [34]. Emore et al. (2007) developed an inversion method to simultaneously estimate the baseline shifts and ultimate displacements with the constraints imposed by GNSS displacements [2]. Wang et al. (2013) modeled the baseline shifts by a linear function combined with a sinusoidal series. The baseline shifts were estimated to achieve the best fit with the GNSS displacement and then they could be eliminated from the strong motion record [35]. Taking the GNSS-derived displacements as a reference, Tu et al. (2013) extracted the baseline shift in the accelerometer record by employing a smoother. The smoothing process, akin to a low-pass filter, effectively filters out high-frequency seismic signals [36]. With the physical constraints derived from high-rate

GNSS deformations, the peak ground displacement and static permanent displacements are maximally preserved [37].

Unlike the above-mentioned integration methods at the level of displacements, Smyth and Wu (2007) proposed a method to combine low-sampling-rate displacements and high-sampling-rate acceleration records to obtain very broadband waveforms by a multi-rate Kalman filter [38]. Bock et al. (2011) presented the combination of GNSS RP displacements and strong motion accelerations to retrieve the broadband coseismic waveforms of the 2010 Mw 7.2 EI Mayor–Cucapah earthquake [39]. Song and Xu (2018) modified the filter model to reduce the influence of GNSS-colored noise [40]. Shu et al. (2018) also improved the integrated model to solve the aliasing problem in GNSS-derived waveforms with an application to the 2016 Mw 7.8 Kaikoura earthquake [41]. In these schemes, strong-motion acceleration records were regarded as input signals to the state equation, and the baseline shift of acceleration and the measurement noise of strong motion seismometer were unified as process noise. Apart from these schemes, Geng et al. (2013) proposed a tight integration method in which GNSS measurements and strong motion accelerations were directly integrated into the filter process of PPP-AR [42]. An extra parameter was introduced to estimate the baseline shift, and it was modeled as a random walk. Li et al. (2013) demonstrated that precise dynamical information provided by strong motion acceleration records provides a tight constraint to improve the PPP-AR solution strength [43]. Tu et al. (2014) refined the tight integration model by utilizing the estimated baseline shift from the previous epoch to correct the acceleration of the current epoch. This refinement led to improved integration outcomes [44]. Guo et al. (2021) proposed a loose integration model that incorporates a virtual acceleration parameter for estimating baseline shift corrections. This estimation of virtual acceleration was treated as a random walk process [45].

In the aforementioned Kalman filter-based methods, a fixed value is usually used to account for the strong motion acceleration noise uncertainty, and this value is determined by computing the variance of pre-event strong motion acceleration records [39–48]. Nevertheless, the rise in noise level due to baseline shifts during the strong motion period was not accounted for in the estimation of the ambient pre-event acceleration noise. Therefore, an acceleration variance multiplier is necessary. If the noise level of strong motion acceleration is too small, it cannot reflect the absorbed baseline shift in the noise of strong motion acceleration. While the noise level is too large, it is impossible to effectively utilize the information of the strong motion acceleration. Considering that there are significantly different errors introduced by the above-mentioned tilt, rotation, hysteresis, and non-linear instrument response at different stations or at different times at the same station, it is inappropriate to employ a fixed noise uncertainty for strong motion accelerations in multi-rate Kalman filter.

In this contribution, we present a method to integrate high-rate GNSS and strong motion records based on the Sage–Husa Kalman filter with an adaptive estimation of strong motion acceleration noise uncertainty. In the proposed method, the noise uncertainty of strong motion acceleration is adaptively determined utilizing the Sage–Husa sliding-window estimation principle, at the same time the effect of the baseline shift is accommodated through an adaptive variance inflation. This method significantly improves the accuracy of the system's process noise representation and thereby improving filter performance. The paper is organized as follows: At first, the traditional Kalman filter method is presented. Then, the proposed integration method of high-rate GNSS and strong motion records based on the Sage–Husa Kalman filter is discussed in detail. Finally, the proposed method was verified through a shake table simulation experiment and an application to the 2023 Mw 7.8 and Mw 7.6 earthquake doublet in southeast Turkey.

2. Methodology

2.1. Traditional Kalman Filter Method

As for the estimation of ground displacement at an observing station, we can assume three-dimensional (3D) motion and formulate the problem independently for each coordinate direction as a first-order linear differential equation using the continuous state-space representation following Bock et al. [39]. The system model of 3D motion in the discrete form is given as follows:

$$\boldsymbol{X}_k = \boldsymbol{\Phi}_{k/k-1}\boldsymbol{X}_{k-1} + \boldsymbol{B}_{k/k-1}\boldsymbol{a}_{k/k-1} + \boldsymbol{w}_k \tag{1}$$

where $\boldsymbol{X}_k = \begin{bmatrix} \boldsymbol{d}_k & \boldsymbol{v}_k \end{bmatrix}^T$ represents the state parameter composed of the 3D displacement vector $\boldsymbol{d}_k$ and the velocity vector $\boldsymbol{v}_k$, $\boldsymbol{\Phi}_{k/k-1} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \tau_a \boldsymbol{I}_{3\times3} \\ 0_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix}$ is the transition matrix from epoch $k-1$ to k, $\boldsymbol{a}_{k/k-1}$ is the system input vector defined as the raw 3D acceleration record in this contribution, $\boldsymbol{B}_{k/k-1} = \begin{bmatrix} 0.5\tau_a^2 \boldsymbol{I}_{3\times3} & \tau_a \boldsymbol{I}_{3\times3} \end{bmatrix}^T$ is the input control matrix, $\boldsymbol{w}_{k/k-1}$ is the system noise vector with a Gaussian distribution $\boldsymbol{w}_{k/k-1} \sim N\left(0, \boldsymbol{Q}_{\boldsymbol{w}_{k/k-1}}\right)$, and $\boldsymbol{Q}_{\boldsymbol{w}_{k/k-1}} = \begin{bmatrix} \frac{q}{3}\tau_a^3 \boldsymbol{I}_{3\times3} & \frac{q}{2}\tau_a^2 \boldsymbol{I}_{3\times3} \\ \frac{q}{2}\tau_a^2 \boldsymbol{I}_{3\times3} & q\tau_a \boldsymbol{I}_{3\times3} \end{bmatrix}$ stands for the covariance matrix of the acceleration process noise. Here, τ_a is the sampling interval of the accelerometer, q is the acceleration noise uncertainty, $\boldsymbol{I}_{3\times3}$ represents the three-row by three-column (3 by 3) unit matrix, and $\boldsymbol{0}_{3\times3}$ stands for the 3 by 3 zero matrix.

The measurement for the GNSS displacement in the local coordinate frame is defined as follows:

$$\boldsymbol{L}_k = \boldsymbol{H}_k \boldsymbol{X}_k + \boldsymbol{\Lambda}_k \tag{2}$$

where $\boldsymbol{L}_k$ is a column vector including the pre-processed GNSS displacement in each direction, $\boldsymbol{H}_k = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \end{bmatrix}$ is the design matrix, $\boldsymbol{\Lambda}_k$ is the GNSS displacement noise with a Gaussian distribution $\boldsymbol{\Lambda}_{\mathrm{k}} \sim N(0, \boldsymbol{R}_k)$, and $\boldsymbol{R}_k$ is the covariance matrix of the GNSS displacement noise. Typically, covariance matrices $\boldsymbol{R}_k$ and $\boldsymbol{Q}_{\boldsymbol{w}_{k/k-1}}$, for GNSS displacements and strong motion acceleration records, are empirically determined by pre-event noise.

The Kalman filter is composed of two steps, the time update for the system model and the measurement update for the measurement model [49]. The time update can be expressed as follows:

$$\boldsymbol{X}_k^- = \boldsymbol{\Phi}_{k/k-1}\hat{\boldsymbol{X}}_{k-1} + \boldsymbol{B}_{k/k-1}\boldsymbol{a}_{k/k-1} \tag{3}$$

$$\boldsymbol{P}_k^- = \boldsymbol{\Phi}_{k/k-1}\boldsymbol{P}_{k-1}\boldsymbol{\Phi}_{k/k-1}^T + \boldsymbol{Q}_{\boldsymbol{w}_{k/k-1}} \tag{4}$$

where $\boldsymbol{X}_{\boldsymbol{k}}^-$ is the predicted state with covariance matrix $\boldsymbol{P}_k^-$ at current epoch k, and $\hat{\boldsymbol{X}}_{k-1}$ is the estimated state with the covariance matrix $\boldsymbol{P}_{k-1}$ at the previous epoch of $k-1$.

The measurement update can be written as follows:

$$\boldsymbol{K}_k = \boldsymbol{P}_k^- \boldsymbol{H}_k^T \left(\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{R}_k\right)^{-1} \tag{5}$$

$$\hat{\boldsymbol{X}}_k = (\boldsymbol{I}_{6\times6} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{X}_k^- + \boldsymbol{K}_k\boldsymbol{L}_k \tag{6}$$

$$\boldsymbol{P}_k = (\boldsymbol{I}_{6\times6} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{P}_k^- \tag{7}$$

where $\boldsymbol{K}_k$ is the gain matrix, $\hat{\boldsymbol{X}}_k$ is the estimated state with the covariance matrix $\boldsymbol{P}_k$, $\boldsymbol{I}_{6\times6}$ stands for the 6 by 6 unit matrix. Due to the varying sampling rates of GNSS displacement and strong motion acceleration, the time updates of Equations (3) and (4) are performed upon each accelerometer sampling, while measurement updates of Equations (5)–(7) are performed at each GNSS sampling [50].

2.2. A Sage–Husa Kalman Filter Method with Adaptive Estimation of Strong Motion Acceleration Noise Uncertainty

Considering that there are significantly different errors introduced by the above-mentioned tilt, rotation, hysteresis, and non-linear instrument response at different stations or at different times at the same station, it is inappropriate to employ a fixed noise uncertainty for strong motion accelerations. To address this issue, a Sage–Husa Kalman filter method with adaptive estimation of strong motion acceleration noise uncertainty is proposed. In the proposed method, the noise uncertainty is adaptively estimated based on the Sage–Husa sliding-window estimation principle, at the same time the effect of baseline shift is accommodated through adaptive variance inflation.

Adaptive estimation of noise uncertainty allows the filter to dynamically adjust to changes in measurement conditions, making it suitable for applications where traditional fixed noise uncertainty assumptions may be inadequate. Consideration of baseline shift allows for a more accurate representation of the system's process noise, and thereby enhances the filter performance.

The predicted residual of the state vector after the measurement update reads as follows:

$$\boldsymbol{V}_{\boldsymbol{X}_k^-} = \hat{\boldsymbol{X}}_k - \boldsymbol{X}_k^- \tag{8}$$

Utilizing the law of variance–covariance propagation, the covariance matrix of the predicted residual can be expressed as follows:

$$\boldsymbol{Q}_{\boldsymbol{V}_{\boldsymbol{X}_k^-}} = Cov\left(\hat{\boldsymbol{X}}_k - \boldsymbol{X}_k^-\right) = Cov\left(\hat{\boldsymbol{X}}_k\right) + Cov\left(\boldsymbol{X}_k^-\right) - 2Cov\left(\hat{\boldsymbol{X}}_k, \boldsymbol{X}_k^-\right) \tag{9}$$

where $Cov(\cdot)$ represents the covariance operator. The covariance matrixes of $\hat{\boldsymbol{X}}_k$ and $\boldsymbol{X}_k^-$, and their cross-term can be calculated as follows:

$$\begin{cases} Cov\left(\hat{\boldsymbol{X}}_k\right) = \boldsymbol{P}_k \\ Cov\left(\boldsymbol{X}_k^-\right) = \boldsymbol{P}_k^- \\ Cov\left(\hat{\boldsymbol{X}}_k, \boldsymbol{X}_k^-\right) = \boldsymbol{P}_k^- - \boldsymbol{K}_k\boldsymbol{H}_k\boldsymbol{P}_k^- \end{cases} \tag{10}$$

Combining Equation (9) with Equation (10), the covariance of the prediction residual can be reformulated as follows:

$$\boldsymbol{Q}_{\boldsymbol{V}_{\boldsymbol{X}_k^-}} = \boldsymbol{P}_k + \boldsymbol{P}_k^- - 2\left(\boldsymbol{P}_k^- - \boldsymbol{K}_k\boldsymbol{H}_k\boldsymbol{P}_k^-\right) = \boldsymbol{P}_k^- - \boldsymbol{P}_k \tag{11}$$

With $\boldsymbol{P}_k = \boldsymbol{Q}_{\hat{\boldsymbol{X}}_k}$ defined, and substituting Equation (4) into Equation (11), the covariance of the prediction residual is calculated as follows:

$$\boldsymbol{Q}_{\boldsymbol{V}_{\boldsymbol{X}_k^-}} = \boldsymbol{Q}_{\boldsymbol{w}_{k/k-1}} + \boldsymbol{\Phi}_{k,k-1}\boldsymbol{Q}_{\hat{\boldsymbol{X}}_{k-1}}\boldsymbol{\Phi}_{k,k-1}^T - \boldsymbol{Q}_{\hat{\boldsymbol{X}}_k} \tag{12}$$

Given the expectation $E\left(\boldsymbol{V}_{\boldsymbol{X}_k^-}\right) = 0$, the covariance matrix of $\boldsymbol{V}_{\boldsymbol{X}_k^-}$ can be approximately calculated by using historical predicted residuals in the sliding-window [51]:

$$\hat{\boldsymbol{Q}}_{\boldsymbol{V}_{\boldsymbol{X}_k^-}} \approx \frac{1}{m}\sum_{i=0}^{m-1} \boldsymbol{V}_{\boldsymbol{X}_{k-i}^-}\boldsymbol{V}_{\boldsymbol{X}_{k-i}^-}^T \tag{13}$$

where m stands for the number of acceleration records in the sliding-window.

When the system model maintains stability and consistency within the sliding window, the historical predicted residuals encapsulated by this window should accurately mirror the process noise level of the current state. Consequently, by integrating Equation (13) into Equation (12), the covariance matrix of process noise can also be approximately calculated as follows:

$$\hat{\boldsymbol{Q}}_{\boldsymbol{w}_{k/k-1}} \approx \frac{1}{m}\sum_{i=0}^{m-1} \boldsymbol{V}_{\boldsymbol{X}_{k-i}^-}\boldsymbol{V}_{\boldsymbol{X}_{k-i}^-}^T - \boldsymbol{\Phi}_{k,k-1}\boldsymbol{Q}_{\hat{\boldsymbol{X}}_{k-1}}\boldsymbol{\Phi}_{k,k-1}^T + \boldsymbol{Q}_{\hat{\boldsymbol{X}}_k} \tag{14}$$

Such an estimation encompasses both acceleration measurement noise and the impact of strong motion acceleration baseline shift. Historical predicted residuals in the recent period of time provide a more accurate reference for the noise level of strong motion acceleration. The $\hat{\boldsymbol{Q}}_{w_{k/k-1}}$ is introduced to encapsulate characteristics of the acceleration noise uncertainty, and can be represented as a block matrix:

$$\hat{\boldsymbol{Q}}_{w_{k/k-1}} = \begin{bmatrix} \hat{\boldsymbol{q}}_1 & \hat{\boldsymbol{q}}_2 \\ \hat{\boldsymbol{q}}_3 & \hat{\boldsymbol{q}}_4 \end{bmatrix} \tag{15}$$

where $\hat{\boldsymbol{q}}_i (i = 1, 2, 3, 4)$ are 3 by 3 matrices. The trace of $\hat{\boldsymbol{q}}_4$ is represented by $tr(\hat{\boldsymbol{q}}_4)$, which is a measure scale of the overall noise uncertainty of strong motion accelerations between two adjacent GNSS observation epochs.

Assuming that there are n acceleration records between two adjacent GNSS observation epochs and the effect of a baseline shift for each acceleration record between two adjacent high-rate GNSS observation epochs is approximately uniform, the noise uncertainty of strong motion acceleration can be approximated as follows:

$$\tilde{q} \approx \frac{1}{3n} tr(\hat{\boldsymbol{q}}_4) \tag{16}$$

The covariance matrix of the process noise for the next epoch is calculated by using adaptive estimation of noise uncertainty $\tilde{q}$, and it can be rewritten as follows:

$$\overset{\sim}{\boldsymbol{Q}}_{w_{k/k-1}} = \begin{bmatrix} \frac{\tilde{q}}{3}\tau_a^3 \boldsymbol{I}_{3\times3} & \frac{\tilde{q}}{2}\tau_a^2 \boldsymbol{I}_{3\times3} \\ \frac{\tilde{q}}{2}\tau_a^2 \boldsymbol{I}_{3\times3} & \tilde{q}\tau_a \boldsymbol{I}_{3\times3} \end{bmatrix} \tag{17}$$

By applying the adaptive estimation $\overset{\sim}{\boldsymbol{Q}}_{w_{k/k-1}}$ into Equations (4) and (5), the updated gain matrix is rewritten as follows:

$$\overset{\sim}{\boldsymbol{K}}_k = \overset{\sim}{\boldsymbol{P}}_k^- \boldsymbol{H}_k^T \left(\boldsymbol{H}_k \overset{\sim}{\boldsymbol{P}}_k^- \boldsymbol{H}_k^T + \boldsymbol{R}_k \right)^{-1} \tag{18}$$

where $\overset{\sim}{\boldsymbol{P}}_k^-$ is the covariance matrix of predicted state vector based on the adaptive estimation of process noise. This can be expressed as the following equation:

$$\overset{\sim}{\boldsymbol{P}}_k^- = \boldsymbol{\Phi}_{k/k-1} \boldsymbol{P}_{k-1} \boldsymbol{\Phi}_{k/k-1}^T + \overset{\sim}{\boldsymbol{Q}}_{w_{k/k-1}} \tag{19}$$

Combining the updated gain matrix $\overset{\sim}{\boldsymbol{K}}_k$ with Equation (6), the state vector and its covariance matrix based on the adaptive estimation of strong motion acceleration noise uncertainty can be estimated by the following equations:

$$\hat{\boldsymbol{X}}_k = \left(\boldsymbol{I}_{6\times6} - \overset{\sim}{\boldsymbol{K}}_k \boldsymbol{H}_k \right) \boldsymbol{X}_k^- + \overset{\sim}{\boldsymbol{K}}_k \boldsymbol{L}_k \tag{20}$$

$$\boldsymbol{P}_k = \left(\boldsymbol{I}_{6\times6} - \overset{\sim}{\boldsymbol{K}}_k \boldsymbol{H}_k \right) \boldsymbol{P}_k^- \tag{21}$$

Figure 1 illustrates the process flowchart of the Sage–Husa Kalman filter method with adaptive estimation of strong motion acceleration noise level. The process unfolds in three steps:

Step 1. Time Update

The time update provides a prediction for the current state and covariance estimation ahead of time. The state estimation is determined based on the system's dynamic model. The dynamic model incorporates the control input's strong motion acceleration. And the covariance matrix is updated to reflect the process noise associated with the acceleration noise level of the control input.

Step 2. Measurement Update

The measurement update refines the state and covariance estimations with new GNSS displacements. The gain matrix is calculated first, and this calculation determines the influences of new GNSS displacements on the state of estimation. Then the state estimation is updated by incorporating the GNSS displacements and the gain matrix. The covariance matrix is also updated to reflect the noise level of the GNSS displacement.

Step 3. Adaptive Estimation of Noise Uncertainty

A sliding-window estimation is employed to compute the noise uncertainty of the strong motion acceleration. Concurrently, the effect of the baseline shift is accommodated through adaptive variance inflation. With the adaptive estimation of acceleration noise uncertainty in place, the covariance of process noise is then calibrated.

After each time update, measurement update, and adaptive estimation of the process noise level cycle, the process is repeated with the updated state estimation and covariance serving as the basis for the next time update step.

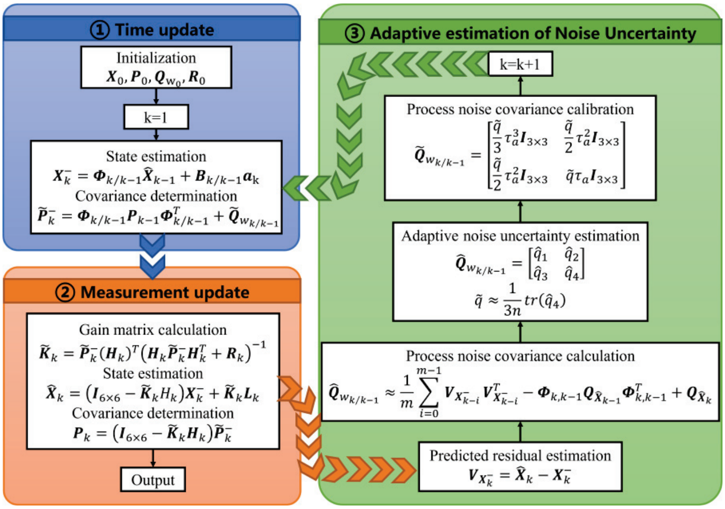

① Time update
Initialization
X_0, P_0, Q_{w_0}, R_0
k=1
State estimation
$X_k^- = \Phi_{k/k-1}\hat{X}_{k-1} + B_{k/k-1}a_k$
Covariance determination
$\tilde{P}_k^- = \Phi_{k/k-1}P_{k-1}\Phi_{k/k-1}^T + \tilde{Q}_{w_{k/k-1}}$
② Measurement update
Gain matrix calculation
$\tilde{K}_k = \tilde{P}_k^-(H_k)^T(H_k\tilde{P}_k^-H_k^T + R_k)^{-1}$
State estimation
$\hat{X}_k = (I_{6\times6} - \tilde{K}_kH_k)X_k^- + \tilde{K}_kL_k$
Covariance determination
$P_k = (I_{6\times6} - \tilde{K}_kH_k)\tilde{P}_k^-$
Output
③ Adaptive estimation of Noise Uncertainty
k=k+1
Process noise covariance calibration
$\tilde{Q}_{w_{k/k-1}} = \begin{bmatrix} \frac{\tilde{q}}{3}\tau_a^3 I_{3\times3} & \frac{\tilde{q}}{2}\tau_a^2 I_{3\times3} \\ \frac{\tilde{q}}{2}\tau_a^2 I_{3\times3} & \tilde{q}\tau_a I_{3\times3} \end{bmatrix}$
Adaptive noise uncertainty estimation
$\hat{Q}_{w_{k/k-1}} = \begin{bmatrix} \hat{q}_1 & \hat{q}_2 \\ \hat{q}_3 & \hat{q}_4 \end{bmatrix}$
$\tilde{q} \approx \frac{1}{3n} tr(\hat{q}_4)$
Process noise covariance calculation
$\hat{Q}_{w_{k/k-1}} \approx \frac{1}{m}\sum_{i=0}^{m-1} V_{X_{k-i}^-} V_{X_{k-i}^-}^T - \Phi_{k,k-1}Q_{\hat{X}_{k-1}}\Phi_{k,k-1}^T + Q_{\hat{X}_k}$
Predicted residual estimation
$V_{X_k^-} = \hat{X}_k - X_k^-$

Figure 1. Flowchart of the Sage–Husa Kalman filter method with adaptive noise uncertainty estimation.

3. Experiments, Results and Discussion

3.1. Data Processing Strategy

To evaluate the performance of the proposed Sage–Husa Kalman filter method with adaptive estimation of strong motion acceleration noise uncertainty, two experiments including a shake table simulation experiment and an application to Mw 7.8 and Mw 7.6 earthquake doublet were carried out. The data processing strategy of strong motion acceleration and GNSS displacement before integration are outlined in Table 1.

Table 1. Data processing strategies for GNSS displacement and strong motion acceleration before integration.

Items	Processing Information
Strong motion acceleration	De-mean the first 5 s from acceleration records
Covariance matrix of process noise	Determined by pre-event acceleration noises
GNSS displacement	Retrieved from TPP method
Covariance matrix of displacement noise	Determined by pre-event displacement noises

During the time update step, the acceleration recorded by a strong motion seismometer before an earthquake is not always zero. Such a phenomenon can be due to factors like initial adjustment errors or changes in environmental temperature. This non-zero acceleration value before the earthquake serves as the baseline initial value. If not corrected, it can lead to significant drifts in displacement waveforms after integration. Therefore, before integrating the GNSS displacement with the strong motion acceleration, the average acceleration value of 5 s before the earthquake was calculated as the initial baseline shift value. This value was deducted from the entire acceleration records to remove the initial baseline shift. These corrected accelerations served as the control input vectors. At the same time, the standard deviation of the pre-event acceleration records was calculated as the measuring noise uncertainty of the strong motion acceleration. The covariance matrix of the initial process noise was then determined by this measuring noise uncertainty. Concurrently, the coseismic displacements were recovered by double-integration of these corrected accelerations.

During the measurement update step, the GNSS-derived displacements were acquired using the TPP method. The underlying models, conventions, and strategies for the TPP approach have been detailed in our previous study [29]. The initial covariance matrix of GNSS displacement is also determined by pre-event noises. To highlight the benefits of integrating strong motion acceleration with GNSS displacement, the displacement derived from the GNSS TPP method and strong motion acceleration double-integration method are labeled as GNSS and Acc in the subsequent analysis.

For comparative purposes, the traditional Kalman filter method with empirically fixed noise was employed to validate the performance of the proposed method. The noise uncertainty of the traditional Kalman filter method was defined by referring to the multiplier selection strategy of Bock et al. [39]. The two processing schemes are presented in Table 2. For the sake of convenience, these two processing schemes are sequentially denoted as KF and KF + ANUE in the following. The software for the integration of high-rate GNSS and strong motion records was programmed using the C language, following the methods of KF and KF + ANUE.

Table 2. Two processing schemes.

Scheme	Method	Acceleration Noise Uncertainty
KF	Traditional Kalman filter method	Determined based on the pre-event noise with a multiplier
KF + ANUE	The proposed method	Adaptively estimated based on the Sage–Husa sliding-window estimation

3.2. Result and Discussion

3.2.1. Shake Table Simulation Experiment

To validate the performance of the proposed integration method, we utilized a shake table dataset open accessed by Wuhan University [52]. In this dataset, a Quanser Shake Table II was applied to simulate the seismic waveforms of a real earthquake. The shake table is a middle-size, single-axis earthquake simulator that can generate pre-loaded acceleration profiles of real earthquakes for seismogeodesy research. Equipped with a high-resolution encoder, the shake table can measure stage position at a sampling rate of up to 2000 Hz with a resolution of 3.10 µm. This simulation experiment provided a controlled environment for testing the proposed method.

During the shake table simulation experiment, the north–south accelerometer recordings from the Mianzhu-Qingping station were utilized. This station is located 1 km away from the epicenter of the 2008 May 12 Mw 7.9 Wenchuan earthquake. The data duration was about 150 s, with a recorded peak acceleration of about 2 g and a peak-to-peak displacement of about 12 cm. The records of simulated seismic displacement waveform (abbreviated as shake table) can be measured by the embedded encoder, serving as the truth benchmarks of ground movement. GNSS observations were collected at a sampling rate of

20 Hz using a Trimble NetR9 GNSS receiver with a Zephyr-2 geodetic antenna installed on the platform. A Leador PPOI-A15 navigation-grade triaxial inertial measurement unit (IMU) firmly anchored at the platform were used to gather acceleration records at 200 Hz. Initially, the IMU was aligned to the east, north, and up directions and was utilized as a strong motion accelerometer. Figure 2 illustrates the equipment setup for the shake table experiment.

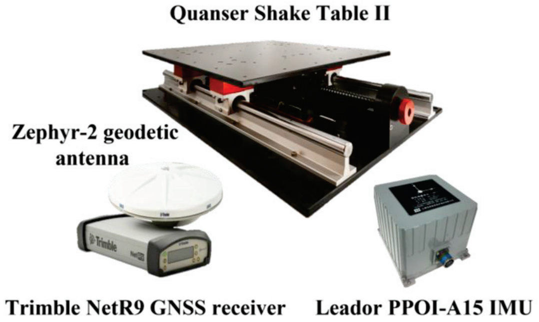


Figure 2. The equipment setup for the shake table experiment.

According to the aforementioned processing strategy, the 20 Hz GNSS observations and 200 Hz acceleration records were used to carry out displacement retrieval. Only the displacement in the north–south direction was involved in this assessment, in order to resemble the scene of a real earthquake occurrence more closely. Figure 3a presents the north–south oriented displacement waveforms after processing, which include Acc-derived, GNSS-derived, two integration solutions and shake table reference. The low-frequency noises are amplified in the Acc-derived displacement, and this leads to non-physical baseline drifts. This drift becomes larger and larger with the increase of integration time. Due to the mismatch of system model process noise, the filtered signal derived from the KF method exhibits unsmooth high-frequency oscillations resembling a sawtooth pattern. Compared to the traditional KF method, the KF + ANUE method exhibits better consistency with the shake table digital signal references. The effect of high-frequency sawtooth and baseline drifts is also suppressed.

To verify the characteristics of the proposed method in the frequency domain, the displacement power spectrum density (PSD) of the KF and KF + ANUE methods were calculated using the Welch algorithm [53], as shown in Figure 3b. In the low-frequency region below 1 Hz (Zone-1 and Zone-2), the PSD of KF and KF + ANUE is relatively consistent with the PSD of GNSS and shake table reference, indicating fewer errors in these frequency bands. In the frequency region above 1 Hz (Zone-3 and Zone-4), the displacement PSD of KF + ANUE is smaller than that obtained by the KF method, proving that the proposed method has a suppressive effect on high-frequency noise in coseismic displacement signals. The PSDs of the Acc displacement waveforms are larger than those of the GNSS, the KF + ANUE, and the shake table waveforms at the whole frequency domain, due to the effect of baseline shift.

The time series and histograms of the errors in computed displacement waveforms are further presented in Figure 4a,b, indicating the variability of their performance over the main shake period. The displacement retrieved from the KF + ANUE method exhibits a more stable and lower error range throughout the main shake period, which is indicative of its superior performance relative to the KF method. The error range for the KF method starts at approximately 0.01 cm and increases to around 2.48 cm over the course of 150 s. The KF + ANUE method maintains a lower error range, with values consistently below 1.23 cm, showcasing enhanced accuracy and stability. There is a visible trend in the KF method's error, which suggests a potential systematic increase in error over time, whereas the KF + ANUE method's error remains relatively flat, suggesting a robustness against time-varying effects of the baseline shift implicated in strong motion acceleration.

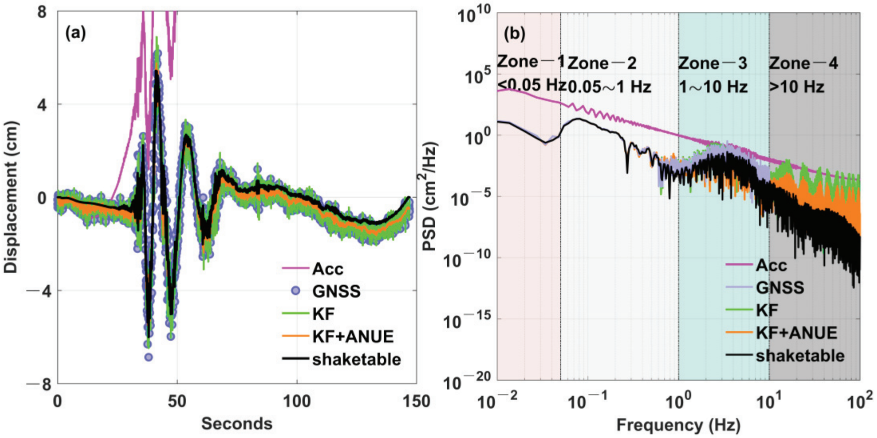


Figure 3. (**a**,**b**) Displacement time series and PSDs in the shake table experiment. (**a**) illustrates the time series of the integrated displacements. (**b**) showcases the PSDs of the integrated displacements.

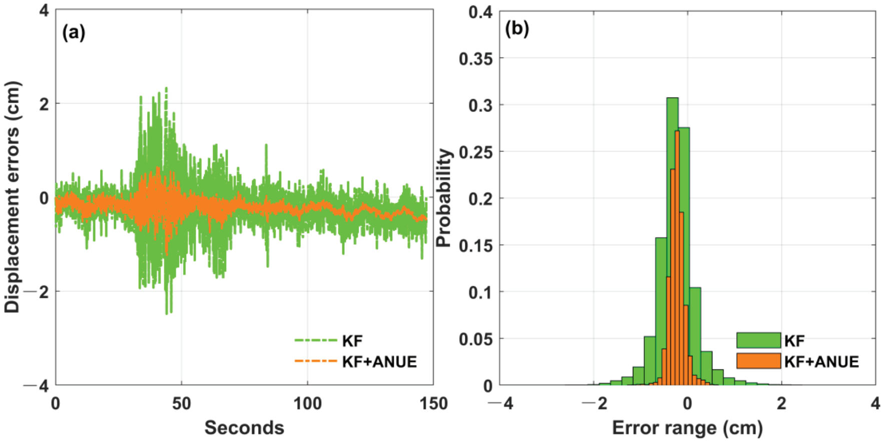


Figure 4. (**a**,**b**) Time series and histograms of the integrated displacement errors. (**a**) depicts the time series of the integrated displacement errors. (**b**) displays the distribution of the integrated displacement errors.

The root mean square error (RMSE) of the differences between the integrated displacement and the reference was calculated to verify the accuracy of the proposed method in this study. For a more rigorous analysis, we also computed the cross-correlation coefficient (CC) of each result related to the reference, as shown in Table 3. When the KF method is employed, the accuracy of the derived displacement is determined to be 0.45 cm, with the CC values reaching a level of 0.95. In contrast to the KF method, the displacement derived from the KF + ANUE method demonstrated superior alignment with the digital signal references obtained from the shake table. This enhanced consistency is evidenced by its exceptional CC of 0.99 and an RMSE of 0.32 cm for the displacement derived from the KF + ANUE method. The KF + ANUE method signifies a substantial improvement of up to 28% over the KF method in terms of displacement accuracy.

These preliminary validations indicate that the proposed method can provide broadband and more accurate displacement waveforms compared with the KF method. The historical predicted residuals in the recent period of time can be applied to provide a more accurate reference for the noise level of strong motion acceleration. This advancement is particularly significant in the context of seismic event analysis, where the fidelity of displacement retrieval is paramount for precise seismic waveform capture and seismic rupture propagation.

Table 3. The CC and RMSE for the integrated coseismic displacement.

Scheme	CC	RMSE (cm)
KF	0.95	0.45
KF + ANUE	0.99	0.32

3.2.2. Application to Mw 7.8 and Mw 7.6 Earthquake Doublet

To further validate the performance of the proposed KF + ANUE method for the application of real earthquake events, a representative example of a real historical earthquake is indispensable. On 6 February 2023, an earthquake doublet with magnitudes of Mw 7.8 and Mw 7.6 struck the East Anatolian Fault Zone, causing significant damage to civil and infrastructure throughout southeast Turkey and northwest Syria [54]. The entire main shaking process was successfully recorded by a dense network of high-rate GNSS with a sample rate of 1 Hz and strong motion stations operating at a sampling rate of 100 Hz. These data can be obtained from https://www.tusaga-aktif.gov.tr/, accessed on 25 March 2024.

We defined a collection of seismometers positioned within 5 km of each other as a pair. It is generally observed that seismic waveforms derived from GNSS and strong motion accelerometers remain consistent with each other for seismometer separations of less than 15 km [2]. Following this guideline, we identified 15 suitable co-locations in this contribution, with a separation distance ranging from 0.02 km to 3.37 km. The basic information about those co-located station pairs is provided in Table 4, and the distribution of co-located station pairs is shown in Figure 5. Following the processing strategy described in Section 3.1, the 1 Hz GNSS observations and 100 Hz acceleration records were processed to retrieve three-direction displacements for the Mw 7.8 and Mw 7.6 earthquake doublet in southeast Turkey and northwest Syria. The calculation period is set to more than 2 min, covering the main shake period. The accuracy of the displacements derived from the proposed method is assessed by taking the results obtained from PRIDE PPP-AR software ver. 3.0 developed by Wuhan University as references [55].

Table 4. Information on collected high-rate GNSS stations and strong motion seismometer stations.

Station	Latitude (°N)	Longitude (°E)	Dist to Epic [1] (km)	Separation Distance (km)
ANTE/2703	37.06	37.37	38.74	2.23
MAR1/4617	37.59	36.86	49.92	2.89
MAR1/4620	37.59	36.86	49.92	3.37
ONIY/8003	37.10	36.25	61.95	2.43
TUF1/0129	38.26	36.21	87.30	0.26
FEEK/0127	37.81	35.91	110.51	0.72
KLS1/7901	36.71	37.12	143.17	1.10
SIV1/6303	37.75	39.32	192.57	0.66
ADN2/0123	36.98	35.32	196.85	2.87
POZA/0124	37.40	34.87	210.46	1.63
AKLE/6306	36.71	38.95	214.09	1.97
ERGN/2104	38.27	39.76	230.07	0.58
VIR2/6302	37.22	39.75	244.56	0.98
MRSI/3301	36.78	34.60	262.70	0.02
DIY1/2101	37.95	40.19	266.92	2.32

[1] Dist to epic stands for distance from station location to epicenter location.

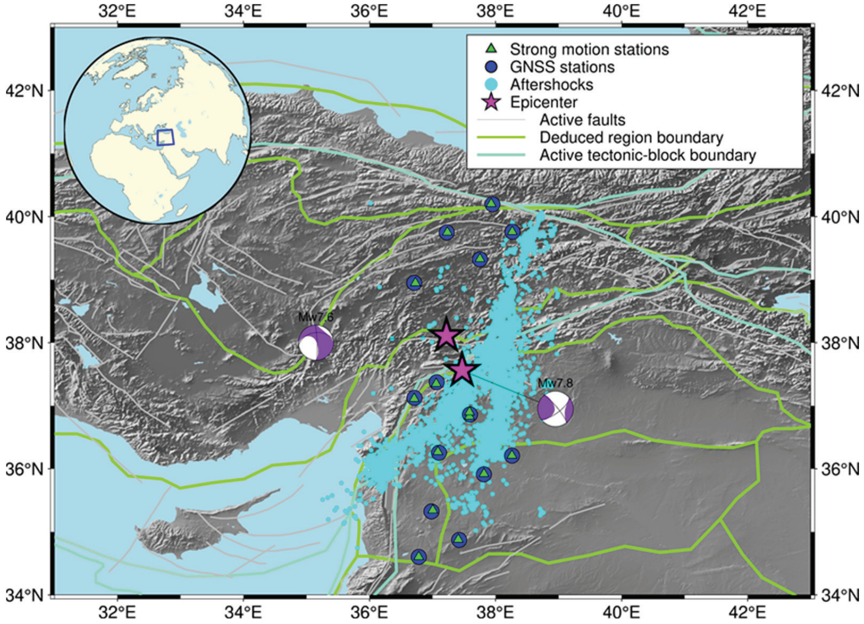


Figure 5. Location of the co-located high-rate GNSS and strong motion seismometer stations collected during the Mw 7.8 and Mw 7.6 earthquake doublet.

The left subgraphs in Figure 6 show the coseismic displacement of a typical station, SIV1/6303, in the east, north, and up directions during the first Mw 7.8 earthquake, respectively. This station is situated at a distance of 192.57 km from the epicenter of the same earthquake. The seismic signals recorded at this station are relatively complete. The "0" value on the horizontal axis signifies the moment the main shock occurred. Before the seismic waves arrive, the integrated displacements from the KF and KF + ANUE methods closely align with the GNSS-derived displacement. As the seismic waves propagate and the station experiences escalating shaking, the displacements derived from KF and KF + ANUE methods continue to track the general trend observed in the GNSS. In contrast, the displacements of Acc scheme exhibit non-physical drifts, which can be attributed to the impact of baseline shifts. The smoothness of the displacement waveforms across the KF and KF + ANUE methods is notably inconsistent. The high-frequency sawtooth evident in the displacement of KF and KF + VCE methods is less pronounced in the KF + ANUE scheme. Among these methods, KF + ANUE demonstrated the highest degree of congruence with the GNSS-derived displacement at equivalent time intervals, particularly as compared to the KF method. The baseline drift displayed in the Acc-derived displacement is also eliminated in the result of KF + ANUE.

To investigate the integration method in the frequency domain, the right subgraphs of Figure 6 show the PSDs of the GNSS-derived, Acc-derived, and integrated displacements of SIV1/6303 in the north, east, and up directions, respectively. The PSDs of the waveforms derived from the KF + ANUE demonstrate a greater alignment with those of the GNSS waveforms at low frequencies, below 1 Hz (Zone-1 and Zone-2), especially in the north and up directions. In the frequency band region above 1 Hz (Zone-3 and Zone-4), the displacement PSD of KF + ANUE is smaller than that obtained by the KF method. It means the high-frequency noises in coseismic displacement signals are effectively suppressed by the proposed method due to the adaptive variance inflation, which is similar to the results displayed in the shake table experiments.

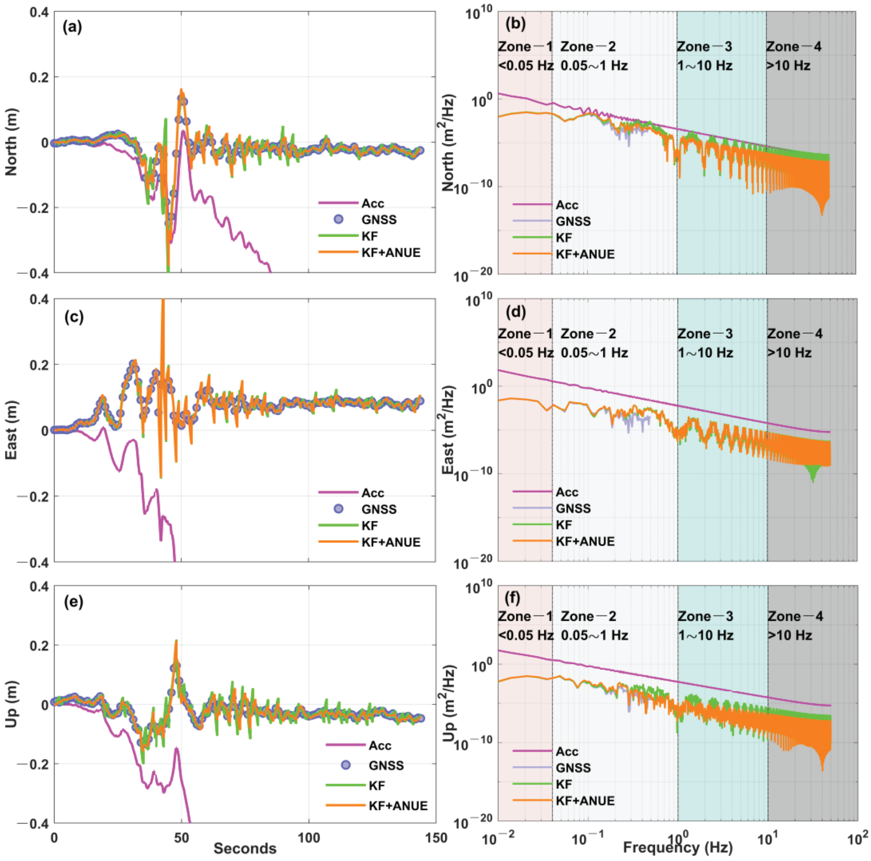


Figure 6. (**a**–**f**) Displacements and PSDs at the co-located station pair SIV1/6303 during the first Mw 7.8 earthquake. (**a**,**c**,**e**) illustrate the displacements of North, East and Up directions, respectively. (**b**,**d**,**f**) depict the displacement PSDs of North, East and Up directions, respectively.

Coseismic displacement waveforms of KF exhibit their disadvantages as in our previous analyses. The displacement waveforms derived from the KF + ANUE method can avoid the drawbacks and preserve the benefits of GNSS and strong motion records, providing broadband displacements with a high-precision level. To depict seismic rupture propagation, Figures 7 and 8 present the displacement waveforms derived from the KF + ANUE method for 11 available station pairs during the Mw 7.8 and Mw 7.6 earthquake doublet, respectively. Displacements derived solely from GNSS and strong motion seismometers are also presented for comparison. It should be noticed that the coseismic displacements of each station are vertically shifted according to the epicentral distance. Displacement waveforms from GNSS and KF + ANUE remained generally consistent until data transmissions from the MAR1 and AKLE stations were disrupted. However, due to the absence of subsequent GNSS displacements, the effectiveness of integrated displacement throughout the earthquake remains undetermined. At station ONIY/8003, located 61.95 km from the hypocenter, the integrated displacement waveform exhibits two ambiguous phases of seismic energy release, attributable to permanent ground displacement. The remaining eight station pairs recorded coseismic displacement waveforms throughout the earthquake, with no significant systematic deviation observed between GNSS-derived displacements and those obtained via the KF + ANUE method.

Similar to the results of the first Mw 7.8 earthquake, Figure 8 displays the KF + ANUE integrated displacement sequences of 11 available station pairs in the second Mw 7.6 earthquake. Due to damage caused by the strong earthquake, four station pairs are different from those in the first earthquake case. Prior to the seismic wave's arrival, the integrated displacement aligns closely with that recorded by strong motion seismometers. Following the wave's arrival, the integrated displacement mirrors the result of GNSS, albeit with shake amplitudes less than those of identical station pairs in the first event, remaining under 40 cm. In general terms, GNSS displacements suffer from low sampling rates and high noise levels, failing to capture seismic wave signals in detail. Strong motion seismometers, despite their high sampling rates, exhibit significant drift post-integration. The proposed method has greater superiority over these two single observing techniques.

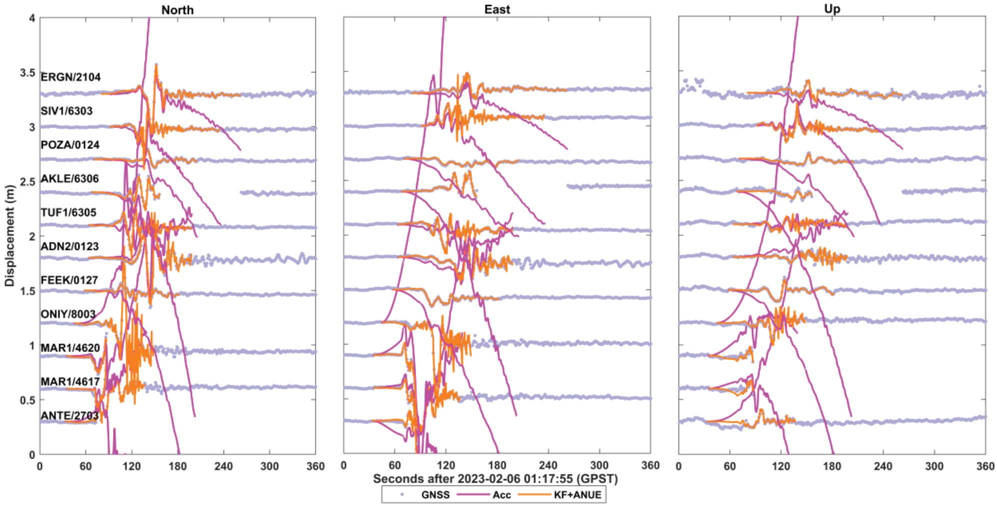


Figure 7. Coseismic displacements at 11 station pairs during the first Mw 7.8 earthquake.

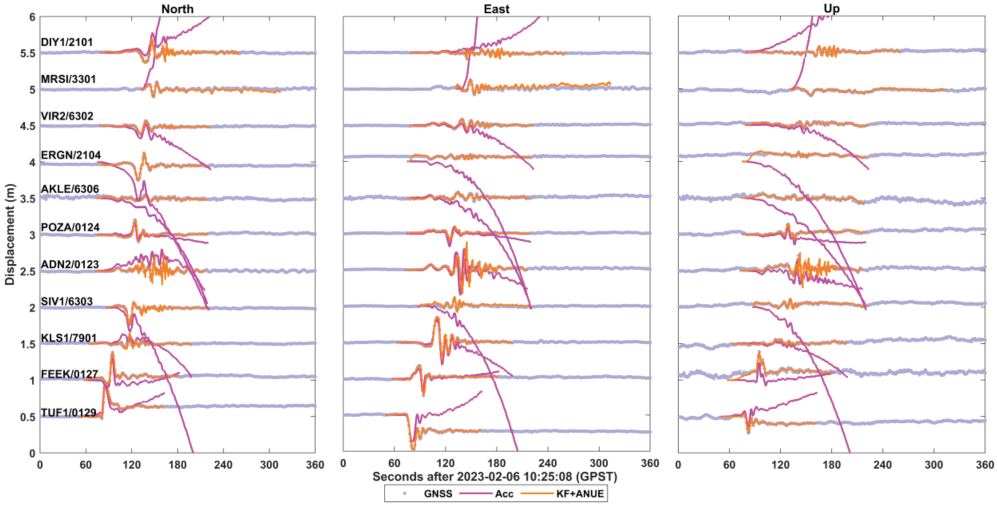


Figure 8. Coseismic displacements at 11 station pairs during the second Mw 7.6 earthquake.

Table 5 provides a quantitative analysis of the KF + ANUE method's performance in capturing seismic wave displacements. The mean CC and RMSE for co-located GNSS and strong motion station pairs during the Mw 7.8 and Mw 7.6 earthquake doublet are presented. Compared to the KF scheme, displacements derived from the KF + ANUE method demonstrated superior performance in terms of both accuracy and signal correlation for the Mw 7.8 and Mw 7.6 earthquake doublet. For the Mw 7.8 event, the KF method achieves accuracies of 2.24 cm, 1.92 cm, and 2.11 cm in their respective directions, with corresponding CC values of 0.85, 0.87, and 0.86. The CC values for the KF + ANUE method both exceeded 0.90 in all three directions. Conversely, the KF + ANUE method achieves displacement extraction accuracies of 1.31 cm, 1.02 cm, and 1.64 cm in the north, east, and up directions, respectively. This represents an improvement of up to 44% in the horizontal direction and 22% in the vertical direction over the KF method.

Mirroring its accuracy in the preceding event, the proposed method can retrieve displacement for the Mw 7.6 event with remarkable precision. It achieved accuracies of 1.33 cm, 1.43 cm, and 1.35 cm in the north, east, and up directions, respectively. The CC values for the KF + ANUE method average at 0.94 in the horizontal and vertical directions. In comparison, the KF method's performance in these directions is marked by accuracies of 1.44 cm, 1.72 cm, and 1.47 cm, respectively. The average CC value for the three directions is 0.92. While there are only minor differences in the accuracies of displacements retrieved from the KF and KF + ANUE methods in the north and up directions, a significant improvement was observed in the displacements in the east direction during the second Mw 7.6 earthquake. Despite these nuances, the proposed method exhibits an overall improvement of up to 12% in the horizontal direction and 8% in the vertical direction as compared to the KF method.

By synthesizing the accuracy of two distinct events, the proposed method significantly enhances the capture of seismic waveforms, showing improvements of 46% in the horizontal direction and 23% in the vertical direction over the outcomes of the KF method. The above results show that the proposed method can provide much more precise coseismic displacement than the KF method. These integrated displacements of the collected GNSS and strong motion stations are consistent with the reference displacement values processed by PRIDE PPP-AR software.

Table 5. The mean CC and RMSE (cm) for the retrieved coseismic displacements in the Mw 7.8 and Mw 7.6 earthquake doublet.

Scheme	The First Mw 7.8 Event						The Second Mw 7.6 Event					
	CC			RMSE (cm)			CC			RMSE (cm)		
	North	East	Up	North	East	Up	North	East	Up	North	East	Up
KF	0.85	0.87	0.86	2.24	1.92	2.11	0.93	0.90	0.93	1.44	1.72	1.47
KF + ANUE	0.95	0.98	0.91	1.31	1.02	1.64	0.95	0.93	0.94	1.33	1.43	1.35

In a word, the KF + ANUE method markedly outperforms the KF method by adaptively estimating the noise uncertainty of strong motion acceleration utilizing the Sage–Husa sliding-window estimation principle. In the pre-filtering and mid-filtering stages, the influences of baseline shift were accommodated through adaptive variance inflation, thereby eliminating the drift in seismic signals effectively. In the later filtering stages, GNSS-derived displacements significantly constrained the integration results, enabling accurate recording of seismic waves' low-frequency information. Despite being approximately 200 km from the epicenter, stations like POZA/0124 and five others successfully detected seismic signals. The proposed method combines the strengths of both GNSS and strong motion seismometers, ensuring comprehensive seismic signal capture without divergence throughout the earthquake event.

4. Conclusions

In this paper, we present a Sage–Husa Kalman filter method, where the noise uncertainty of strong motion acceleration is adaptively estimated to integrate GNSS and strong motion for obtaining displacement series. The broadband displacement waveforms in a great earthquake can be achieved by using the proposed method. Compared with the traditional Kalman filter method, historical predicted residuals in the recent times are applied to provide a more accurate reference for the noise level of strong motion acceleration. The noise uncertainty of strong motion acceleration is adaptively determined utilizing the Sage–Husa sliding-window estimation principle, at the same time the effect of the baseline shift is accommodated through adaptive variance inflation. This method significantly improved the accuracy of the system's process noise representation, and thereby improved filter performance. The performance of the proposed method was assessed through a shake table simulation experiment and by analyzing data from GNSS/strong motion co-located stations during the 2023 Mw 7.8 and Mw 7.6 earthquake doublet in Southeast Turkey. The results indicate that the method significantly improves adaptability to variations in strong motion accelerometer noises and enhances accuracy of the integrated displacement series. Specifically, in the shake table test, the proposed method retrieved the displacements with an accuracy of 0.32 cm. This represents an improvement of up to 28% over the traditional Kalman filter method. The displacement waveforms derived from the proposed method achieved a correlation coefficient of 0.99 with respect to reference values, outperforming the traditional Kalman filter method by a margin of 0.04. These increases in the retrieval accuracy and correlation coefficient signify a notably more accurate alignment with the benchmark signal. When applied to the earthquake event, the proposed method demonstrated an ability to capture seismic waveforms with an improvement of 46% and 23% in the horizontal and vertical directions, respectively, as compared with the traditional Kalman filter-based results. This contribution shows that the proposed method can provide a more accurate estimation of broadband displacement waveforms to support earthquake early warning and rapid response in a great earthquake.

Author Contributions: Conceptualization, Z.N. and Y.Z.; methodology, Y.Z.; software, Y.Z.; validation, Z.N., Y.Z. and Z.W.; formal analysis, Y.Z.; investigation, Y.Z.; resources, G.Z. and X.S.; data curation, Y.Z.; writing—original draft preparation, Y.Z.; writing—review and editing, Z.N.; visualization, Y.Z.; supervision, Z.W., G.Z. and X.S.; project administration, Z.N. and Z.W.; funding acquisition, Z.N. All authors have read and agreed to the published version of the manuscript.

Funding: This study was funded by the National Natural Science Foundation of China (Grant No. 42104011), the Shandong Provincial Natural Science Foundation (Grant No. ZR2021QD069), and Fundamental Research Funds for the Central Universities (Grant No. 22CX06032A).

Data Availability Statement: The datasets analyzed in this study are managed by the College of Oceanography and Space Informatics, China University of Petroleum, Qingdao, China, and can be available upon request from the corresponding author.

Acknowledgments: We thank Wuhan University for providing the seismic shake table data and PPP-AR software. The authors also thank Guohong Zhang and Xinjian Shan at the Institute of Geology, China Earthquake Administration, for sharing geodetic observations of the 2023 Turkey earthquake doublet.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Saunders, J.K.; Goldberg, D.E.; Haase, J.S.; Bock, Y.; Offield, D.G.; Melgar, D.; Restrepo, J.; Fleischman, R.B.; Nema, A.; Geng, J.; et al. Seismogeodesy Using GPS and Low-Cost MEMS Accelerometers: Perspectives for Earthquake Early Warning and Rapid Response. *Bull. Seismol. Soc. Am.* **2016**, *106*, 2469–2489. [CrossRef]
2. Emore, G.L.; Haase, J.S.; Choi, K.; Larson, K.M.; Yamagiwa, A. Recovering Seismic Displacements through Combined Use of 1-Hz GPS and Strong-Motion Accelerometers. *Bull. Seismol. Soc. Am.* **2007**, *97*, 357–378. [CrossRef]
3. Dahmen, N.; Hohensinn, R.; Clinton, J. Comparison and Combination of GNSS and Strong-Motion Observations: A Case Study of the 2016 Mw 7.0 Kumamoto Earthquake. *Bull. Seismol. Soc. Am.* **2020**, *110*, 2647–2660. [CrossRef]

4. Iwan, W.D.; Moser, M.A.; Peng, C.Y. Some Observations on Strong-Motion Earthquake Measurement Using a Digital Accelerograph. *Bull. Seismol. Soc. Am.* **1985**, *75*, 1225–1246. [CrossRef]
5. Boore, D.M. Comments on Baseline Correction of Digital Strong-Motion Data: Examples from the 1999 Hector Mine, California, Earthquake. *Bull. Seismol. Soc. Am.* **2002**, *92*, 1543–1560. [CrossRef]
6. Boore, D.M.; Bommer, J.J. Processing of strong-motion accelerograms: Needs, options and consequences. *Soil Dyn. Earthq. Eng.* **2005**, *25*, 93–115. [CrossRef]
7. Wu, Y.M.; Wu, C.F. Approximate Recovery of Coseismic Deformation from Taiwan Strong-Motion Records. *J. Seismol.* **2007**, *11*, 159–170. [CrossRef]
8. Chao, W.A.; Wu, Y.M.; Zhao, L. An Automatic Scheme for Baseline Correction of Strong-Motion Records in Coseismic Deformation Determination. *J. Seismol.* **2010**, *14*, 495–504. [CrossRef]
9. Wang, R.; Schurr, B.; Milkereit, C.; Shao, Z.; Jin, M.P. An Improved Automatic Scheme for Empirical Baseline Correction of Digital Strong-Motion Records. *Bull. Seismol. Soc. Am.* **2011**, *101*, 2029–2044. [CrossRef]
10. Larson, K.M.; Bodin, P.; Gomberg, J. Using 1-Hz GPS Data to Measure Deformations Caused by the Denali Fault Earthquake. *Science* **2003**, *300*, 1421–1424. [CrossRef]
11. Li, X.; Ge, M.; Zhang, X.; Zhang, Y.; Guo, B.; Wang, R.; Klotz, J.; Wickert, J. Real-time high-rate co-seismic displacement from ambiguity-fixed precise point positioning: Application to earthquake early warning. *Geophys. Res. Lett.* **2013**, *40*, 295–300. [CrossRef]
12. Fang, R.; Shi, C.; Song, W.; Wang, G.; Liu, J. Determination of earthquake magnitude using GPS displacement waveforms from real-time precise point positioning. *Geophys. J. Int.* **2014**, *196*, 461–472. [CrossRef]
13. Bock, Y.; Nikolaidis, R.M.; de Jonge, P.J.; Bevis, M. Instantaneous geodetic positioning at medium distances with the Global Positioning System. *J. Geophys. Res.* **2000**, *105*, 28223–28253. [CrossRef]
14. Bock, Y.; Prawirodirdjo, L.; Melbourne, T.I. Detection of arbitrarily large dynamic ground motions with a dense high-rate GPS network. *Geophys. Res. Lett.* **2004**, *31*, L06604. [CrossRef]
15. Ohta, Y.; Kobayashi, T.; Tsushima, H.; Miura, S.; Hino, R.; Takasu, T.; Fujimoto, H.; Iinuma, T.; Tachibana, K.; Demachi, T.; et al. Quasi real-time fault model estimation for near-field tsunami forecasting based on RTK-GPS analysis: Application to the 2011 Tohoku-Oki earthquake (Mw 9.0). *J. Geophys. Res. Solid Earth* **2012**, *117*, B02311. [CrossRef]
16. Kouba, J. Measuring Seismic Waves Induced by Large Earthquakes with GPS. *Stud. Geophys. Geod.* **2003**, *47*, 741–755. [CrossRef]
17. Shi, C.; Lou, Y.; Zhang, H.; Zhao, Q.; Geng, J.; Wang, R.; Fang, R.; Liu, J. Seismic deformation of the Mw 8.0 Wenchuan earthquake from high-rate GPS observations. *Adv. Space Res.* **2010**, *46*, 228–235. [CrossRef]
18. Li, X.; Ge, M.; Lu, C.; Zhang, Y.; Wang, R.; Wickert, J.; Schuh, H. High-Rate GPS Seismology Using Real-Time Precise Point Positioning with Ambiguity Resolution. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 6165–6180. [CrossRef]
19. Chen, G.; Zhao, Q. Near-field surface displacement and permanent deformation induced by the Alaska Mw 7.5 earthquake determined by high-rate real-time ambiguity-fixed PPP solutions. *Chin. Sci. Bull.* **2014**, *59*, 4781–4789. [CrossRef]
20. Li, X.; Li, X.; Yuan, Y.; Zhang, K.; Zhang, X.; Wickert, J. Multi-GNSS phase delay estimation and PPP ambiguity resolution: GPS, BDS, GLONASS, Galileo. *J. Geod.* **2018**, *92*, 579–608. [CrossRef]
21. Collins, P.; Henton, J.; Mireault, Y.; Heroux, P.; Schmidt, M.; Dragert, H.; Bisnath, S. Precise point positioning for real-time determination of co-seismic crustal motion. In Proceedings of the ION GNSS 2009, Savannah, GA, USA, 22–25 September 2009; pp. 2479–2488.
22. Colosimo, G.; Crespi, M.; Mazzoni, A. Real-time GPS seismology with a stand-alone receiver: A preliminary feasibility demonstration. *J. Geophys. Res. Solid Earth* **2011**, *116*, B11302. [CrossRef]
23. Geng, T.; Xie, X.; Fang, R.; Su, X.; Zhao, Q.; Liu, G.; Li, H.; Shi, C.; Liu, J. Real-time capture of seismic waves using high-rate multi-GNSS observations: Application to the 2015 Mw 7.8 Nepal earthquake. *Geophys. Res. Lett.* **2016**, *43*, 161–167. [CrossRef]
24. Shu, Y.; Fang, R.; Li, M.; Shi, C.; Li, M.; Liu, J. Very high-rate GPS for measuring dynamic seismic displacements without aliasing: Performance evaluation of the variometric approach. *GPS Solut.* **2018**, *22*, 121. [CrossRef]
25. Li, X.; Ge, M.; Guo, B.; Wickert, J.; Schuh, H. Temporal point positioning approach for real-time GNSS seismology using a single receiver. *Geophys. Res. Lett.* **2013**, *40*, 5677–5682. [CrossRef]
26. Li, X.; Guo, B.; Lu, C.; Ge, M.; Wickert, J.; Schuh, H. Real-time GNSS seismology using a single receiver. *Geophys. J. Int.* **2014**, *198*, 72–89. [CrossRef]
27. Guo, B.; Zhang, X.; Ren, X.; Li, X. High-precision coseismic displacement estimation with a single-frequency GPS receiver. *Geophys. J. Int.* **2015**, *202*, 612–623. [CrossRef]
28. Chen, K.; Ge, M.; Babeyko, A.; Li, X.; Diao, F.; Tu, R. Retrieving real-time co-seismic displacements using GPS/GLONASS: A preliminary report from the September 2015 Mw 8.3 Illapel earthquake in Chile. *Geophys. J. Int.* **2016**, *206*, 941–953. [CrossRef]
29. Zhang, Y.; Nie, Z.; Wang, Z.; Wu, H.; Xu, X. Real-Time Coseismic Displacement Retrieval Based on Temporal Point Positioning with IGS RTS Correction Products. *Sensors* **2021**, *21*, 334. [CrossRef] [PubMed]
30. Genrich, J.F.; Bock, Y. Instantaneous geodetic positioning with 10–50 Hz GPS measurements: Noise characteristics and implications for monitoring networks. *J. Geophys. Res. Solid Earth* **2006**, *1111*, B03403. [CrossRef]
31. Larson, K.M.; Bilich, A.; Axelrad, P. Improving the precision of high-rate GPS. *J. Geophys. Res.* **2007**, *112*, B05422. [CrossRef]
32. Gao, Z.; Li, Y.; Shan, X.; Zhu, C. Earthquake Magnitude Estimation from High-Rate GNSS Data: A Case Study of the 2021 Mw 7.3 Maduo Earthquake. *Remote Sens.* **2021**, *13*, 4478. [CrossRef]

33. Li, Z.; Zang, J.; Fan, S.; Wen, Y.; Xu, C.; Yang, F.; Peng, X.; Zhao, L.; Zhou, X. Real-Time Source Modeling of the 2022 Mw 6.6 Menyuan, China Earthquake with High-Rate GNSS Observations. *Remote Sens.* **2022**, *14*, 5378. [CrossRef]
34. Boore, D.M. Effect of Baseline Corrections on Displacements and Response Spectra for Several Recording of the 1999 Chi-Chi, Taiwan, Earthquake. *Bull. Seismol. Soc. Am.* **2001**, *91*, 1199–1211. [CrossRef]
35. Wang, R.; Parolai, S.; Ge, M.; Jin, M.P.; Walter, T.R.; Zschau, J. The 2011 Mw 9.0 Tohoku earthquake: Comparison of GPS and strong-motion data. *Bull. Seismol. Soc. Am.* **2013**, *103*, 1336–1347. [CrossRef]
36. Tu, R.; Wang, R.; Ge, M.; Walter, T.R.; Ramatschi, M.; Milkereit, C.; Bindi, D.; Dahm, T. Cost-effective monitoring of ground motion related to earthquakes, landslides, or volcanic activity by joint use of a single-frequency GPS and a MEMS accelerometer. *Geophys. Res. Lett.* **2013**, *40*, 3825–3829. [CrossRef]
37. Melgar, D.; Bock, Y.; Sanchez, D.; Crowell, B.W. On robust and reliable automated baseline corrections for strong motion seismology. *J. Geophys. Res. Solid Earth* **2013**, *118*, 1177–1187. [CrossRef]
38. Smyth, A.; Wu, M. Multi-rate Kalman filtering for the data fusion of displacement and acceleration response measurements in dynamic system monitoring. *Mech. Syst. Signal Process.* **2007**, *21*, 706–723. [CrossRef]
39. Bock, Y.; Melgar, D.; Crowell, B.W. Real-Time Strong-Motion Broadband Displacements from Collocated GPS and Accelerometers. *Bull. Seismol. Soc. Am.* **2011**, *101*, 2904–2925. [CrossRef]
40. Song, C.; Xu, C. Loose integration of high-rate GPS and strong motion data considering coloured noise. *Geophys. J. Int.* **2018**, *215*, 1530–1539. [CrossRef]
41. Shu, Y.; Fang, R.; Geng, J.; Zhao, Q.; Liu, J. Broadband Velocities and Displacements From Integrated GPS and Accelerometer Data for High-Rate Seismogeodesy. *Geophys. Res. Lett.* **2018**, *45*, 8939–8948. [CrossRef]
42. Geng, J.; Bock, Y.; Melgar, D.; Crowell, B.W.; Haase, J.S. A new seismogeodetic approach applied to GPS and accelerometer observations of the 2012 Brawley seismic swarm: Implications for earthquake early warning. *Geochem. Geophys. Geosyst.* **2013**, *14*, 2124–2142. [CrossRef]
43. Li, X.; Ge, M.; Zhang, Y.; Wang, R.; Klotz, J.; Wicket, J. High-rate coseismic displacements from tightly-integrated processing of raw GPS and accelerometer data. *Geophys. J. Int.* **2013**, *195*, 612–624. [CrossRef]
44. Tu, R.; Ge, M.; Wang, R.; Walter, T.R. A new algorithm for tight integration of real-time GPS and strong-motion records, demonstrated on simulated, experimental, and real seismic data. *J. Seismol.* **2014**, *18*, 151–161. [CrossRef]
45. Guo, B.; Di, M.; Song, F.; Li, J.; Shi, S.; Limsupavanich, N. Integrated coseismic displacement derived from high-rate GPS and strong-motion seismograph: Application to the 2017 Ms 7.0 Jiuzhaigou Earthquake. *Measurement* **2021**, *182*, 109735. [CrossRef]
46. Fang, R.; Zheng, J.; Shu, Y.; Lv, H.; Shi, C.; Liu, J. A new tightly coupled method for high-rate seismogeodesy: A shake table experiment and application to the 2016 Mw 6.6 central Italy earthquake. *Geophys. J. Int.* **2021**, *227*, 1846–1856. [CrossRef]
47. Xin, S.; Geng, J.; Zeng, R.; Zhang, Q.; Ortega-Culaciati, F.; Wang, T. In-situ real-time seismogeodesy by integrating multi-GNSS and accelerometers. *Measurement* **2021**, *179*, 109453. [CrossRef]
48. Chen, C.; Lin, X.; Li, W.; Cheng, L.; Wang, H.; Zhang, Q.; Wang, Z. Adaptive coloured noise multirate Kalman filter and its application in coseismic deformations. *Geophys. J. Int.* **2023**, *234*, 1236–1253. [CrossRef]
49. Jing, C.; Huang, G.; Zhang, Q.; Li, X.; Bai, Z.; Du, Y. GNSS/Accelerometer Adaptive Coupled Landslide Deformation Monitoring Technology. *Remote Sens.* **2023**, *14*, 3537. [CrossRef]
50. Paziewski, J.; Stepniak, K.; Sieradzki, R.; Yigit, C.O. Dynamic displacement monitoring by integrating high-rate GNSS and accelerometer: On the possibility of downsampling GNSS data at reference stations. *GPS Solut.* **2023**, *27*, 157. [CrossRef]
51. Jing, C.; Huang, G.; Li, X.; Zhang, Q.; Yang, H.; Zhang, K.; Liu, G. GNSS/accelerometer integrated deformation monitoring algorithm based on sensors adaptive noise modeling. *Measurement* **2023**, *179*, 113179. [CrossRef]
52. Geng, J.; Wen, Q.; Zhang, T.; Li, C. Strong-motion seismogeodesy by deeply coupling GNSS receivers with inertial measurement units. *Geophys. Res. Lett.* **2020**, *47*, e2020GL087161. [CrossRef]
53. Qu, X.; Ding, X.; Xu, Y.; Yu, W. Real-time outlier detection in integrated GNSS and accelerometer structural health monitoring systems based on a robust multi-rate Kalman filter. *J. Geod.* **2023**, *97*, 38. [CrossRef]
54. Liu, C.; Lay, T.; Wang, R.; Taymaz, T.; Xie, Z.; Xiong, X.; Irmak, T.S.; Kahraman, M.; Erman, C. Complex multi-fault rupture and triggering during the 2023 earthquake doublet in southeastern Türkiye. *Nat. Commun.* **2023**, *14*, 5564. [CrossRef] [PubMed]
55. Geng, J.; Chen, X.; Pan, Y.; Mao, S.; Li, C.; Zhou, J.; Zhang, K. PRIDE PPP-AR: An open-source software for GPS PPP ambiguity resolution. *GPS Solut.* **2019**, *23*, 91. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

A Low-Cost 3D SLAM System Integration of Autonomous Exploration Based on Fast-ICP Enhanced LiDAR-Inertial Odometry

Conglin Pang [1], Liqing Zhou [1] and Xianfeng Huang [2,3,*]

[1] Electronic Information School, Wuhan University, Wuhan 430072, China; pang_conglin@whu.edu.cn (C.P.); zlq@whu.edu.cn (L.Z.)

[2] The State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, China

[3] Intellectual Computing Laboratory for Cultural Heritage, Wuhan University, Wuhan 430072, China

* Correspondence: huangxf@whu.edu.cn

Abstract: Advancements in robotics and mapping technology have spotlighted the development of Simultaneous Localization and Mapping (SLAM) systems as a key research area. However, the high cost of advanced SLAM systems poses a significant barrier to research and development in the field, while many low-cost SLAM systems, operating under resource constraints, fail to achieve high-precision real-time mapping and localization, rendering them unsuitable for practical applications. This paper introduces a cost-effective SLAM system design that maintains high performance while significantly reducing costs. Our approach utilizes economical components and efficient algorithms, addressing the high-cost barrier in the field. First, we developed a robust robotic platform based on a traditional four-wheeled vehicle structure, enhancing flexibility and load capacity. Then, we adapted the SLAM algorithm using the LiDAR-inertial Odometry framework coupled with the Fast Iterative Closest Point (ICP) algorithm to balance accuracy and real-time performance. Finally, we integrated the 3D multi-goal Rapidly exploring Random Tree (RRT) algorithm with Nonlinear Model Predictive Control (NMPC) for autonomous exploration in complex environments. Comprehensive experimental results confirm the system's capability for real-time, autonomous navigation and mapping in intricate indoor settings, rivaling more expensive SLAM systems in accuracy and efficiency at a lower cost. Our research results are published as open access, facilitating greater accessibility and collaboration.

Keywords: cost-effective; SLAM systems; autonomous exploration; LiDAR-inertial odometry

Citation: Pang, C.; Zhou, L.; Huang, X. A Low-Cost 3D SLAM System Integration of Autonomous Exploration Based on Fast-ICP Enhanced LiDAR-Inertial Odometry. *Remote Sens.* **2024**, *16*, 1979. https://doi.org/10.3390/rs16111979

Academic Editors: Zhetao Zhang, Guorui Xiao, Zhixi Nie, Vagner Ferreira and Giuseppe Casula

Received: 25 March 2024
Revised: 24 May 2024
Accepted: 24 May 2024
Published: 30 May 2024

1. Introduction

In the interdisciplinary realm of robotics, computer vision and artificial intelligence, Simultaneous Localization and Mapping (SLAM) [1] stands out as a hot topic technology that enables autonomous systems to navigate and understand their surroundings. SLAM's principle entails mapping an uncharted environment and concurrently monitoring the agent's position within this space [2]. This technology is important for many applications, including self-driving cars [3–5] and augmented reality [6–8], making SLAM a major focus in the development of smart, self-guiding machines.

The development of SLAM technology is rapid, driven by continuous advancements in computational power, algorithmic efficiency, and sensor technology. Over the years, various SLAM solutions have been developed, each employing distinct methodologies and technological advancements to address the challenges of mapping and localization. Early approaches [9–11] relied heavily on expensive and high-precision sensors, like lasers, to achieve detailed environmental mapping and accurate localization. These systems, while effective, were often limited by their high costs and substantial computational requirements, which deviated from the original intention of SLAM technology. Additionally, early SLAM systems exhibited limited adaptability to dynamic environments and were susceptible to interference, necessitating high-quality and high-performance LiDAR sensors. This indirectly escalated the operational costs of SLAM systems and restricted their widespread adoption in various application scenarios.

One of the recent advancements in SLAM has seen the integration of visual SLAM (VSLAM) techniques [12–17], which relies on camera images to perform localization and mapping tasks. While VSLAM systems reduced the reliance on costly sensors, they introduced new challenges, particularly in terms of real-time processing capabilities. Processing high-resolution video feeds in real-time requires significant computational power, which can still lead to high system costs [14]. Furthermore, visual SLAM systems can struggle in low-light conditions or environments with repetitive textures, leading to decreased reliability and accuracy, which adversely affects the stability of low-cost SLAM systems.

Recently, solid-state LiDARs have gained prominence as a primary trend in LiDAR development due to advancements in technology, especially in micro-electro-mechanical systems (MEMS) [18] and rotating prisms [19]. These LiDARs offer improved reliability and cost-effectiveness, as they eliminate the need for rotating mechanical parts, resulting in a more compact design. Their lightweight and high-performance characteristics enable solid-state LiDARs to deliver accurate 3D measurements over long ranges, making them ideal for autonomous vehicles, industrial robots, and drones that require precise 3D mapping in complex or challenging environments. Solid-state LiDARs are becoming critical for enabling intelligent systems and integrating seamlessly into lightweight, low-cost SLAM systems. However, due to the high resolution of solid-state LiDAR, its scans typically contain a significant number of feature points (ranging from thousands to tens of thousands). Processing the data from the IMU combined with the numerous feature points can overwhelm low-cost edge computing systems. Furthermore, the scan frequency often exceeds that of the IMU, resulting in motion distortion, which makes registration challenging for LiDAR-Inertial Odometry (LIO) based LiDAR SLAM algorithms [20].

Another critical issue facing contemporary SLAM systems is how to incorporate autonomous exploration modules. Autonomous exploration refers to a system's ability to independently navigate and collect information from its environment under the guidance of its own decision-making processes, widely applied in robotic exploration and mapping in unknown complex environments [21]. Early exploration algorithms, such as Frontier-Based Exploration [22] and Greedy Search [23], prioritized discovering unexplored regions to improve map coverage but tended to get trapped in local optima in complex terrains. Additionally, they faced real-time decision-making challenges due to computational limitations and an inability to effectively handle dynamic environmental changes.

Recent advances have combined exploration with SLAM, improving how autonomous systems understand and adapt to their surroundings [21,24,25]. Despite these advancements, a significant limitation remains the traditional focus on 2D exploration strategies, primarily designed for ground-based robots. This emphasis limits SLAM systems' ability to autonomously reconstruct 3D environments in real-world settings [26], which are intrinsically three-dimensional and complex.

Addressing these challenges, this paper proposes a novel, cost-effective SLAM system that balances cost, accuracy, and real-time performance. The primary contributions of our work are threefold:

1. We developed a low-cost mobile robot SLAM system, significantly reducing manufacturing costs and enhancing system performance and 3D exploration capabilities through careful design of the robot's structure and selection of high-performing yet affordably priced sensors and components.
2. We successfully deployed an integrated LiDAR and IMU fusion SLAM algorithm framework with 3D autonomous exploration capabilities on the robot and conducted targeted optimizations for this algorithm based on our robot, achieving superior performance on our hardware platform.
3. Our research findings have been made available as open-source, providing a high-performance solution for SLAM research under budget constraints and facilitating the wider adoption and application of advanced SLAM technologies.

The structure of the remaining sections in this paper is delineated as follows: Section 2 delves into the technical background of existing SLAM solutions, highlighting their strengths and delineating their limitations in terms of cost and real-time performance. Section 3 describes our methodology for developing a cost-effective SLAM system, focusing on hardware design, algorithm development and SLAM system optimizations. Section 4 presents our experimental results, demonstrating the effectiveness of our system in various scenarios. Section 5 discusses these methods in the broader context of SLAM research, summing up the advantages and disadvantages of the design. Finally, Section 6 concludes the paper with a summary of our contributions and outlines directions for further research.

2. Related Work

2.1. SLAM

The framework of SLAM systems generally consists of five parts: sensor data acquisition and processing, front-end odometry (indicated within the dashed red boundary in Figure 1), back-end optimization (indicated within the dashed blue boundary in Figure 1), loop closure detection, and map construction. Initially, LiDAR or visual sensors collect data, which is then roughly estimated for pose transformations between data frames by the front-end odometry. This is followed by global trajectory optimization through the back-end optimizer, after which precise poses are used for map construction. During the operation of SLAM, loop closure detection continuously identifies scenes traversed by the robot to eliminate cumulative errors.

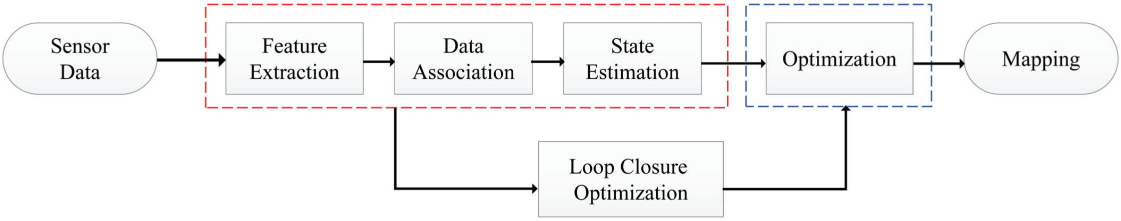


Figure 1. Pipelines of the traditional SLAM method.

Depending on the type of sensor used, mainstream SLAM systems are typically categorized into single-sensor SLAM, such as LiDAR-based [27–30] and vision-based [12–16], and multi-sensor fusion SLAM techniques [31–35], integrating various sensors to enhance SLAM performance. Visual SLAM systems, favored for their rich information acquisition, lightweight sensors, and low cost, face challenges due to their sensitivity to light, substantial accuracy variance under varying illumination, and extensive image data processing that compromises real-time performance.

LiDAR-based SLAM systems, on the other hand, effectively address these issues. LiDAR can directly measure distances with minimal influence from ambient light, providing more accurate environmental perception and capturing the spatial location and shape information of objects while delivering data that is straightforward to process. This foundation enables LiDAR SLAM systems to be more real-time, stable, and reliable. From initial 2D mapping techniques to complex 3D modeling methods, LiDAR SLAM has made significant advancements and has become a hot technology in robotics and autonomous driving, offering high-resolution, three-dimensional environmental perception and mapping capabilities. In 2012, A. Geiger, P. Lenz, and R. Urtasun introduced the KITTI Vision Benchmark Suite, followed by the release of the KITTI public dataset in 2013, intended for building and evaluating SLAM algorithms. The KITTI dataset includes real image and three-dimensional LiDAR data collected from urban, rural, and highway scenes, providing diverse environmental data for the development of 3D LiDAR SLAM algorithms, thereby accelerating their advancement.

Early 3D LiDAR SLAM algorithms heavily relied on The Iterative Closest Point (ICP) for point cloud matching and map construction. The ICP algorithm was initially proposed by Besl and McKay in 1992 [36]. It is designed to align two sets of point cloud data in three-dimensional space. The ICP algorithm seeks the optimal rigid-body transformation (including rotation and translation) through an iterative process, aligning one point cloud with another to minimize the distance between points across the two sets. The core advantage of the ICP algorithm lies in its simplicity and wide applicability, making it a crucial tool in fields such as 3D scanning and robotic vision. Over time, researchers have made various enhancements to the ICP algorithm to improve its robustness and efficiency. For instance, the introduction of a multi-resolution strategy to accelerate the alignment process [37]. In 2015, Pomerleau, Colas, and Siegwart discussed various ICP algorithms and their applications across different robotic platforms, environments, and tasks [38]. Although the ICP algorithm has been widely used for registering LIDAR scan data, its efficiency issues persist. The authors of [39] propose a method to accelerate the obstacle detection process by directly monitoring outliers detected after ICP matching, incorporating an improved ICP implementation within the SLAMICP library.

With the growing demand for more complex spatial understanding, the simple ICP algorithm, due to its high computational load and long processing time, no longer meets the real-time requirements of SLAM algorithms. The shift towards 3D LiDAR SLAM is increasingly evident, highlighted by the introduction of pioneering frameworks such as LOAM (LiDAR Odometry and Mapping) [40] and its variants [29,30], which strike a good balance between real-time processing and mapping accuracy. Despite these advancements, LiDAR SLAM still faces challenges, especially in dynamic environments and during extended operations, where cumulative errors can pose significant obstacles. Moreover, the typically high expenses associated with LiDAR SLAM systems, along with the significant computational resource demands of LOAM, present challenges for the broader adoption of LiDAR SLAM systems.

LiDAR-inertial odometry [20], which incorporates Inertial Measurement Units (IMUs), bolsters autonomous navigation systems by offering a synergy between LiDAR's detailed 3D environmental mappings through laser scanning and the IMU's motion data, including acceleration and rotational changes. This fusion enhances positioning and mapping with greater accuracy and robustness, compensating for the inherent limitations of each sensor type. Furthermore, while the prevalent LiDAR Odometry and Mapping (LOAM) [28] framework somewhat improves 3D LiDAR SLAM capabilities, its reliance on high-precision LiDAR and susceptibility to dynamic objects and environmental variations poses significant challenges.

As sensor technology progresses and hardware capabilities improve, there is a growing trend in the volume of data that SLAM systems must process. Consequently, there is a gradual increase in the demand for computational resources by algorithms. In this context, the challenge of ensuring algorithm performance and system stability while effectively controlling costs has become a significant concern in the development of SLAM technology. Faced with increasingly complex application scenarios and expanding data processing requirements, exploring more cost-effective computational methods and optimizing algorithms to reduce dependence on high-performance hardware holds practical significance for swiftly adapting to the ongoing evolution in the SLAM domain.

2.2. Autonomous Exploration

Robotic autonomous exploration enables robots to independently navigate unknown environments without human intervention, relying on their own sensors, algorithms, and decision-making capabilities. This process involves several steps, including environmental perception, map construction, path planning, and decision execution. While autonomous exploration is often considered a distinct research area akin to SLAM, recent years have seen a growing interest in integrating autonomous exploration into SLAM systems, demonstrating heightened potential in terms of robotic intelligence and integration [41].

Common autonomous exploration algorithms employ a breadth-first search [42] strategy to identify the nearest boundary points to the robot, followed by path planning to navigate toward these points. If a boundary point is reached or deemed unreachable within a certain period, the algorithm proceeds to the next search cycle, repeating this process until a complete environmental map is obtained. While effective in many settings, breadth-first-based autonomous exploration algorithms can lead to frequent changes in the robot's movement and an excessive increase in data volume, placing significant strain on the computational resources of resource-constrained mobile robots.

RRT Exploration [43,44], which is based on the Rapidly exploring Random Tree path planning algorithm [26], offers significant improvements in search efficiency and adaptability to dynamic environments over traditional breadth-first search strategies [45]. It begins from an initial node and randomly selects points within the exploration area, extending new nodes towards these points in the absence of obstacles and incorporating them into the tree. This process enables RRT to quickly cover the entire exploration space, generating paths from the start to any point. However, both traditional exploration algorithms and RRT Exploration are generally confined to 2D operations using Scan LiDAR, which limits their effectiveness in 3D environments.

The 3D RRT algorithm [46], an extension of the traditional RRT, represents a significant advancement in the field of autonomous exploration in three-dimensional spaces, specifically tailored for 3D path planning challenges. It employs heuristic strategies to guide the random selection of target points across the 3D space, expanding nodes across all three dimensions (x, y, and z). Following the addition of an endpoint, 3D RRT conducts collision checks to ensure that the path from the nearest node to the new node does not intersect with any obstacles. Upon passing the collision test, the new node is integrated into the tree connected to the nearest node by an edge. The 3D RRT framework efficiently explores and generates paths from the starting point to the endpoint within 3D spaces, rapidly adapting to complex environments without demanding excessive computational resources, offering a notable performance advantage over other more costly and complex autonomous exploration systems. Although the 3D RRT algorithm represents a significant advancement in enhancing autonomous exploration within three-dimensional spaces, its wider implementation is constrained by considerations of cost, computational resource allocation, and practical deployment challenges. Similar autonomous exploration algorithms like [47] that operate in 3D spaces require high-performance computational capabilities for constructing three-dimensional maps and planning exploration paths, potentially leading to increased operational costs, especially for extensive or prolonged tasks.

Furthermore, the resource limitations of mobile robotic platforms often restrict the extent to which such advanced algorithms can be implemented without compromising other essential functions. Balancing the requirements of sophisticated autonomous exploration algorithms like 3D RRT with the practicalities of cost and computational efficiency remains a key challenge, necessitating innovative solutions to render these technologies more accessible and feasible for broad application.

3. Method

3.1. Cost-Effective Hardware Design

In the hardware design of SLAM systems, prioritizing cost efficiency is paramount, yet not at the expense of performance and reliability. Currently, there are two main solutions for ground robots: the first approach utilizes a three-wheeled design [48] consisting of two active drive wheels and one or more passive support wheels (similar to a robotic vacuum), facilitating simple steering and navigation in confined spaces. Although this design simplifies the control system, it also limits dynamic stability and load-bearing capacity. The alternative solution involves a four-wheeled vehicle with Mecanum wheels [49], which allows for omnidirectional motion. Vehicles equipped with Mecanum wheels offer enhanced maneuverability and superior load-bearing capabilities. Yet, they are susceptible to odometry drift from slippage, and their unique structure and control mechanisms result in higher maintenance costs.

Addressing the limitations of existing solutions, this paper draws inspiration from automotive structures, adopting a four-wheeled rubber tire and custom carbon plate chassis to enhance the vehicle's stability and load capacity through the superior ground traction of rubber tires and the high structural integrity of carbon plates. Departing from the traditional Ackermann steering geometry found in automobiles, this vehicle incorporates four fixed DC geared motors, a configuration that simplifies the mechanical structure and reduces component count, thereby significantly lowering manufacturing and maintenance costs while ensuring agility and stability. The ability to independently control the direction and speed of each motor endows the system with effective spot-turning capabilities, essential for meticulous steering and path planning in restricted areas. The base plate's design features a 17:11 aspect ratio and a triple-layered structure, expanding sensor installation space while optimizing body volume and weight to achieve a balance between high performance and minimized cost and size. The result is shown in Figure 2, and The specific parameters of the radar sensors used in this SLAM system are shown in Table 1.

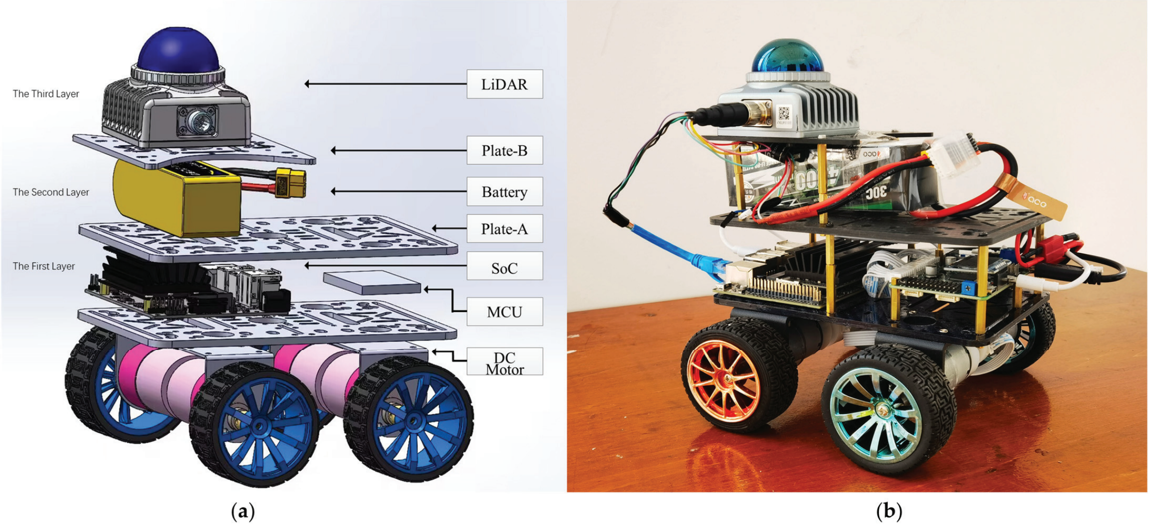


Figure 2. The results of the hardware design for the system, where (**a**) shows the vertical structure diagram of the cart, annotating the structure and items placed on each layer, and (**b**) depicts an actual photo of the system.

Benefiting from the recent cost reduction in LiDAR and single-board computers, our system integrates high-performance sensors within a compact design. The main computing unit employs the Jetson Nano ($129) paired with the MID-360 ($556.44), forming the core of our system. This cost-efficient configuration facilitates the deployment of advanced SLAM capabilities in budget-constrained applications, thereby democratizing sophisticated navigation and mapping technologies. The total expenses for the hardware system can be seen in Table 2.

Table 1. Relevant information about the MID-360 LiDAR scanner used in our system.

Specification	Description
Laser Wavelength	905 nm
Detection Range (@ 100 klx)	40 m @ 10% reflectivity 70 m @ 80% reflectivity
Close Proximity Blind Zone [2]	0.1 m
Point Rate	200,000 points/s (first return)
Frame Rate	10 Hz (typical)
IMU	ICM40609
FOV	Horizontal: 360° Vertical: −7°~52°
Range Precision [3] (1σ)	≤2 cm [4] (@ 10 m) ≤3 cm [5] (@ 0.2 m)
Angular Precision (1σ)	<0.15°

[2] Target objects within 0.1 to 0.2 m from Mid-360 can be detected and point cloud data can be recorded. However, since the detection precision cannot be guaranteed, the data should be taken as a reference only. [3] To detect objects having different reflectivities within the detection range, the accuracy of point cloud data of very few positions might decrease slightly. [4] Tested in an environment at a temperature of 25 °C (77 °F) with a target object that has a reflectivity of 80% and is 10 m away from Livox Mid-360. [5] Tested in an environment at a temperature of 25 °C (77 °F) with a target object that has a reflectivity of 80% and is 0.2 m away from Livox Mid-360. For target objects within 0.1 to 1 m away from the Mid-360, if they have a low reflectivity or are thin and tiny, the detection effect cannot be guaranteed. These objects include but are not limited to black foam and the surface of water or objects that have been polished, have a matte finish, thin lines, etc.

Table 2. Cost breakdown of SLAM system components.

Component	Description	Quantity	Unit Cost ($)	Total Cost ($)
Mid-360	3D LiDAR sensor with IMU	1	749	556.44
Jetson nano 4 GB	SoC, Data Processing Unit	1	129	129
STM32F407VET6	MCU, ROS base plate master control	2	23.5	47
carbon plate-A	Body structure, porous rectangles	1	20	20
carbon plate-B	LiDAR Support Structure	1	5	5
MG513 motor	DC-coded motor, with rubber wheel	4	6.97	27.89
Battery	4000 mAh-30C and 1200 mAh-45C			45
Others	All kinds of wire and copper column			10
Total Cost				840.33

3.2. LiDAR-Inertial Odometry Using Fast-ICP

3.2.1. SLAM Framework Design

Due to the need for localization in high-dynamic environments, to maintain computational efficiency, and to mitigate sensor noise and data ambiguity, 3D LiDAR SLAM struggles to ensure real-time performance at low cost, especially traditional odometry frameworks that are unable to allow robust navigation and mapping. So, we propose an efficient method to ensure robustness and balance performance with cost-effectiveness.

We adopt the high-performance LiDAR-inertial odometry framework FAST-LIO in [31] as our foundation to build up our SLAM system. Different from [31], we incorporate an ICP optimization module into it [36,37] that utilizes KD-trees for fast nearest-neighbor searches and employs the Gauss–Newton iterative method, as illustrated in Figure 3 (on the right side). Our approach comprises four main stages. Initially, the system collects point cloud and IMU data from the Mid-360 LiDAR and preprocesses the LiDAR data. Subsequently, extracted features are fed into the state estimation module for initial state estimation and vehicle localization. Following that, the Fast ICP algorithm [37] precisely aligns the current point cloud frame with the previous one, optimizing state estimation and setting the stage for precise mapping. Ultimately, the map is updated using the optimized state estimates, continuously integrating new point cloud data into the existing map for real-time maintenance. The overview of our workflow is shown in Figure 3.

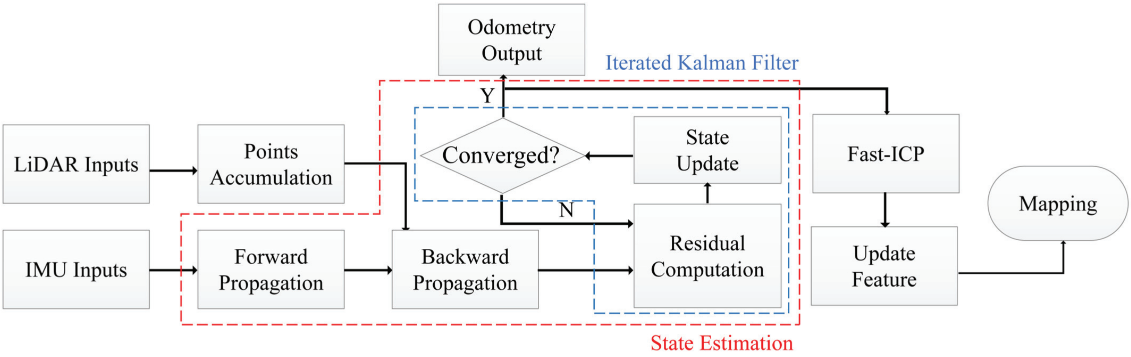


Figure 3. An illustrative representation of the Fast-ICP enhanced LiDAR-inertial odometry structure.

3.2.2. State Estimation Based on State Iterated Kalman Filter

State estimation plays a pivotal role in SLAM, involving the determination of an object's orientation and position within a given environment. Traditional state estimation algorithms, which rely on feature extraction and matching, excel in environments with distinct landmarks but falter in dynamic or feature-sparse areas owing to occlusion risks and extensive computational demands. Some works directly enhance SLAM with real-time 3D mapping via LiDAR yet falter in challenging conditions or under computational constraints due to reliance on high-quality data and intensive processing [28–30].

We choose to employ a tightly coupled iterated Kalman filter [35,50] for enhanced State estimation within our SLAM system. This filter is particularly adept at handling the uncertainties and dynamics of real-world environments by cohesively integrating sensor measurements. This method not only addresses the challenges of occlusions and feature scarcity but also ensures robust performance despite computational constraints. This seamless integration of sensor data, particularly from LiDAR and IMU, forms the basis for our approach to handling the high-frequency data and temporal alignment challenges inherent in real-time SLAM applications.

Given the original LiDAR's high sampling rate, processing each point in real-time is impractical. Therefore, we accumulate these points over a period, and an accumulated set of points before collective processing is referred to as a scan. Following each scan's preprocessing, the data, along with that from the IMU, is input into the Kalman filter for Forward Propagation. This involves state prediction using the system's predictive equations under the assumption of zero noise:

$$\hat{x}_{i+1} = \hat{x}_i \boxplus (\Delta t f(\hat{x}_i, u_i, 0)); \hat{x}_0 = \overline{x}_{k-1}. \tag{1}$$

In this context, $\hat{x}_i$ represents the propagated vector at the i-th IMU sample time in a LiDAR scan, $\overline{x}_k$ is the update vector at the scan-end time of the k-th LiDAR scan, Δt denotes the time difference between two IMU frames, and f describes the dynamic system model governing state changes within the paper. Encapsulation operators $\boxplus$ defined in article [31] can support formulation in rotation matrices as well as manifolds. Let $\mathcal{M}$ be the manifold of dimension 15 in consideration, and they are defined as follows:

$$\begin{gathered}
\mathcal{M} = SO(3) \times \mathbb{R}^{15},\ dim(\mathcal{M}) = 18 \\
x = \left[{}^{G}R_{I}^{T} \quad {}^{G}p_{I}^{T} \quad {}^{G}v_{I}^{T} \quad b_{\omega}^{T} \quad b_{a}^{T} \quad {}^{G}g^{T}\right]^{T} \in \mathcal{M} \\
u = \left[\omega_{m}^{T} \quad a_{m}^{T}\right]^{T}, w = \left[n_{\omega}^{T} \quad n_{a}^{T} \quad n_{b\omega}^{T} \quad n_{ba}^{T}\right]^{T} \\
f(x_i, u_i, w_i) = \begin{bmatrix} \omega_{m_i} - b_{\omega_i} - n_{\omega_i} \\ {}^{G}V_{I_i} \\ {}^{G}R_{I_i}(a_{m_i} - b_{a_i} - n_{a_i}) + {}^{G}g_i \\ n_{b\omega_i} \\ n_{ba_i} \\ 0_{3\times 1} \end{bmatrix}
\end{gathered} \tag{2}$$

where ${}^{G}R_{I}^{T}$, ${}^{G}p_{I}^{T}$ are the transposed attitude and position of IMU in the global frame, ${}^{G}g^{T}$ is the unknown transposed gravity vector in the global frame, ω_{m}^{T} and a_{m}^{T} are the transposed IMU measurements, n_{ω}^{T} and n_{a}^{T} are the transposed white noise of IMU measurements, and b_{ω}^{T} and b_{a}^{T} are the IMU bias modeled as the random walk process with Gaussian noises $n_{b\omega}^{T}$ and n_{ba}^{T} [31].

Then, the covariance matrix prediction equation is calculated:

$$\hat{P}_{i+1} = F_{\tilde{x}} \hat{P}_i F_{\tilde{x}}^{T} + F_w Q F_w^{T};\ \hat{P}_0 = \hat{P}_{k-1}. \tag{3}$$

$F_{\tilde{x}}$ and F_w, the system conversion matrix and noise conversion matrices, are computed by the error state dynamic model, with the detailed calculation process also detailed in the article's appendix [31]. Following this step, an initial estimate of the system's current state is obtained.

It is important to note that since a consists of many points, these points are clearly not measured at the same time, making it challenging to perfectly align the time between the IMU and the LiDAR. To address this issue, we perform backward propagation of the equations to compensate for the motion errors caused by the time difference from Equation (1):

$$\check{x}_{j-1} = \check{x}_j \boxplus \left(-\Delta t f\left(\check{x}_j, u_j, 0\right)\right). \tag{4}$$

While the last three lines (acceleration noise) are zero in Equation (2), backward propagation can be simplified as:

$$\begin{gathered}
{}^{I_k}\check{p}_{I_{j-1}} = {}^{I_k}\check{p}_{I_j} - {}^{I_k}\check{v}_{I_j}\Delta t, \qquad\qquad s.f.\ {}^{I_k}\check{p}_{I_m} = 0; \\
{}^{I_k}\check{v}_{I_{j-1}} = {}^{I_k}\check{v}_{I_j} - {}^{I_k}\check{R}_{I_j}\left(a_{m_{i-1}} - \hat{b}_{a_k}\right)\Delta t - {}^{I_k}\hat{g}_k\Delta t, \\
s.f.\ {}^{I_k}\check{v}_{I_m} = {}^{G}\hat{R}_{I_k}^{T}\ {}^{G}\hat{v}_{I_k},\ {}^{I_k}\hat{g}_k = {}^{G}\hat{R}_{I_k}^{T}\ {}^{G}\hat{g}_k; \\
{}^{I_k}\check{R}_{I_{j-1}} = {}^{I_k}\check{v}_{I_j} Exp\left(\left(\hat{b}_{\omega_k} - \omega_{m_{i-1}}\right)\Delta t\right),\ s.f.\ {}^{I_k}R_{I_m} = I.
\end{gathered} \tag{5}$$

Among them, s.f. represents "starting from", which defines the relative pose of the point obtained by backward propagation as ${}^{I_k}\check{T}_{I_j} = \left({}^{I_k}\check{R}_{I_j}, {}^{I_k}\check{p}_{I_j}\right)$. We should project the point ${}^{L_j}p_{f_j}$ to the scan end time t_k:

$${}^{L_k}p_{f_j} = {}^{I}T_{L}^{-1}\ {}^{I_k}\check{T}_{I_j}\ {}^{I_k}T_{L}\ {}^{L_j}p_{f_j}. \tag{6}$$

This process adjusts the previous state estimates, compensating for the motion that occurred between two LiDAR scans, ensuring temporal consistency between the LiDAR data and the system's state estimate [31].

After we compensate for the motion distortion of the LiDAR, the received points are used for residual calculation. Calculating the residuals involves determining the difference between the actual observations and the predicted observations based on the current state estimate and assessing the accuracy of the current state estimate. Based on the results of the residual calculation, the system iteratively updates the state estimate. When the residuals are below a specific value, the iteration reaches the termination condition and completes the state estimation process once.

3.2.3. Fast ICP for Improved State Estimation

IMU drift can become significant over extended periods, potentially impacting outcomes, as discussed in Section 3.2.2. To further enhance the SLAM system's mapping precision and robustness in complex environments, this stage utilizes the Fast ICP algorithm to refine the current state estimation [37], thereby alleviating point cloud inconsistencies caused by IMU drift.

Integrating Fast ICP post-initial state estimation, our system introduces a vital refinement phase before the final map synthesis. The process starts by identifying the closest point correspondences between the latest point cloud and the existing map, leveraging efficient spatial data structures for rapid nearest-neighbor searches. For each point p_i in P, Fast ICP seeks the closest point m_j in M using KD-trees [51], a type of spatial data structure, resulting in pairs (p_i,m_j). Once the point correspondences are established, the alignment error is quantified through a cost function E, typically defined as the sum of squared distances between paired points:

$$E(R,t) = \sum_{i=1}^{N} \| Rp_i + t - m_j \|^2 \tag{7}$$

R represents the rotation matrix, while t is used to symbolize the translation vector. Typically, two sets of point clouds are not identical, hindering the precise determination of rotation R and translation t matrices. However, by formulating this loss function, we transform the point cloud registration challenge into an optimization problem [36]. In order to determine the optimal R and t to minimize E, employing iterative methods like the Gauss–Newton algorithm:

$$\begin{bmatrix} \Delta R \\ \Delta t \end{bmatrix} = -\left(J^T J\right)^{-1} J^T \nabla E \tag{8}$$

where J is the Jacobian matrix of partial derivatives of E with respect to R and t, and ∇E is the gradient of E. In each iteration, the pose parameters are updated based on the optimization step:

$$R \leftarrow R + \Delta R, t \leftarrow t + \Delta t \tag{9}$$

Iterative enhancements continuously refine the pose, recalibrate point correspondences, and reduce alignment errors until convergence is achieved. This method guarantees accurate pose estimation despite significant IMU drift, maintaining map integrity and continuity.

Each iteration involves processing extensive point cloud data and complex mathematical operations, requiring multiple iterations to converge on a stable solution. Frequent optimization using all points in the Fast ICP module for large-scale point cloud data can significantly consume computational resources and impact the system's real-time performance, and some pairs may be mismatched due to nearest neighbor search errors, point cloud occlusion or data quality issues, thereby affecting convergence speed and increasing memory usage. To address this issue, the ICP module performs down-sampling using normal-space sampling before processing [37]. It first calculates the normal for each point and maps these normals onto a unit sphere. The sphere is then divided into several regions (bins), from which points are uniformly selected to ensure an even distribution in the normal direction. This method effectively reduces the number of points while preserving the surface's key geometric features and structures, improving the convergence speed and effectiveness of Fast-ICP. Simultaneously, outlier pairs are removed using the standard deviation method to improve alignment accuracy and algorithm convergence speed.

Additionally, this module does not always run continuously; instead, it flexibly adjusts whether to perform the Fast ICP module for drift correction based on the point cloud matching error. This approach allows for rapid approximation of optimal drift correction while reducing resource usage. We set an initial error threshold at the start of the LiDAR-inertial odometry to assess the combined pose estimation of the odometry and IMU. If the error exceeds the threshold, the Fast ICP module intervenes to optimize pose estimation, correct drift, and ensure efficient and accurate alignment. When the error decreases to an acceptable range and remains stable, the threshold is dynamically adjusted, and iterations are suspended to ensure the Fast ICP module does not significantly increase resource consumption. This approach intelligently balances pose optimization and computational efficiency.

Pose optimization via Fast ICP effectively counters IMU drift, enhancing accuracy and resilience during the mapping phase, compensating for sensor errors, and adapting to environmental dynamics. This achieves localized optimization of existing algorithms and flexible adaptation to our hardware system without compromising the real-time performance of the SLAM system.

3.2.4. Map Update

Sections 3.2.2 and 3.2.3 set the stage for 3D map construction. This process involves projecting each feature point onto the coordinate system of the IMU and then transforming these points into the real-world coordinate system to facilitate map updates. Upon completion of this process, the feature points in the global coordinate system are appended to the existing map, incorporating all feature points from prior stages [31]. The map is continuously updated by integrating new observational data, thus augmenting the SLAM system's ability to construct environmental models.

3.3. 3D Auto-Exploration Using RRT

In the context of SLAM, 3D auto-exploration stands as a pivotal component, particularly for Unmanned Aerial Vehicles (UAVs) and robotic systems navigating complex environments. When developing the autonomous exploration algorithm, it is essential to focus on the algorithm's real-time performance, efficiency, and robustness to suit our low-cost SLAM system. This entails real-time updates, environmental map maintenance, and efficient path planning to navigate uncharted territories with minimal energy expenditure. The exploration challenge centers on expanding the known spatial domain strategically selecting unexplored areas to maximize information gain. Path planning, on the other hand, concentrates on directing the robot or UAV along an optimal path efficiently and safely, taking into account actual physical constraints and obstacles within the environment. This is crucial in our low-cost SLAM system to ensure efficient exploration tasks with lower memory usage.

To obtain the optimal trajectory for autonomous exploration, we have coupled the two problems into a minimization issue:

$$\begin{aligned} Minimize\ J_a(u) + J_d(x) + J_e(v) \\ subj.\ to:\ x \subset V_{free} \\ J_e(v) \neq 0 \end{aligned} \tag{10}$$

In the equation, $J_a(u)$ denotes the actuation cost, $J_d(x)$ represents the distance cost, and $J_e(v)$ signifies the exploration or information gain cost. The obstacle-free space, $V_{free} \subset V_{map}$, indicates the navigable area within the three-dimensional positional space contained by the current map configuration, $V_{map} \subset \mathbb{R}^3$. By finding the minimum values, the optimal trajectory for the exploration task is determined, delicately balancing between swiftly discovering more space, limiting actuation based on a dynamic system model, and minimizing effort in movement. For efficient and precise exploration path derivation, we employ a multi-goal RRT framework, utilizing Nonlinear Model Predictive Control

(NMPC) [52] to optimize path execution, minimizing energy consumption and ensuring path feasibility [53]. The exploration framework is shown in the following Figure 4.

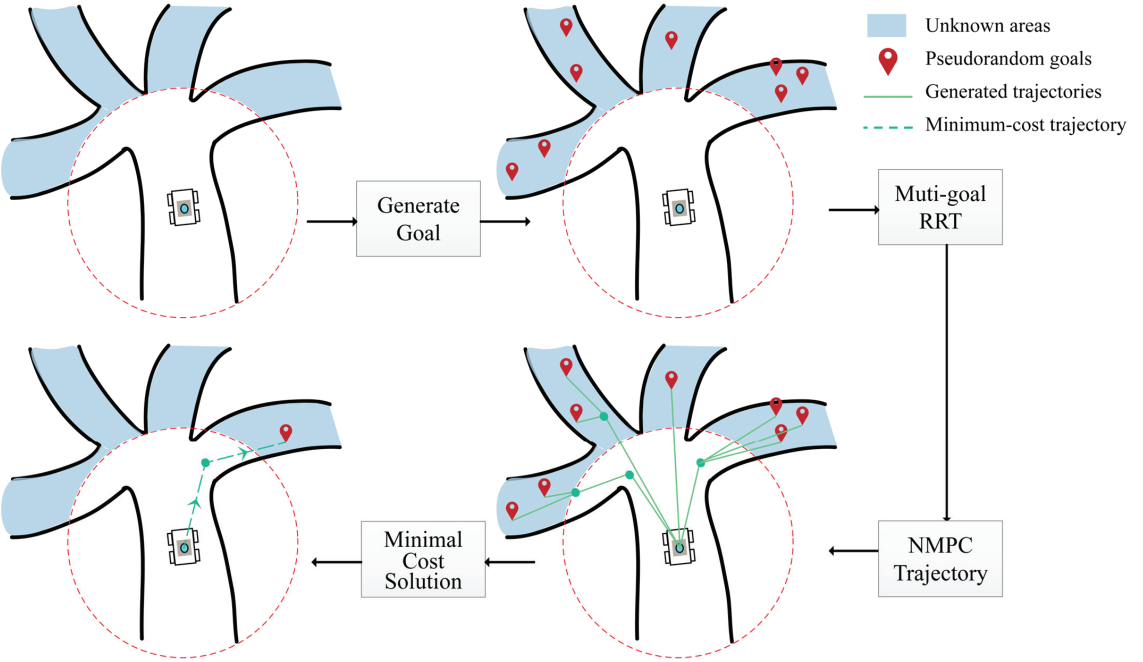


Figure 4. The 3D multi-goal RRT framework using the NMPC Trajectory Optimizer.

3.3.1. 3D Multi-Goal RRT

The RRT algorithm [26] is a path-planning method that expands branches from a root node towards unexplored areas, featuring inherent randomness that enables effective navigation through unpredictable terrains. The advantages of RRT lie in its computational efficiency and the ability to directly integrate additional functionalities into the core planning process [54]. However, a limitation is the inability to guarantee the shortest path to the target within a limited number of iterations. Additionally, autonomous exploration schemes based on RRT are more prevalent in two-dimensional spaces, with less application in three-dimensional spaces.

To address this issue, this paper employs a 3D Multi-goal RRT framework. This framework, grounded in the 3D RRT structure, utilizes multiple random goals to enable efficient navigation through complex environments by iteratively constructing a space-filling tree [53]. The multi-goal RRT framework establishes multiple objectives, facilitating simultaneous expansions in various directions with each iteration, thereby boosting exploration efficiency and path comprehensiveness. First, define the root node of the RRT tree based on the current position of the vehicle and generate random target coordinates through probabilistic sampling. Next, search the existing tree structure to find multiple nodes that are closest to the random target point, and each node will expand to a new node in the direction of the target. Then, traversability checks on the path between the new node and the nearest node are performed parallelly. If there are no obstacles or other hindrances along the path, add the new node to the tree as a new branch, increasing the RRT tree's coverage and node density throughout the search space. Otherwise, abandon the target point and re-sample randomly, ensuring the tree continues expanding into unexplored areas. By repeating this process continuously, the RRT tree gradually explores the entire space, ultimately covering the target area with a certain resolution and path quality while meeting predefined exploration termination criteria.

While the multi-goal 3D RRT considers the vertical dimension, our mobile robots operate only on the ground plane in practice. Therefore, our approach optimizes the 3D RRT for 2D planar motion by constraining vertical expansion to a specific height range, ensuring the generated paths align with the robots' actual mobility. The system confines target point generation, path planning, and motion control to a plane, implementing a 3D to 2D filtering prior to the exploration process depicted in Figure 4. Once the exploration path for the vehicle is determined, it transitions back to three-dimensional space for reconstruction. This approach not only simplifies the path planning problem but also improves computational efficiency by reducing the spatial dimensions that the algorithm must account for.

3.3.2. NMPC Trajectory Optimizer

To optimize path execution, ensure feasibility, and minimize energy consumption, we employ an NMPC strategy [52]. NMPC is an advanced control strategy that utilizes the dynamic model of the system to predict its behavior over a future time horizon and calculates control inputs by optimizing this predicted behavior. This method is particularly suited for dealing with complex, highly nonlinear system dynamics and can accommodate the constraints of the system.

Building upon a 3D multi-objective RRT framework for path planning, NMPC designs optimal control strategies for each segment of the planned path. Firstly, based on the robot's dynamic model and current state, establish a nonlinear model for it:

$$\begin{bmatrix} x_i \\ y_i \\ \theta_i \end{bmatrix} = \begin{bmatrix} x_{i-1} + v_i t cos\theta_{i-1} \\ y_{i-1} + v_i t sin\theta_{i-1} \\ \theta_{i-1} + \omega_i t \end{bmatrix} \tag{11}$$

where x_i, y_i, and θ_i represent the initial coordinates (x-coordinate and y-coordinate) and heading angle of the robot in the global coordinate system, i is the index of the robot's movement state, and v and ω denote the robot's linear and angular velocities, respectively. From the relationship between the target point and the robot's position, we can obtain the error:

$$\begin{bmatrix} E_{cte_i} \\ E_{\theta_i} \end{bmatrix} = \begin{bmatrix} f(x_i) - y_i \\ arctan(f'(x_i)) - \theta_i \end{bmatrix} \tag{12}$$

where E_{cte_i} and E_{θ_i} represent the linear and angular differences between the robot and the target point in the global coordinate system, and $f(x)$ is used to fit a smooth curve to the path. We then define the objective function:

$$\begin{aligned} \min_{U} J = \sum_{i=1}^{n} [q_1(E_{cte_i})^2 + q_2(E_{\theta_i})^2 + q_3(v_i)^2 + q_4(v_i^2 + \omega_i^2) \\ + q_5((v_{i+1} - v_i)^2 + (\omega_{i+1} - \omega_i)^2)] \end{aligned} \tag{13}$$

q_1, q_2, q_3, q_4, and q_5 are weight parameters used to adjust the motion error in the execution path and the motion feedback control inputs. The q_1 and q_2 terms represent the cross-track error and heading error, respectively. The q_3 term represents the speed error, maintaining the target speed. The q_4 term is the weight coefficient for linear and angular velocities, limiting the vehicle's linear and angular velocities to prevent it from going too fast or turning too sharply. The q_5 term is the weight coefficient for changes in linear and angular velocities, smoothing the control inputs to avoid sudden speed and steering changes, thereby improving driving stability. We constructed an NMPC problem model using the geometric kinematics model of unmanned vehicles. To minimize model loss and improve the accuracy and real-time performance of trajectory tracking, we set our weight parameters as follows:

$$\begin{aligned} \underset{U}{min}\, J = \sum_{i=1}^{n} \Big[& 1000(E_{cte_i})^2 + 1000(E_{\theta_i})^2 + 100(v_i)^2 + 10\big(v_i^2 + \omega_i^2\big) \\ & +1\big(\big(v_{i+1} - v_i\big)^2 + \big(\omega_{i+1} - \omega_i\big)^2\big)\Big] \end{aligned} \tag{14}$$

Larger q_1 and q_2 ensure precise control of the path and heading, while a moderate q_3 ensures speed stability. These parameter settings provide a balance between path-tracking accuracy, speed stability, control input smoothness, and change smoothness. The NMPC module solves the predicted error using the objective function to minimize the error, thereby optimizing the path planned by the RRT and assisting the robot in exploration tasks.

The key advantage of NMPC is its ability to ensure the feasibility of the path while considering the robot's motion and operational constraints, proactively avoiding potential obstacles, and averting unnecessary energy expenditure and potential collision risks. Moreover, through a continuous optimization process, NMPC dynamically adjusts the robot's velocity and position, ensuring that the robot travels most economically along the planned path, thereby significantly enhancing the efficiency and performance of the entire system.

4. Experiment

To illustrate the cost-effectiveness of our SLAM system in comparison to other market offerings, we provide a detailed price-performance analysis of some mainstream robots on the market in Table 3 (prices and parameters are from official information as of March 2024).

Table 3. Comparison of SLAM robot models by cost and capabilities.

Model	Robot Type	LiDAR SLAM Dimension	SoC/CPU	Auto Exploration	Cost ($)
Turtlebot4	Two-wheeled mobile robot	2D LiDAR	Raspberry Pi 4B	No	2191.44
Hiwonder JetAuto Pro	Omnidirectional mobile robot	2D LiDAR	Jetson nano	2D	1399.99
SLAMTEC Hermes	Mobile robot with 2-wheel hub motor	2D LiDAR	Unknown	No	3061.17
Unitree Go 2	Robot dog	3D LiDAR	8 core CPU	Unknown	2588.08
WEILAN AlphaDog C 2022	Robot dog	3D LiDAR	ARM 64 bit	Unknown	5134.41
Ours	Four-wheeled mobile robot	3D LiDAR	Jetson nano	3D	840.33

The table shows that most wheeled robots are only compatible with 2D LiDAR SLAM, while robots equipped with 3D LiDAR SLAM tend to have more complex motion control algorithms, primarily seen in quadruped robots. Despite similar CPU performance, SLAM systems supporting autonomous exploration remain scarce. In contrast, our system, featuring a stable and simple four-wheel mechanical structure, natively supports 3D LiDAR SLAM and autonomous exploration capabilities at a significantly lower price point compared to various LiDAR SLAM robots available in the market. These comparison results verify the highly cost-efficient performance of our scheme, underscoring our system's affordability without sacrificing performance, highlighting our technological advancements and strategic component selection that contribute to a more economical SLAM solution.

4.1. Dataset Collection

To comprehensively cover complex indoor scenarios and validate the localization and mapping performance of the SLAM system, we controlled a small vehicle to collect data along main routes in both indoor and outdoor environments. All experimental data were collected from Wuhan, China, encompassing simple indoor scenes, complex indoor scenes, and large-scale mixed indoor-outdoor settings. The vehicle traveled at a speed of approximately 0.5 m per second through these scenarios, circling around the buildings or scenes before returning to the starting point. The LiDAR and IMU were mounted approximately 20–22 cm above the ground. The scanning rate of the LiDAR was set to 10 Hz, with an average processing time of 25 ms per scan and 20,000 points per scan. The mixed scenes included terrains at varying altitudes, posing additional challenges for the SLAM system's positioning accuracy and three-dimensional reconstruction.

4.2. LiDAR SLAM Experiment

Our robot operates on ROS melodic and Ubuntu 18.04, employing C++ along with PCL 1.9.1 (https://github.com/PointCloudLibrary/pcl (accessed on 23 November 2023)) [55] and Eigen 3.3.4 (http://eigen.tuxfamily.org/index.php?title=Main_Page (accessed on 24 November 2023)) libraries to construct our LiDAR-Inertial Odometry Using Fast Iterative Closest Point (LIO-FICP) algorithm, and the data will be processed and stored locally through our program. The specific experimental environments are detailed in Table 4.

Table 4. Experimental Environment.

Name	Version
OS	Ubuntu 18.04
SoC	NVIDIA Tegra X1
RAM	4 GB
ROM	64 GB
Accelerator Library	PCL v1.9.1 + Eigen v3.3.4

4.2.1. State Location Experiment

Evaluating the positioning accuracy of the SLAM system using Absolute Pose Error (APE) is a common and effective approach. APE evaluates the global consistency of the entire trajectory as a whole, providing an intuitive understanding of the SLAM system's accuracy performance in real-world environments. APE (Absolute Pose Error) can be calculated using the following formula:

$$APE_{trans,k} = \parallel trans(P_{ref,k}^{-1} P_{est,k}) \parallel \tag{15}$$

$$APE_{rot,k} = \parallel rot(P_{ref,k}^{-1} P_{est,k}) - I_{3*3} \parallel_F \tag{16}$$

where $APE_{trans,k}$ and $APE_{rot,k}$ are the translation and rotation error, $P_{ref,k}$ and $P_{est,k}$ are the reference and estimated pose matrices at timestamp k.

We conducted experiments on the publicly available KITTI dataset, specifically on sequences 0–10, using APE as the metric for evaluating the localization accuracy of our system. Additionally, we calculated the APE for LIO-SAM and FAST-LIO2, comparing their performance with our method. The results are shown in Figure 5.

As shown in Table 5 and Figure 5, the proposed algorithm demonstrates lower translation and rotation errors compared to FAST-LIO2, with a reduction in the root mean square error (RMSE) of translation error by approximately 3.8%. When compared to LIO-SAM, there is a more noticeable decrease in error metrics, making it less likely for significant errors to occur. The experimental results suggest that the proposed method is more suitable for use in terrestrial SLAM systems due to its relatively higher robustness and stability.

Table 5. Comparative results of localization errors in the KITTI.

APE / Algorithm	Translation Error (m)				Rotation Error (Degrees)			
	Max	Mean	Min	RMSE	Max	Mean	Min	RMSE
Ours	7.104	2.541	0.192	2.872	2.479	2.434	2.346	2.435
FAST-LIO2	9.456	2.643	0.204	2.986	2.502	2.440	2.346	2.440
LIO-SAM	51.892	4.831	0.112	8.323	2.828	2.315	1.481	2.361

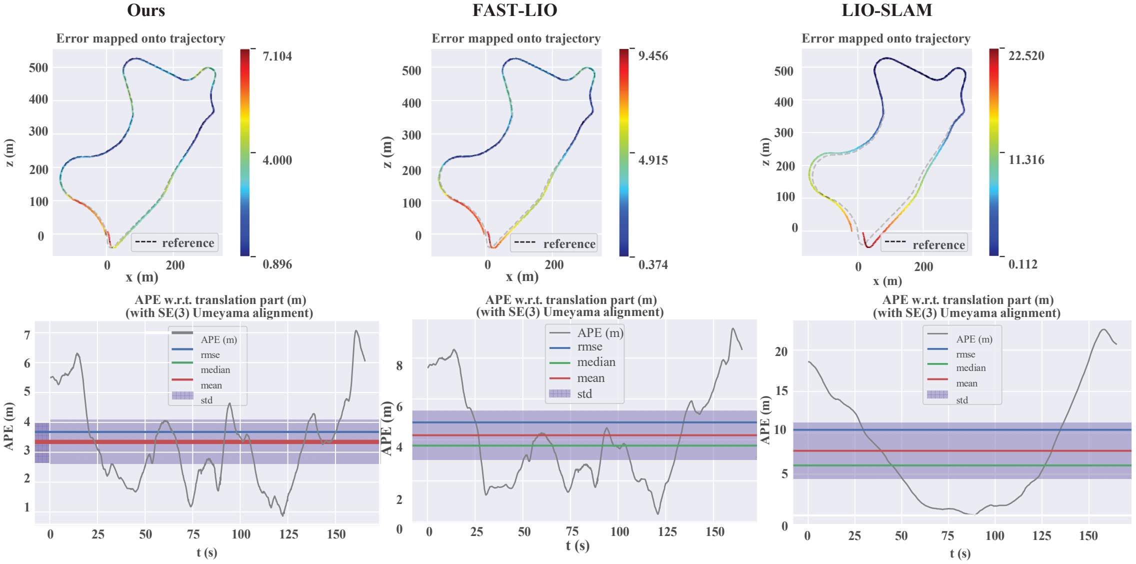


Figure 5. APE of our method in the KITTI dataset: Sequence 9, FAST-LIO, and LIO-SAM.

4.2.2. Loop Closure Experiment

The Loop Closure Experiment in SLAM is crucial in evaluating and enhancing the system's accuracy and consistency. Through this experiment, we can test the system's cumulative errors over extended operation and its robustness in dynamic environments.

In our indoor state localization experiment, we conducted a loop closure test using the collected dataset to assess the stability of the SLAM system's positioning. A robot was programmed to traverse a predetermined path of 340 m and return to the starting point. The initial and final positions estimated by the slam system were recorded to calculate the positioning error upon return, based on the distance between these two points, shown in Figure 6. To thoroughly assess the robot's state estimation capabilities, three different motion strategies were employed along the set route:

1. Motion 1. Linear movement at 0.7 m/s across a flat surface, completing a circuit and returning to the start.
2. Motion 2. Zigzagging motion with the robot swaying left and right, moving in a curved path around the room at an average speed of approximately 0.7 m/s before returning to the starting point.
3. Motion 3. Straight-line movement over uneven terrain, completing a circuit and returning to the start, maintaining an average speed of 0.7 m/s.

Figure 7 illustrates the experimental setup. Each route featured numerous pedestrians walking randomly around the system and doors opening and closing randomly to alter the experimental environment. This setup was used to test the effectiveness of the system's positioning technology under the uncertainty and dynamic conditions of real-world environments.

Each motion type was repeated three times to mitigate random errors, with the average results recorded in Table 6.

We also tested the FAST-LIO2 and LIO-SAM systems under the same dataset, with the results shown in Table 7. The experimental results indicated that the drift of this experiment is less than 0.07% (the maximum drift of 340 m is less than 0.22 m), which is nearly identical in accuracy to FAST-LIO2 and represents a significant improvement over LIO-SAM. On average, each scan identified 1421 valid feature points, enabling the SLAM system to accurately track the vehicle's position and trajectory. Even with the presence of numerous dynamic obstacles during tests, the system showcased its capability for precise state localization within complex indoor environments. It retained high accuracy and robustness against diverse terrains and obstacles.

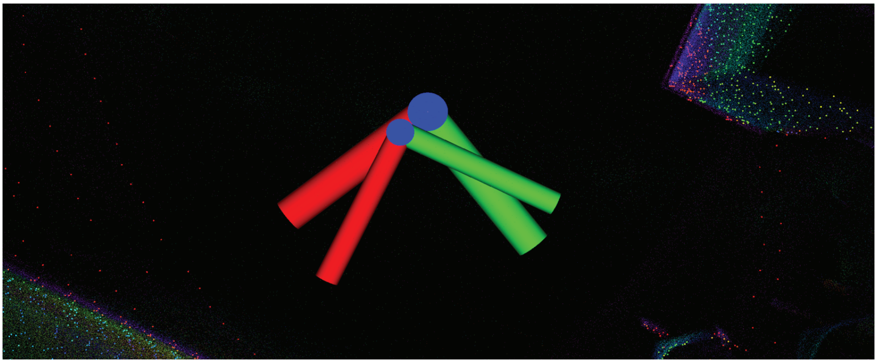


Figure 6. The vehicle's starting point (coordinate system with thicker axis) and ending point (coordinate system with thinner axis) have been marked in Figure 6. The red, blue, and green axes represent the X, Y, and Z axes, respectively. We obtain the Drift Distance by comparing the positions of the two coordinate systems in time and space.

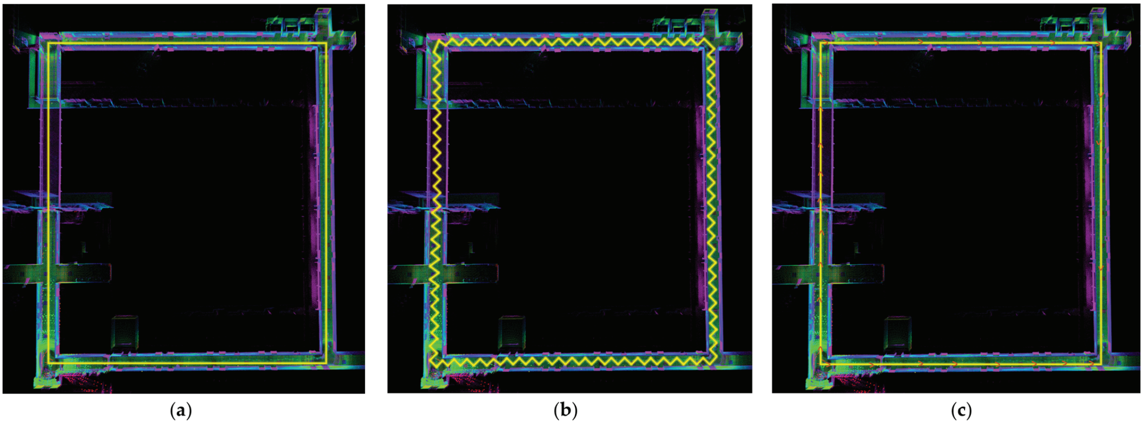


Figure 7. Schematic diagrams of the car for the closed-loop test experiment. In the figure, the yellow line represents the planned scanning route, (**a**) is motion 1, (**b**) motion 2, (**c**) motion 3, and the red curve represents uneven terrain or dynamic obstacles.

Table 6. Our method's drift results of the loop closure test experiment.

Motion Strategies	**Drift (m)**		
	X	Y	Z
Linear + Flat	0.016942	0.003476	0.157648
Zigzagging + Flat	0.016289	0.004651	0.173591
Linear + Uneven	0.016075	0.008972	0.210820

Table 7. The results of the maximum drift loop closure experiment.

Algorithm	Max Drift (m)			Accuracy
	X	Y	Z	
LIO-SAM (LoopClosutre Disable)	1.226639	0.295907	1.364643	0.54676%
FAST-LIO2	0.032126	0.016737	0.278591	0.10450%
Ours	0.016075	0.008972	0.210820	0.06235%

4.2.3. 3D Reconstruction Experiment

The most significant advantage of 3D LiDAR is its ability to provide high-precision three-dimensional spatial data, enabling this SLAM system to accomplish high-precision mapping and surveying tasks. To verify the capabilities of the LIO-FICP in this regard, we conducted indoor and outdoor ground vehicle experiments.

Outdoors, we ran our algorithm using the KITTI dataset to generate a reconstructed 3D road map, as shown in Figure 8. The map distinctly shows vehicles parked along the curb and permanent buildings, which is highly consistent with the scenes depicted in the RGB data from the dataset. During the experiment, the vehicle started from the origin and circled back to the starting point, completing a loop that is visible on the map, demonstrating the high accuracy of our mapping algorithm. Moreover, despite the presence of many moving vehicles and pedestrians along the route, dynamic objects were not represented in the mapping results, reflecting the precision and robustness of our algorithm.

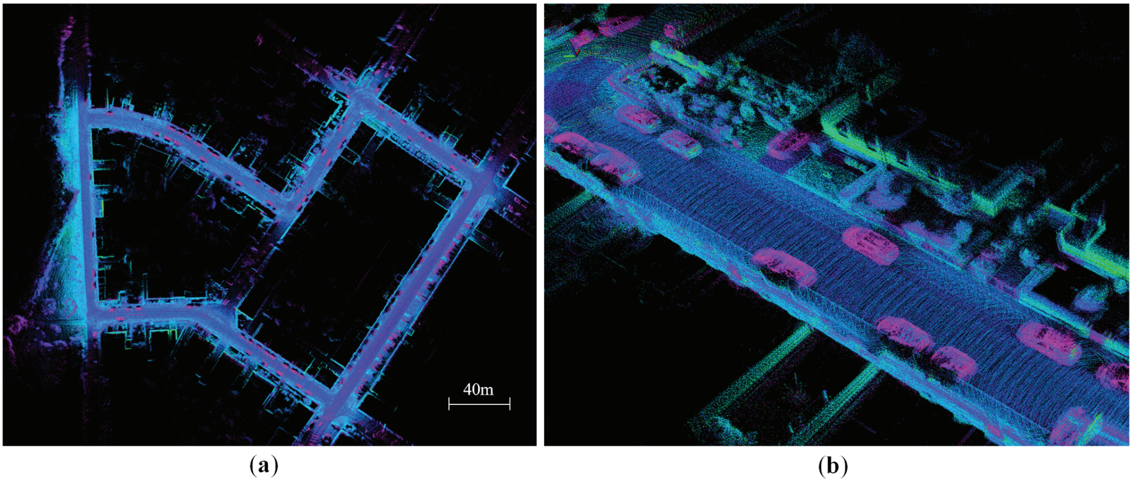


Figure 8. These are the mapping results of the outdoor environment. (**a**) The global map of the environment; (**b**) stationary vehicles on both sides of the road.

Based on a pre-constructed grid map, a route was manually planned within a 35 m × 15 m room; the arrow in Figure 9a. shows the general direction The robot circled the room before moving towards the center, reconstructing the indoor environment in the process. The room layout was a flat quadrilateral with a height of approximately 3.5 m, furnished with tables, chairs, and other irregular items, adding complexity to the mapping task. The vehicle was deliberately not directed to the room's corners to evaluate the SLAM system's comprehensive 3D reconstruction capability with limited path information. Utilizing the Fast-ICP algorithm, the LiDAR-Inertial Odometry allowed for the creation of a highly detailed room map, as shown in Figure 9b, capturing fine details like tables and chairs.

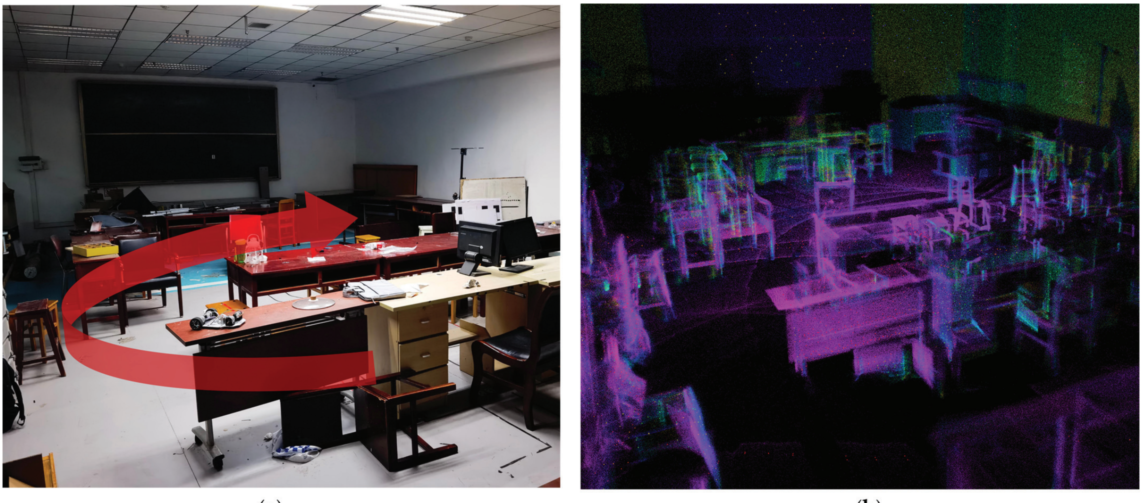

(**a**) (**b**)

Figure 9. These are figures of the indoor environment, the arrow direction represents the planned path direction. (**a**) The real scene of the test room; (**b**) the 3D reconstruction of the room.

The system was then tested in a mixed indoor-outdoor environment. Compared to simple indoor rooms, this setting offers more extensive spaces with numerous dynamic elements, necessitating the system to process more data and exhibit greater robustness. Due to the increased size of the area, longer mapping durations might lead to IMU drift, potentially causing overlaps and distortions in the building structure.

A start and end point were designated for the system, with the vehicle initiating its navigation task indoors before moving outdoors at about 0.7 m/s through the corridor to reach the predetermined endpoint. The corridor is surrounded by an open outdoor environment, significantly increasing the computational load. Figure 10 shows the actual test scenario, with the vehicle's general direction of movement indicated by arrows.

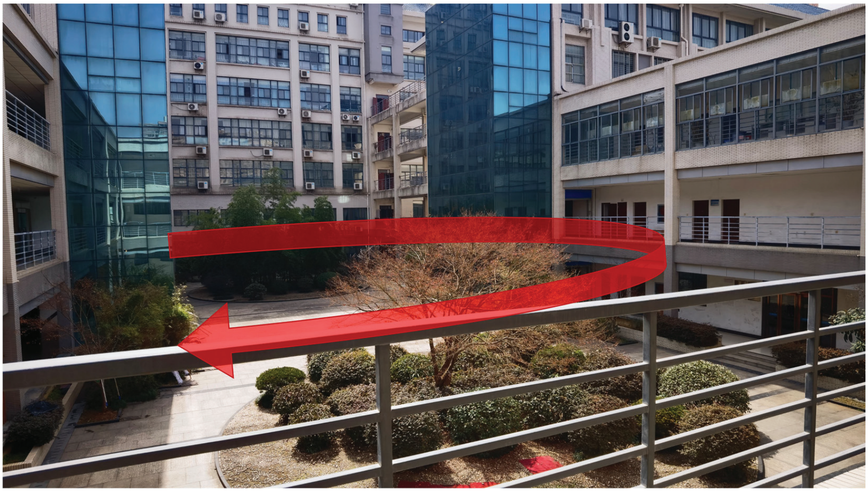


Figure 10. Indoor-outdoor environment: the system starts indoors and passes through a corridor surrounded by outdoors to reach the destination. The arrow direction represents the planned path direction.

Figure 11 displays the system's mapping results. The results show that LIO-FICP performs exceptionally well, successfully accomplishing 3D reconstruction of road sections and enabling the mapping and recognition of distant outdoor landscapes. The details of the outdoor parts, such as leaves and corridors, can be clearly seen in Figure 12.

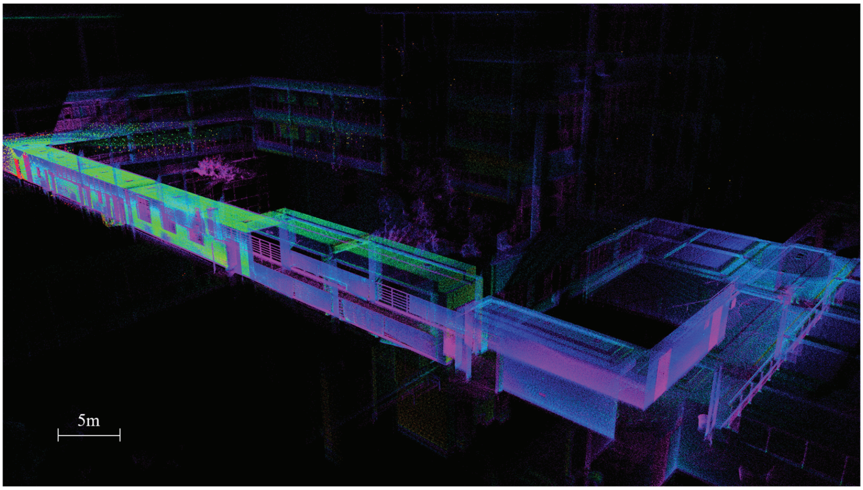
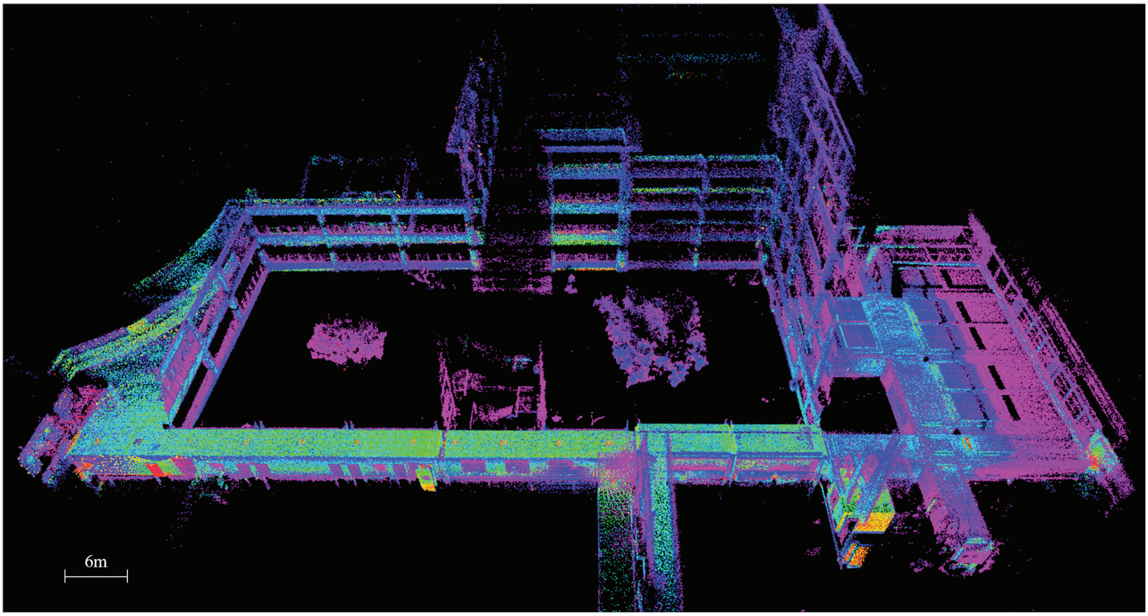


Figure 11. The mapping result of indoor-outdoor environment, the system starts indoors and passes through a corridor surrounded by outdoors to reach the destination.



(**a**)

Figure 12. *Cont.*

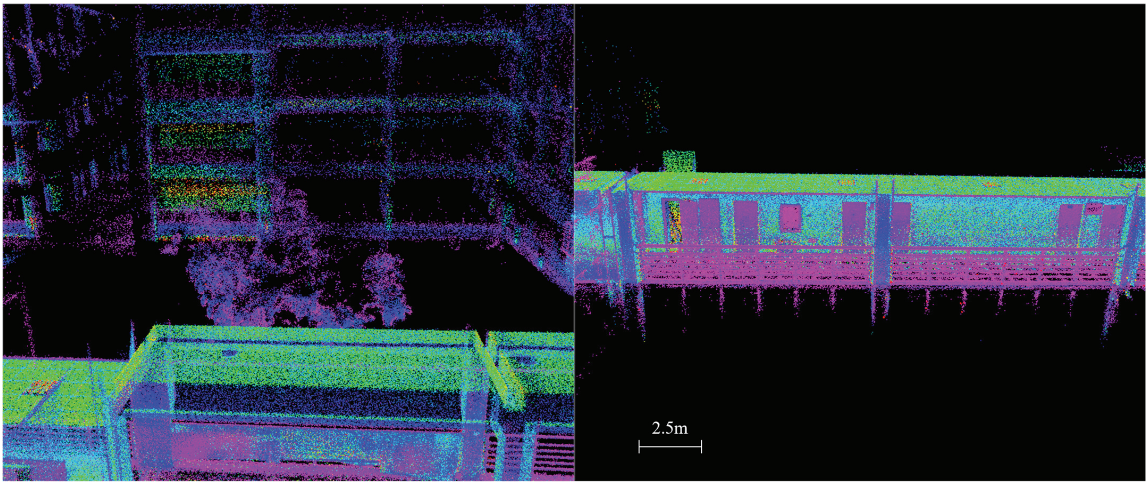


Figure 12. These are figures of the Indoor-outdoor environment. (**a**) The top-down point cloud map of the entire environment; (**b**) shows the details of tree reconstruction, with visible branches and leaves; (**c**) shows the reconstruction results of the corridor.

Figure 13 displays more detailed sections of the reconstructed map. Despite the brief period of point cloud accumulation, delicate structures like floor decorations and fire extinguisher cabinets are distinctly visible, demonstrating the SLAM system's high-precision mapping capability and robustness.

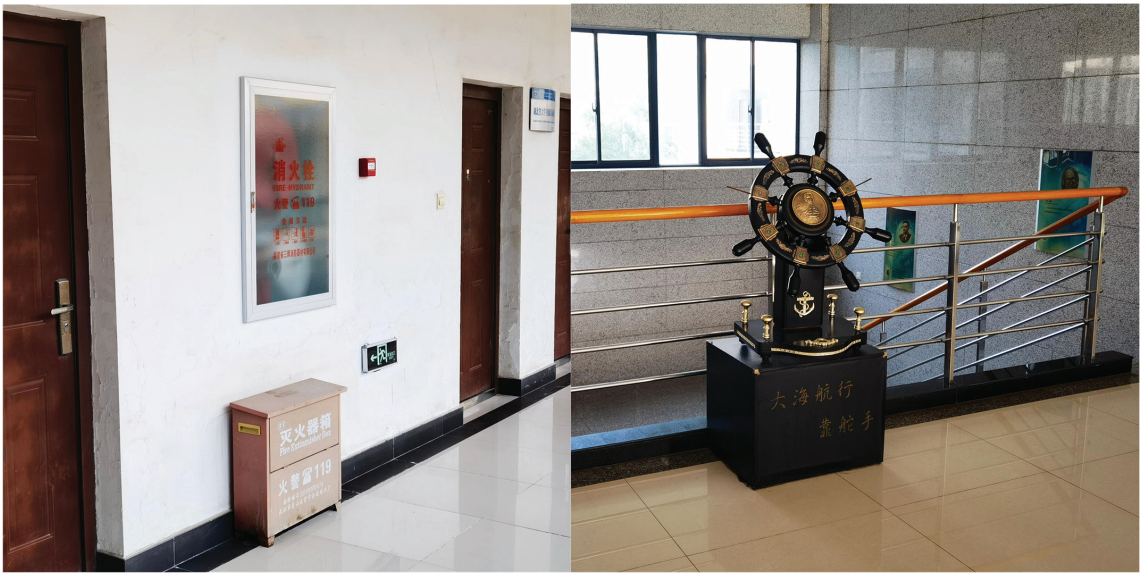


Figure 13. *Cont.*

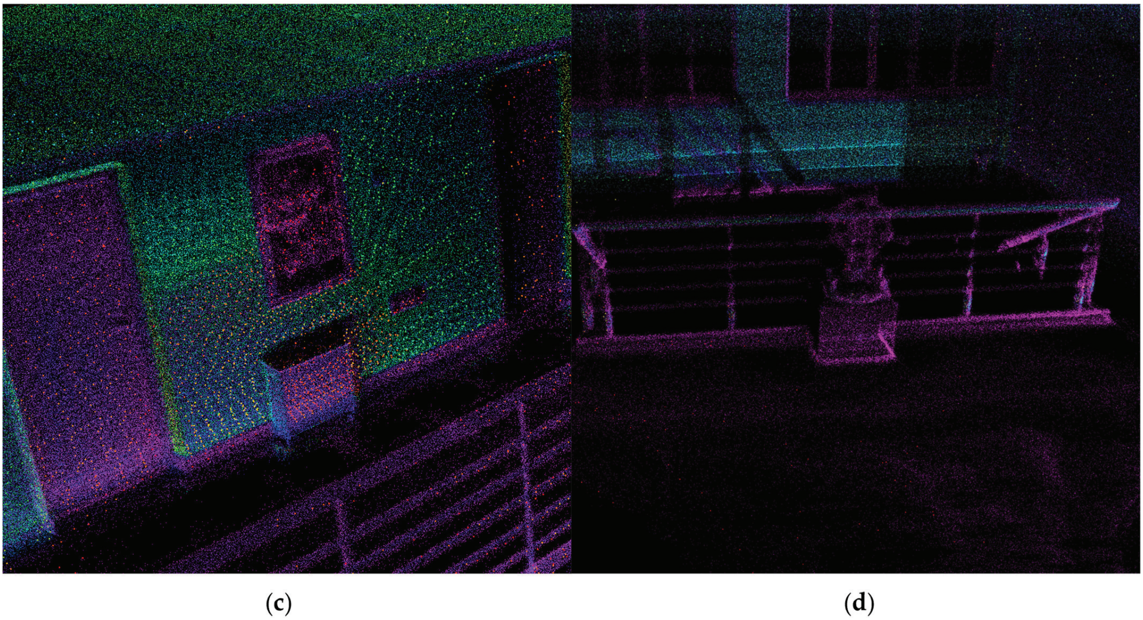


Figure 13. The reconstruction effect of the detail part. (**a**,**c**) Physical drawings and point clouds of fire extinguisher hydrants. (**b**,**d**) Physical drawings and point clouds of decorations.

The time taken by the vehicle from departure to the complete preservation of the 3D model was recorded and compared with the vehicle's movement time. Our calculations show that the system ceased reconstruction immediately after the vehicle stopped and exported the model file within 5 s, confirming the real-time performance of the SLAM system.

4.3. Auto-Exploration Experiment

After completing the localization and mapping experiments with our SLAM system, we deployed it for RRT Exploration within the same system. We continued to test the autonomous exploration capabilities of the SLAM system indoors. The overall 3D Auto-Exploration Using RRT framework was implemented in ROS and C++, under the same configuration. To more comprehensively assess our system's autonomous exploration abilities, we chose an indoor environment designed with a variety of obstacles, different terrains, and complex spatial layouts for experimentation.

We used the same indoor environment as in Section 4.2.3 to test it. At the start of the experiment, we manually placed the robot at the center of the map for initialization. After setting the boundaries of the map, the robot began its autonomous exploration. During the exploration process, the robot dynamically planned its path using the RRT algorithm to avoid obstacles and efficiently cover unknown areas. The 3D multi-goal RRT works as shown in Figure 14; the exploration path is generated on a 2D plane, although the car will eventually be reconstructed on a 3D map. We paid special attention to the robot's responsiveness to sudden obstacles and its navigation and obstacle avoidance efficiency in complex environments. Multiple experiments ensured the system's exploration was robust and continuous.

The experimental results demonstrated that our SLAM system, combined with the 3D RRT exploration framework, was capable of effectively carrying out autonomous exploration within complex indoor environments, while 2D RRT exploration always failed to explore this room. The robot was able to update the map information in real time and dynamically adjust its path based on newly discovered obstacles and unexplored areas.

When encountering unknown obstacles, the robot was able to react quickly and re-plan its path, showing good obstacle avoidance capabilities.

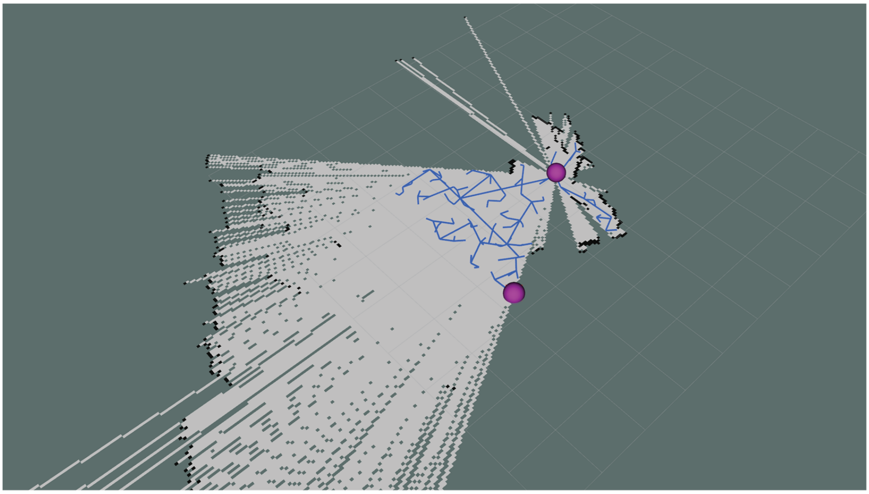


Figure 14. This is the process of multi-goal RRT growth. The parameter for cell size in the grid is 1, meaning that each grid cell represents a 1 m × 1 m area.

The 2D RRT exploration fails to perceive the dimensions of obstacles, resulting in incorrect path planning in front of spatially featured obstacles, and thus fails to explore a complete map (grid map, not a 3D point cloud), as shown in Figure 15. Additionally, traditional RRT exploration lacks specific trajectory tracking optimization, leading to motion deadlocks in some narrow areas from which it cannot escape, culminating in the failure of RRT Exploration in room exploration.

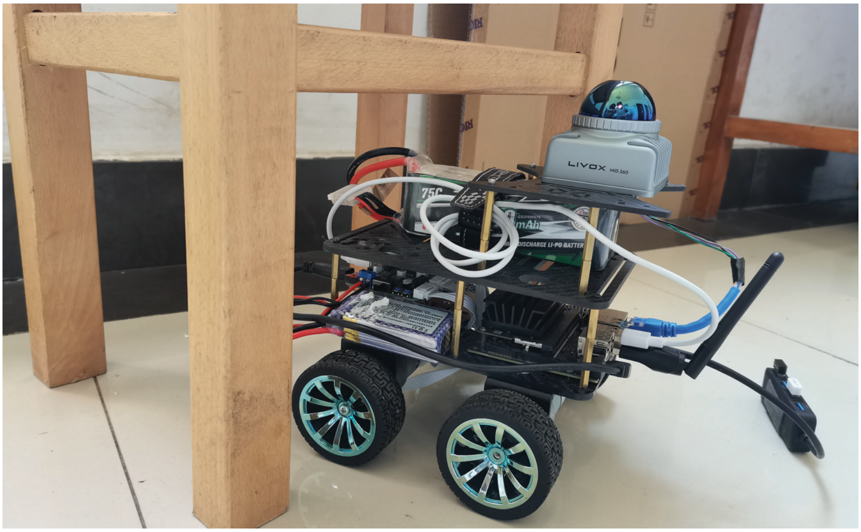


Figure 15. The illustration shows incorrect path planning by 2D RRT Exploration. We converted three-dimensional LiDAR point clouds into two-dimensional point clouds through filtering, which fails to detect obstacles above the space. This leads to erroneous path planning and can result in collisions or even overturning of the robot.

Figure 16 illustrates the maps explored by the robot after the experiment, indicating that the robot covered most of the unknown areas and the path selection was highly optimized. The experimental results demonstrated that our SLAM system, combined with the RRT exploration framework, was capable of effectively carrying out autonomous exploration within complex indoor environments. The robot was able to update the map information in real time and dynamically adjust its path based on newly discovered obstacles and unexplored areas. When encountering unknown obstacles, the robot was able to react quickly and re-plan its path, showing good obstacle avoidance capabilities.

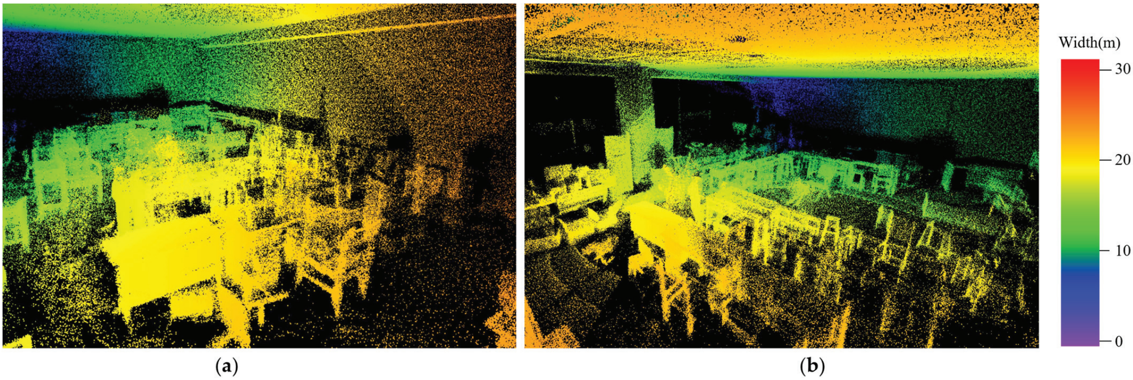


Figure 16. This is the result of the 3D multi-goal RRT exploration. (**a**,**b**) The reconstruction from different angles of the room.

4.4. Analysis of the Resource Occupancy

SAR and VMstat are two commonly used system performance monitoring tools in sysstat (https://github.com/sysstat/sysstat (accessed on 20 January 2024)) widely used in operating systems to monitor system performance and resource usage.

Since this SLAM system primarily utilizes CPU and memory resources, we employed the 'SAR' and 'VMstat' tools to monitor the CPU and memory usage in real time on our hardware platform, thus assessing the resource consumption of our algorithm. All tests were automatically conducted in a consistent environment using scripts and carried out on the 4.1 dataset. We also measured the resource utilization of SLAM systems based on fast-lio2 and lio-sam for comparison. The results are shown in Figure 17.

Data from Figure 17 and Table 8 demonstrate that our method, compared to FAST-LIO2, has lower CPU resource usage and memory consumption. Although our method consumes slightly more memory than LIO-SAM, it offers higher positioning accuracy, lower loop closure errors, improved stability and robustness, and a significant advantage in CPU utilization. This makes it particularly suitable for cost-conscious SLAM systems. Experiments show that in cost-sensitive application scenarios, our system can achieve efficient spatial localization and mapping with lower operational costs, not only meeting the performance requirements of low-cost SLAM systems but also providing significant advantages in sustainability and practicality, thus enabling a wide range of real-world applications.

Table 8. The average CPU and memory usage results of the SLAM system during testing.

Algorithm	The Average CPU Usage	The Average Memory Usage
LIO-SAM	66.75%	47.54%
FAST-LIO2	30.32%	56.73%
Ours	23.66%	52.45%

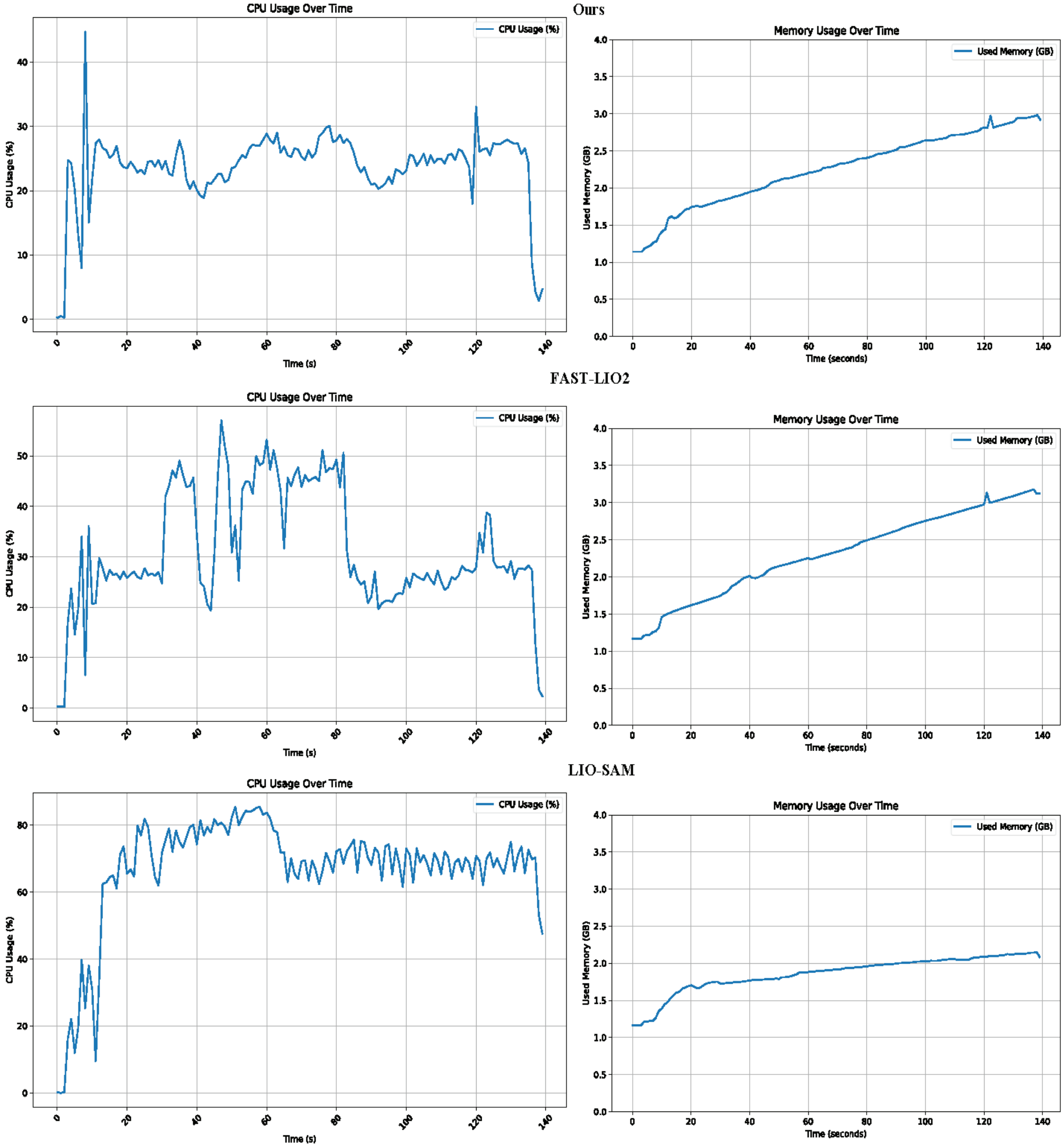


Figure 17. This is the results of system resource occupancy testing, conducted simultaneously with the experiments described in Section 4.2.3.

5. Discussion

We have designed a SLAM system that integrates LiDAR-Inertial Odometry with RRT Exploration. It is characterized by its low-cost yet efficient pose estimation, mapping, and autonomous 3D space exploration. The design encompasses three modules. To achieve high performance within budget constraints, we developed a stable and adaptable vehicle structure utilizing a four-wheel configuration, incorporating cost-effective sensors. Subsequently, we developed the SLAM algorithm based on the LiDAR-Inertial

Odometry framework. We achieved continuous, higher accuracy at a significantly lower computational load than traditional LOAM algorithms through precise optimization of point cloud matching and pose estimation with the Fast ICP algorithm. Finally, we adapted and enhanced the 3D RRT Exploration algorithm, equipping the system with advanced autonomous exploration functionalities. As UAVs advance in SLAM and three-dimensional reconstruction fields, this system has the potential to be applied in outdoor Unmanned Aircraft Systems (UAS) [56], offering prospective applications in data collection and mapping with small drones [57].

However, three limitations remain:

1. Although the LiDAR-Inertial Odometry framework enhances the efficiency and accuracy of pose estimation, it still heavily relies on the quality of data collected by sensors. Sensor performance degradation in harsh environments could directly affect the system's precision in localization and mapping.
2. While the 3D RRT Exploration algorithm grants the system high-performance autonomous exploration capabilities, its computational complexity substantially escalates in environments with dynamic obstacles, potentially diminishing exploration efficiency and prolonging reaction times. Additionally, we must conduct more experiments to adjust the NMPC weight parameters to make the system's motion control smoother and easier to port.
3. Generating 3D maps offline necessitates extra storage, and activating 3D autonomous exploration and Fast ICP optimization concurrently can result in elevated memory consumption.

6. Conclusions

This paper presents a low-cost SLAM system design that maintains high-performance SLAM algorithms under budget constraints. First, in terms of hardware, we designed a small vehicle with a three-layer structure and four rubber wheels, incorporating a control system that enhances the SLAM system's stability, adaptability, and load capacity, drawing inspiration from current mobile robot technologies. Secondly, we developed a foundational SLAM solution based on the LiDAR-Inertial Odometry and Fast ICP algorithm and melded it with a 3D multi-objective RRT Exploration strategy grounded in NMPC, not only adapts mainstream SLAM algorithms but also makes localized improvements for enhanced performance. Then, we designed the experiments that confirm our system's capability for accurate localization and mapping in intricate indoor settings, as well as autonomous reconstruction and surveying. Finally, we open-sourced our work to provide a low-cost SLAM solution with high performance. Although our approach strikes a good balance between cost and performance, it still encounters some constraints that must be addressed in future works focused on developing cost-effective SLAM systems.

Author Contributions: Conceptualization, X.H. and L.Z.; Data curation, X.H.; Formal analysis, C.P., L.Z. and X.H.; Funding acquisition, X.H.; Methodology, C.P.; Writing—original draft, C.P.; writing—review & editing, C.P., L.Z. and X.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Key R&D Program of China (No.2023YFC3209105) and the Fundamental Research Funds for the Central Universities (2042024kf0035).

Data Availability Statement: Code and datasets have been made available at: https://github.com/pcl5/Low-cost-system (accessed on 13 March 2024).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Smith, R.C.; Cheeseman, P. On the representation and estimation of spatial uncertainty. *Int. J. Robot. Res.* **1986**, *5*, 56–68. [CrossRef]
2. Durrant-Whyte, H.; Bailey, T. Simultaneous localization and mapping: Part I. *IEEE Robot. Autom. Mag.* **2006**, *13*, 99–110. [CrossRef]
3. Grigorescu, S.; Trasnea, B.; Cocias, T.; Macesanu, G. A survey of deep learning techniques for autonomous driving. *J. Field Robot.* **2020**, *37*, 362–386. [CrossRef]
4. Takleh, T.T.O.; Bakar, N.A.; Rahman, S.A.; Hamzah, R.; Aziz, Z. A brief survey on SLAM methods in autonomous vehicle. *Int. J. Eng. Technol.* **2018**, *7*, 38–43. [CrossRef]

5. Singandhupe, A.; La, H.M. A review of slam techniques and security in autonomous driving. In Proceedings of the 2019 Third IEEE International Conference on Robotic Computing (IRC), Naples, Italy, 25–27 February 2019; pp. 602–607.
6. Polvi, J.; Taketomi, T.; Yamamoto, G.; Dey, A.; Sandor, C.; Kato, H. SlidAR: A 3D positioning method for SLAM-based handheld augmented reality. *Comput. Graph.* **2016**, *55*, 33–43. [CrossRef]
7. Liu, H.; Zhang, G.; Bao, H. Robust keyframe-based monocular SLAM for augmented reality. In Proceedings of the 2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Merida, Mexico, 19–23 September 2016; pp. 1–10.
8. Chen, C.-W.; Chen, W.-Z.; Peng, J.-W.; Cheng, B.-X.; Pan, T.-Y.; Kuo, H.-C.; Hu, M.-C. A real-time markerless augmented reality framework based on SLAM technique. In Proceedings of the 2017 14th International Symposium on Pervasive Systems, Algorithms and Networks & 2017 11th International Conference on Frontier of Computer Science and Technology & 2017 Third International Symposium of Creative Computing (ISPAN-FCST-ISCC), Exeter, UK, 21–23 June 2017; pp. 127–132.
9. Dissanayake, M.G.; Newman, P.; Clark, S.; Durrant-Whyte, H.F.; Csorba, M. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Trans. Robot. Autom.* **2001**, *17*, 229–241. [CrossRef]
10. Montemerlo, M.; Thrun, S.; Koller, D.; Wegbreit, B. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In Proceedings of the Eighteenth National Conference on Artificial Intelligence, Edmonton, AB, Canada, 28 July 2002–1 August 2002; pp. 593–598.
11. Thrun, S.; Burgard, W.; Fox, D. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In Proceedings of the 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065), San Francisco, CA, USA, 24–28 April 2000; pp. 321–328.
12. Davison, A.J.; Reid, I.D.; Molton, N.D.; Stasse, O. MonoSLAM: Real-time single camera SLAM. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 1052–1067. [CrossRef] [PubMed]
13. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [CrossRef]
14. Engel, J.; Schöps, T.; Cremers, D. LSD-SLAM: Large-scale direct monocular SLAM. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 834–849.
15. Mur-Artal, R.; Tardós, J.D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [CrossRef]
16. Campos, C.; Elvira, R.; Rodríguez, J.J.G.; Montiel, J.M.; Tardós, J.D. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. Robot.* **2021**, *37*, 1874–1890. [CrossRef]
17. Chen, W.; Shang, G.; Ji, A.; Zhou, C.; Wang, X.; Xu, C.; Li, Z.; Hu, K. An overview on visual slam: From tradition to semantic. *Remote Sens.* **2022**, *14*, 3010. [CrossRef]
18. Wang, D.; Watkins, C.; Xie, H. MEMS mirrors for LiDAR: A review. *Micromachines* **2020**, *11*, 456. [CrossRef] [PubMed]
19. Liu, Z.; Zhang, F.; Hong, X. Low-cost retina-like robotic lidars based on incommensurable scanning. *IEEE ASME Trans. Mechatron.* **2021**, *27*, 58–68. [CrossRef]
20. Ye, H.; Chen, Y.; Liu, M. Tightly coupled 3d lidar inertial odometry and mapping. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 3144–3150.
21. Almadhoun, R.; Taha, T.; Seneviratne, L.; Dias, J.; Cai, G. A survey on inspecting structures using robotic systems. *Int. J. Adv. Robot. Syst.* **2016**, *13*, 1729881416663664. [CrossRef]
22. Yamauchi, B. Frontier-based exploration using multiple robots. In Proceedings of the Second International Conference on Autonomous Agents, Minneapolis, MN, USA, 10–13 May 1998; pp. 47–53.
23. Chickering, D.M. Optimal structure identification with greedy search. *J. Mach. Learn. Res.* **2002**, *3*, 507–554.
24. Mostegel, C.; Wendel, A.; Bischof, H. Active monocular localization: Towards autonomous monocular exploration for multirotor mavs. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 3848–3855.
25. Placed, J.A.; Castellanos, J.A. A deep reinforcement learning approach for active SLAM. *Appl. Sci.* **2020**, *10*, 8386. [CrossRef]
26. LaValle, S.M. *Rapidly-Exploring Random Trees: A New Tool for Path Planning*; TR 98-11; Department of Computer Science, Iowa State University: Ames, IA, USA, 1998.
27. Grisetti, G.; Stachniss, C.; Burgard, W. Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. In Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 18–22 April 2005; pp. 2432–2437.
28. Zhang, J.; Singh, S. LOAM: Lidar odometry and mapping in real-time. In Proceedings of the Robotics: Science and Systems, Berkeley, CA, USA, 12–16 July 2014; pp. 1–9.
29. Shan, T.; Englot, B. Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 4758–4765.
30. Lin, J.; Zhang, F. Loam livox: A fast, robust, high-precision LiDAR odometry and mapping package for LiDARs of small FoV. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 3126–3131.
31. Xu, W.; Zhang, F. Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter. *IEEE Robot. Autom. Lett.* **2021**, *6*, 3317–3324. [CrossRef]

32. Hess, W.; Kohler, D.; Rapp, H.; Andor, D. Real-time loop closure in 2D LIDAR SLAM. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1271–1278.
33. Xu, W.; Cai, Y.; He, D.; Lin, J.; Zhang, F. Fast-lio2: Fast direct lidar-inertial odometry. *IEEE Trans. Robot.* **2022**, *38*, 2053–2073. [CrossRef]
34. Shan, T.; Englot, B.; Meyers, D.; Wang, W.; Ratti, C.; Rus, D. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October 2020–24 January 2021; pp. 5135–5142.
35. Qin, C.; Ye, H.; Pranata, C.E.; Han, J.; Zhang, S.; Liu, M. Lins: A lidar-inertial state estimator for robust and efficient navigation. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 8899–8906.
36. Besl, P.J.; McKay, N.D. Method for registration of 3-D shapes. In Proceedings of the Sensor Fusion IV: Control Paradigms and Data Structures, Boston, MA, USA, 12–15 November 1991; pp. 586–606.
37. Rusinkiewicz, S.; Levoy, M. Efficient variants of the ICP algorithm. In Proceedings of the Proceedings Third International Conference on 3-D Digital Imaging and Modeling, Quebec City, QC, Canada, 28 May–1 June 2001; pp. 145–152.
38. Pomerleau, F.; Colas, F.; Siegwart, R. A review of point cloud registration algorithms for mobile robotics. *Found. Trends Robot.* **2015**, *4*, 1–104. [CrossRef]
39. Clotet, E.; Palacín, J. Slamicp library: Accelerating obstacle detection in mobile robot navigation via outlier monitoring following icp localization. *Sensors* **2023**, *23*, 6841. [CrossRef]
40. Grisetti, G.; Kümmerle, R.; Stachniss, C.; Burgard, W. A tutorial on graph-based SLAM. *IEEE Intell. Transp. Syst. Mag.* **2010**, *2*, 31–43. [CrossRef]
41. Ristic, B.; Palmer, J.L. Autonomous exploration and mapping with RFS occupancy-grid SLAM. *Entropy* **2018**, *20*, 456. [CrossRef] [PubMed]
42. Bundy, A.; Wallen, L. Breadth-first search. In *Catalogue of Artificial Intelligence Tools*; Springer: Berlin/Heidelberg, Germany, 1984; p. 13.
43. Umari, H.; Mukhopadhyay, S. Autonomous robotic exploration based on multiple rapidly-exploring randomized trees. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 1396–1402.
44. Mukhopadhyay, S.; Umari, H.; Koirala, K. Multi-robot Map Exploration Based on Multiple Rapidly-Exploring Randomized Trees. *SN Comput. Sci.* **2023**, *5*, 31. [CrossRef]
45. Wu, Z.; Meng, Z.; Zhao, W.; Wu, Z. Fast-RRT: A RRT-based optimal path finding method. *Appl. Sci.* **2021**, *11*, 11777. [CrossRef]
46. Kuffner, J.J.; LaValle, S.M. RRT-connect: An efficient approach to single-query path planning. In Proceedings of the 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065), San Francisco, CA, USA, 24–28 April 2000; pp. 995–1001.
47. Cao, C.; Zhu, H.; Choset, H.; Zhang, J. TARE: A Hierarchical Framework for Efficiently Exploring Complex 3D Environments. In Proceedings of the Robotics: Science and Systems, Virtually, 12–16 July 2021; p. 2.
48. Huston, J.C.; Graves, B.J.; Johnson, D.B. Three wheeled vehicle dynamics. *SAE Trans.* **1982**, *91*, 591–604.
49. Gfrerrer, A. Geometry and kinematics of the Mecanum wheel. *Comput. Aided Geom. Des.* **2008**, *25*, 784–791. [CrossRef]
50. Raitoharju, M.; Piché, R. On computational complexity reduction methods for Kalman filter extensions. *IEEE Aerosp. Electron. Syst. Mag.* **2019**, *34*, 2–19. [CrossRef]
51. Bentley, J.L. Multidimensional binary search trees used for associative searching. *Commun. ACM* **1975**, *18*, 509–517. [CrossRef]
52. Mayne, D.Q.; Rawlings, J.B.; Rao, C.V.; Scokaert, P.O. Constrained model predictive control: Stability and optimality. *Automatica* **2000**, *36*, 789–814. [CrossRef]
53. Lindqvist, B.; Agha-Mohammadi, A.-A.; Nikolakopoulos, G. Exploration-RRT: A multi-objective path planning and exploration framework for unknown and unstructured environments. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 3429–3435.
54. Ma, L.; Xue, J.; Kawabata, K.; Zhu, J.; Ma, C.; Zheng, N. Efficient sampling-based motion planning for on-road autonomous driving. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 1961–1976. [CrossRef]
55. Rusu, R.B.; Cousins, S. 3d is here: Point cloud library (pcl). In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 1–4.
56. Zhang, S.; Bogus, S.M.; Lippitt, C.D.; Kamat, V.; Lee, S. Implementing remote-sensing methodologies for construction research: An unoccupied airborne system perspective. *J. Constr. Eng. Manag.* **2022**, *148*, 03122005. [CrossRef]
57. Lippitt, C.D.; Zhang, S. The impact of small unmanned airborne platforms on passive optical remote sensing: A conceptual perspective. *Int. J. Remote Sens.* **2018**, *39*, 4852–4868. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

Regional Real-Time between-Satellite Single-Differenced Ionospheric Model Establishing by Multi-GNSS Single-Frequency Observations: Performance Evaluation and PPP Augmentation

Ahao Wang [1], Yize Zhang [2,*], Junping Chen [2,3], Xuexi Liu [4] and Hu Wang [5]

[1] College of Geoscience and Surveying Engineering, China University of Mining and Technology-Beijing, Beijing 100083, China; ahao_wang@cumtb.edu.cn
[2] Shanghai Astronomical Observatory, Chinese Academy of Sciences, Shanghai 200030, China; junping@shao.ac.cn
[3] School of Astronomy and Space Science, University of Chinese Academy of Sciences, Beijing 100049, China
[4] School of Environment Science and Spatial Informatics, China University of Mining and Technology, Xuzhou 221116, China; xuexiliu@cumt.edu.cn
[5] Chinese Academy of Surveying & Mapping, Beijing 100036, China; wanghu@casm.ac.cn
* Correspondence: zhyize@shao.ac.cn

Abstract: The multi-global navigation satellite system (GNSS) undifferenced and uncombined precise point positioning (UU-PPP), as a high-precision ionospheric observables extraction technology superior to the traditional carrier-to-code leveling (CCL) method, has received increasing attention. In previous research, only dual-frequency (DF) or multi-frequency (MF) observations are used to extract slant ionospheric delay with the UU-PPP. To reduce the cost of ionospheric modeling, the feasibility of extracting ionospheric observables from the multi-GNSS single-frequency (SF) UU-PPP was investigated in this study. Meanwhile, the between-satellite single-differenced (SD) method was applied to remove the effects of the receiver differential code bias (DCB) with short-term time-varying characteristics in regional ionospheric modeling. In the assessment of the regional real-time (RT) between-satellite SD ionospheric model, the internal accord accuracy of the SD ionospheric delay can be better than 0.5 TECU, and its external accord accuracy within 1.0 TECU is significantly superior to three global RT ionospheric models. With the introduction of the proposed SD ionospheric model into the multi-GNSS kinematic RT SF-PPP, the initialization speed of vertical positioning errors can be improved by 21.3% in comparison with the GRAPHIC (GRoup And PHase Ionospheric Correction) SF-PPP model. After reinitialization, both horizontal and vertical positioning errors of the SD ionospheric constrained (IC) SF-PPP can be maintained within 0.2 m. This proves that the proposed SDIC SF-PPP model can enhance the continuity and stability of kinematic positioning in the case of some GNSS signals missing or blocked. Compared with the GRAPHIC SF-PPP, the horizontal positioning accuracy of the SDIC SF-PPP in kinematic mode can be improved by 37.9%, but its vertical positioning accuracy may be decreased. Overall, the 3D positioning accuracy of the SD ionospheric-constrained RT SF-PPP can be better than 0.3 m.

Keywords: multi-global navigation satellite system (multi-GNSS); precise point positioning (PPP); ionospheric modeling; real-time (RT); single frequency (SF); between-satellite single-differenced

Citation: Wang, A.; Zhang, Y.; Chen, J.; Liu, X.; Wang, H. Regional Real-Time between-Satellite Single-Differenced Ionospheric Model Establishing by Multi-GNSS Single-Frequency Observations: Performance Evaluation and PPP Augmentation. *Remote Sens.* **2024**, *16*, 1511. https://doi.org/10.3390/rs16091511

Academic Editor: Michael E. Gorbunov

Received: 14 March 2024
Revised: 11 April 2024
Accepted: 22 April 2024
Published: 25 April 2024

1. Introduction

As one of the most serious interference factors in radio signals, the ionospheric delay has a significant negative impact on the global navigation satellite system (GNSS) data processing, especially in the precise point positioning (PPP) domain [1,2]. If GNSS users can afford expensive dual-frequency (DF) receivers, ionospheric errors can be eliminated by over 99% by forming a DF ionosphere-free (IF) combination model. However, the majority of smart devices, such as mobile phones and wristwatches, can only be equipped with single-frequency (SF) GNSS chipsets, making it impossible to adopt the IF model to weaken ionospheric errors [3,4]. To achieve low-cost and high-precision SF positioning, many kinds of broadcast or post-processing ionospheric models have been proposed so far. Although the GPS Klobuchar model, BeiDou global broadcast ionospheric delay correction model (BDGIM), and Galileo NeQuick model can be applied to real-time (RT) SF positioning, their ionospheric correction capabilities are limited and no more than 80% [5–8]. The post-processed global ionospheric map (GIM), as one of the most accurate ionospheric models, can provide correction accuracy within two total electron content units (TECU), but it cannot support RT positioning [9]. With the development of the Real-Time Working Group (RTWG) of the International GNSS Service (IGS), an RT-GIM has been provided by some IGS real-time ionosphere centers. The accuracy of RT-GIM is slightly lower than that of post-processed GIM and can reach around 3 TECU [10]. Due to the limited accuracy of the current RT ionospheric models, it is necessary to establish an RT ionospheric model with cm-level accuracy using regional reference networks to improve the performance of RT SF-PPP.

On the premise of not changing the number of regional monitoring stations, there are two main factors that affect the quality of ionospheric modeling. One is the extraction accuracy of ionospheric delay observables, and another is the calculation accuracy of the differential code bias (DCB) for satellite and receiver. The traditional carrier-to-code leveling (CCL) method has been widely used in ionospheric modeling due to its simple structure and high computational efficiency. Nevertheless, the accuracy of ionospheric delay extracted by this method is limited and is susceptible to adverse effects of multipath errors and code noises [11,12]. Thanks to the preservation of ionospheric parameters in the undifferenced and uncombined (UU) PPP, a novel method based on carrier phase observations for extracting ionospheric delays has been proposed [13]. Compared with the CCL method, the accuracy of slant ionospheric delays derived from the UU-PPP can be improved at least three times to 0.1 TECU. As expected, the modeling accuracy of regional ionospheric vertical total electron content (VTEC) using the UU-PPP method is better than that using the CCL method [14].

Due to the presence of hardware bias in retrieved ionospheric observables, it is necessary to simultaneously estimate both satellite and receiver DCB parameters when modeling ionospheric VTEC. The satellite DCB has great long-term stability, so its solution accuracy is high and reliable [15]. Unfortunately, the receiver DCB is easily influenced by various factors, such as ambient temperature and hardware alternation. Thus, its apparent short-term time-varying characteristics can be observed [16,17]. If the receiver DCB as an additional parameter was estimated in UU-PPP, although pure slant ionospheric delays can be obtained [18], the increase in estimated parameters reduces the strength and computational efficiency of the UU-PPP model. Meanwhile, the high-frequency parameterization of receiver DCB will inevitably increase the burden and cost of data transmission, making it difficult to apply to RT ionospheric modeling based on dense reference networks.

To completely remove the adverse effects of receiver DCB and meet the requirements of time-critical in RT ionospheric modeling, a regional RT ionospheric model was established using the classical between-satellite single-differenced (SD) method [19]. Since the receiver DCB can be precisely eliminated and high-precision ionospheric observables are derived from the UU-PPP, the accuracy of the proposed RT ionospheric model is significantly better than that of the post-processing GIM model. However, the implementation of this high-quality ionospheric model must be based on the use of GNSS DF observations. Thus, the hardware cost of the modeling system is very expensive. Considering that the multi-GNSS SF UU-PPP has the ability to achieve cm-level accuracy at present [20,21], in our study, the possibility of modeling a regional RT between-satellite SD ionospheric model based on SF observations was explored and verified. This is extremely important for low-cost SF positioning users. Different from the literature [19] only using GPS and Galileo observations, the BDS-3 observations were introduced into multi-GNSS processing in this contribution, which can improve the performance of the regional RT between-satellite SD ionospheric model by increasing the spatial resolution of the ionosphere pierce point (IPP). In summary, this contribution is to attempt to reduce the hardware cost of ionospheric modeling and provide an effective solution for building high-precision regional ionospheric models only using affordable SF devices or chips in the future.

The writing structure of this paper is as follows: first, the extraction method of slant ionospheric observables using multi-GNSS SF UU-PPP technology, modeling algorithm of the regional RT between-satellite SD ionospheric delay, and ionospheric-constrained multi-GNSS SF-PPP model are introduced in detail. Then, the experimental data for the European region and processing strategies in both ionospheric modeling and positioning domains are presented. After evaluating the performance of the regional RT between-satellite SD ionospheric model, its contribution to the multi-GNSS RT SF-PPP is analyzed. In the final section, some new findings and conclusions are summarized.

2. Methods

Thanks to the advantages of abundant satellite resources and high sampling rates for the GNSS technology, ionospheric observables with low-cost and high spatial-temporal resolution can be obtained from the multi-GNSS SF UU-PPP. The modeling method of the regional RT between-satellite SD ionospheric delay is described in this section. To optimize the RT kinematic positioning performance of SF users, a novel multi-GNSS RT SF-PPP model based on the SD ionospheric constraints is proposed.

2.1. Extraction of Slant Ionospheric Observables from Single-Frequency UU-PPP

The GNSS raw code $P_{r,1}^s$ and phase $L_{r,1}^s$ observations at the first frequency can be expressed as [14]

$$\begin{cases} P_{r,1}^s = \rho_r^s + c \cdot (dt_r - dt^s) + T_r^s + I_1^s + B_{r,1} - B_1^s + \varepsilon_{p_1} \\ L_{r,1}^s = \rho_r^s + c \cdot (dt_r - dt^s) + T_r^s - I_1^s + \omega_1^s + N_{r,1}^s + b_{r,1} - b_1^s + \varepsilon_{L_1} \end{cases} \quad (1)$$

where the superscript s and subscript r denote the satellite and GNSS receiver, respectively. ρ_k^s is the calculated distance between the satellite and the GNSS receiver. c denotes the speed of light. dt_r and dt^s denote the clock errors of the satellite and GNSS receiver, respectively. T_r^s denotes the slant tropospheric delay errors, I_1^s denotes the slant ionospheric errors. ω_1^s denotes the carrier phase wind-up errors. $N_{r,1}^s$ denotes the integer ambiguity of carrier phase. $B_{r,1}$ denotes the code hardware delays of the GNSS receiver, and B_1^s denotes the code hardware delays of the satellite. $b_{r,1}$ denotes the phase hardware delays of the GNSS receiver, and b_1^s denotes the phase hardware delays of the satellite. ε_{p_1} and ε_{L_1} denote the code and phase observation noises, respectively.

In RT SF-PPP, the satellite and receiver clock errors need to be corrected using code hardware delays because the RT precise satellite clocks provided by the IGS real-time service (RTS) are obtained from the DF IF observations. Hence, the reparametrized clock errors of the satellite $d\overline{t^s}$ and GNSS receiver $d\bar{t}_r$ at the first frequency can be expressed as

$$\begin{cases} d\overline{t^s} = dt^s + \frac{d_{IF}^s}{c} \\ d\overline{t_r} = dt_r + \frac{d_{r,IF}}{c} \end{cases} \quad (2)$$

with

$$\begin{cases} d_{IF}^s = \frac{(f_1)^2 \cdot B_1^s - (f_i)^2 \cdot B_i^s}{(f_1)^2 - (f_i)^2} \\ d_{r,IF} = \frac{(f_1)^2 \cdot B_{r,1} - (f_i)^2 \cdot B_{r,i}}{(f_1)^2 - (f_i)^2} \end{cases} \quad (3)$$

where d_{IF}^s and $d_{r,IF}$ are the IF code hardware delay of the satellite and GNSS receiver, respectively. f is the frequency value and i is the i-th frequency. It should be noted that $i = 2$ for GPS, GLONASS and Galileo satellites, while for BDS satellites, i= 3. The code hardware delay at a specific frequency is usually not directly obtainable, and only DCB values can be used as follows:

$$\begin{cases} DCB^s = B_1^s - B_i^s \\ DCB_r = B_{r,1} - B_{r,i} \end{cases} \tag{4}$$

When substituting Equations (2)–(4) into Equation (1), the new code and phase equations of RT SF-PPP can be expressed as

$$\begin{cases} P_{r,1}^s = \rho_r^s + c \cdot (d\overline{t_r} - d\overline{t^s}) + T_r^s + I_1^s - \frac{(f_i)^2}{(f_1)^2 - (f_i)^2}(DCB_r - DCB^s) + \varepsilon_{p_1} \\ L_{r,1}^s = \rho_r^s + c \cdot (d\overline{t_r} - d\overline{t^s}) + T_r^s - I_1^s + N_{r,1}^s + d_{r,IF} - d_{IF}^s + b_{r,1} - b_1^s + \varepsilon_{L_1} \end{cases} \tag{5}$$

Due to the linear correlation between hardware delays and estimated parameters such as ionosphere and ambiguity, the Equation (5) is rank deficient and multiple unknown parameters cannot be solved at the same time. The reparametrized ionosphere $\overline{I_1^s}$ and ambiguity $\overline{N_{r,1}^s}$ parameters can be expressed as

$$\begin{cases} \overline{I_1^s} = I_1^s - \frac{(f_i)^2}{(f_1)^2 - (f_i)^2}(DCB_r - DCB^s) \\ \overline{N_{r,1}^s} = N_{r,1}^s + d_{r,IF} - d_{IF}^s + b_{r,1} - b_1^s - \frac{(f_i)^2}{(f_1)^2 - (f_i)^2}(DCB_r - DCB^s) \end{cases} \tag{6}$$

Equation (6) is substituted into Equation (5) and linearize it. Both satellite positions and clock errors can be corrected using the RT precise products provided by IGS RTS [22]. The dry part of the tropospheric delay is generally corrected using an empirical model, and its wet part as unknowns to estimate. As for ionospheric delay, the GPS Klobuchar model is first used to correct it in real time, and then its residual parts can be estimated as parameter. Hence, the $\overline{I_1^s}$ represents the sum of ionospheric model values and residual values, which are the extracted slant ionospheric observables. To sum up, the final GNSS undifferenced and uncombined observation equations can be expressed as

$$\begin{cases} P_{r,1}^s = \mathbf{e}_r^s \cdot \mathbf{g} + c \cdot d\overline{t_r} + M \cdot T_{ZWD} + \overline{I_1^s} + \varepsilon_{p_1} \\ L_{r,1}^s = \mathbf{e}_r^s \cdot \mathbf{g} + c \cdot d\overline{t_r} + M \cdot T_{ZWD} - \overline{I_1^s} + \overline{N_{r,1}^s} + \varepsilon_{L_1} \end{cases} \tag{7}$$

where $\mathbf{e}_r^s$ is the unit vector of the range between the satellite and GNSS receiver. $\mathbf{g}$ is the vector of the three- dimensional (3D) position errors. T_{ZWD} denotes the wet part of tropospheric delay in zenith direction. M denotes the mapping function for tropospheric delay. The final parameters E that need to be estimated for SF UU-PPP can be summarized as

$$E = [\mathbf{g}, d\overline{t_r}, T_{ZWD}, \overline{I_1^s}, \overline{N_{r,1}^s}] \tag{8}$$

2.2. Modeling Algorithm of the Regional Real-Time between-Satellite Single-Differenced Ionospheric Delay

In the real-time extraction process of slant ionospheric delay, to improve the accuracy of ionospheric observables, the 3D coordinates of the monitoring stations should be fixed precisely by using the Solution-Independent Exchange (SINEX) data. The ionospheric delay in the line-of-sight direction can be expressed as [12]

$$I_1^s = \frac{40.3 \times 10^{16}}{(f_1)^2} \cdot \nabla^s \cdot VTEC^s \tag{9}$$

with

$$\begin{cases} \nabla^s = 1/\cos(\mu) \\ \mu = \arcsin(\frac{R_E \cdot \sin(\alpha \cdot Z)}{R_E + H_{iono}}) \end{cases} \tag{10}$$

where ∇^s is the mapping function used to convert ionospheric VTEC to slant TEC (STEC). $VTEC^s$ is the ionospheric vertical TEC. R_E is the mean radius of the earth, which can be set to 6371 km. α is an empirical constant that can be set to 0.9782. Z is the zenith distance at the GNSS receiver. H_{iono} is the assumed height of the single-layer spherical shell, which is set to 450 km in this contribution.

Substituting Equation (9) into Equation (6), the extracted ionospheric observables $\overline{I_1^s}$ can be re-written as

$$\overline{I_1^s} = \frac{40.3 \times 10^{16}}{(f_1)^2} \cdot \nabla^s \cdot VTEC^s - \frac{(f_i)^2}{(f_1)^2 - (f_i)^2}(DCB_r - DCB^s) \tag{11}$$

Considering the great long-term stability of satellite DCB within several days, its estimation accuracy can be better than 0.1 ns [15]. The multi-GNSS experiment (MGEX) final DCB products are generally used to correct the satellite DCB DCB^s in Equation (11). The receiver DCB DCB_r can be removed using the between-satellite SD algorithm. Thus, we can obtain the SD ionospheric delay $\Delta\overline{I_1^s}$ as follows:

$$\Delta\overline{I_1^s} = \overline{I_1^s} - \overline{I_1^{ref}} = \frac{40.3 \times 10^{16}}{(f_1)^2}(\nabla^s \cdot VTEC^s - \nabla^{ref} \cdot VTEC^{ref}) \tag{12}$$

where ref represents the reference satellite in the between-satellite SD algorithm, which selects the satellite with the highest elevation from a constellation at each epoch.

To meet the time-critical requirements of RT ionospheric modeling, the polynomial function with simple structure and high computational efficiency was adopted to model the regional RT between-satellite SD ionospheric delay as follows:

$$\begin{aligned} \Delta\overline{I_1^s} = & \frac{40.3\times 10^{16}}{(f_1)^2}\nabla^s \cdot \left[\sum_{i=0}^{n}\sum_{j=0}^{m} E_{ij}(\varphi^s - \varphi_0)^i(\theta^s - \theta_0)^j\right] \\ & - \frac{40.3\times 10^{16}}{(f_1)^2}\nabla^{ref} \cdot \left[\sum_{i=0}^{n}\sum_{j=0}^{m} E_{ij}(\varphi^{ref} - \varphi_0)^i(\theta^{ref} - \theta_0)^j\right] \end{aligned} \tag{13}$$

with

$$\theta - \theta_0 = \frac{(\lambda - \lambda_0)}{15} + (t - t_0) \tag{14}$$

where n and m are the orders of the polynomials, both of which are set to 2 in this study. E_{ij} is the coefficient that needs to be estimated, with a number of $3 \times 3 = 9$. φ and φ_0 denote the geodetic latitude for the IPP and regional center point, respectively. θ denotes the solar hour angle at the observation time t. θ_0 denotes the reference time of ionospheric modeling t_0. λ and λ_0 denote the geodetic longitude for the IPP and regional center point, respectively.

In this contribution, the observation window of the regional RT between-satellite SD ionospheric modeling was set to 20 min. The reference time was selected as the middle time of this sliding window. The nine estimated coefficients are fitted by the observations with the interval of 30 s collected from the past 20 min, and this sliding window of modeling moves forward for 10 min each time. Thus, all coefficients are updated every 10 min and broadcast to positioning users in real-time.

2.3. Multi-GNSS Real-Time Single-Frequency PPP Enhanced by Regional between-Satellite Single-Differenced Ionospheric Model

When the GNSS SF users receive the fitting coefficients of the regional RT between-satellite SD ionospheric model, the SD ionospheric delay of each available satellite can be calculated in real-time. Adding it as a virtual observation to the RT SF UU-PPP model, the fast and precise solution of positioning parameters can be achieved. The regional between-satellite SD ionospheric-constrained multi-GNSS RT SF-PPP model can be expressed as

$$\begin{cases} P_{r,1}^{G} = \mathbf{e}_r^{G} \cdot \mathbf{g} + c \cdot \overline{dt_r} + M \cdot T_{ZWD} + \overline{I_1^{G}} + \varepsilon_{p_1^{G}} \\ P_{r,1}^{C} = \mathbf{e}_r^{C} \cdot \mathbf{g} + c \cdot \overline{dt_r} + ISB^{C} + M \cdot T_{ZWD} + \overline{I_1^{C}} + \varepsilon_{p_1^{C}} \\ P_{r,1}^{E} = \mathbf{e}_r^{E} \cdot \mathbf{g} + c \cdot \overline{dt_r} + ISB^{E} + M \cdot T_{ZWD} + \overline{I_1^{E}} + \varepsilon_{p_1^{E}} \\ L_{r,1}^{G} = \mathbf{e}_r^{G} \cdot \mathbf{g} + c \cdot \overline{dt_r} + M \cdot T_{ZWD} - \overline{I_1^{G}} + \omega_1^{G} + \overline{N_{r,1}^{G}} + \varepsilon_{L_1^{G}} \\ L_{r,1}^{C} = \mathbf{e}_r^{C} \cdot \mathbf{g} + c \cdot \overline{dt_r} + ISB^{C} + M \cdot T_{ZWD} - \overline{I_1^{C}} + \omega_1^{C} + \overline{N_{r,1}^{C}} + \varepsilon_{L_1^{C}} \\ L_{r,1}^{E} = \mathbf{e}_r^{E} \cdot \mathbf{g} + c \cdot \overline{dt_r} + ISB^{E} + M \cdot T_{ZWD} - \overline{I_1^{E}} + \omega_1^{E} + \overline{N_{r,1}^{E}} + \varepsilon_{L_1^{E}} \\ \chi^{G/C/E} = \Delta\overline{I_1^{G/C/E}} + \varepsilon_{\chi^{G/C/E}} \end{cases} \tag{15}$$

where G, C, and E denote the GPS, BDS-3, and Galileo satellites, respectively. ISB denotes the inter-system bias (ISB). χ is the virtual observation of the regional between-satellite SD ionospheric delays. ε_χ is the noise of virtual observation. It is worth noting that the coefficient matrix of virtual observation is given by Equation (13) in [19]. The estimable parameters $\prod$ of the multi-GNSS RT SF-PPP can be summarized as

$$\prod = [\mathbf{g}, \overline{dt_r}, ISB^{C/E}, T_{ZWD}, \overline{I_1^{G/C/E}}, \overline{N_{r,1}^{G/C/E}}] \tag{16}$$

To achieve fast convergence of RT SF-PPP, the weight of virtual ionospheric observations needs to be given as accurately as possible. Through extensive testing, an empirical variance $\sigma^2_{\Delta\overline{I_1}}$ of the regional between-satellite SD ionospheric delay can be set as

$$\sigma^2_{\Delta\overline{I_1}} = (v^2 + \frac{v^2}{\sin(Ele)}) \cdot \beta \tag{17}$$

where v and β are empirical coefficients, which can be set to 0.5 and 40, respectively. Ele denotes the satellite elevation.

3. Experiment Datasets and Processing Strategies

Before verifying the feasibility and effectiveness of the novel algorithm proposed in this study, the datasets and processing strategies of the experiment need to be described in detail.

3.1. Experiment Datasets

Twenty-two MGEX stations located in Europe were selected to build a regional reference network. The distribution of these multi-GNSS monitoring stations is shown in Figure 1. It should be noted that 18 blue stations are used for modeling the regional RT between-satellite SD ionospheric delays, and 4 red stations are used to carry out the multi-GNSS RT SF-PPP based on the between-satellite SD ionospheric constraints. The experimental period was set between DoY (Day of Year) 305 to 314 in 2022. The ionospheric conditions over these days can be presented in Figure 2. Except for DoY 310 and 314, the ionosphere is relatively active on all other dates, with over half of the geomagnetic Kp values exceeding 2. Especially on the DoY 307 and 311, some Kp values may even up to 4–5 and indicating intense ionospheric activity. Corresponding, the relatively high solar activity can be observed during the testing period since the most F10.7 index more than 20.

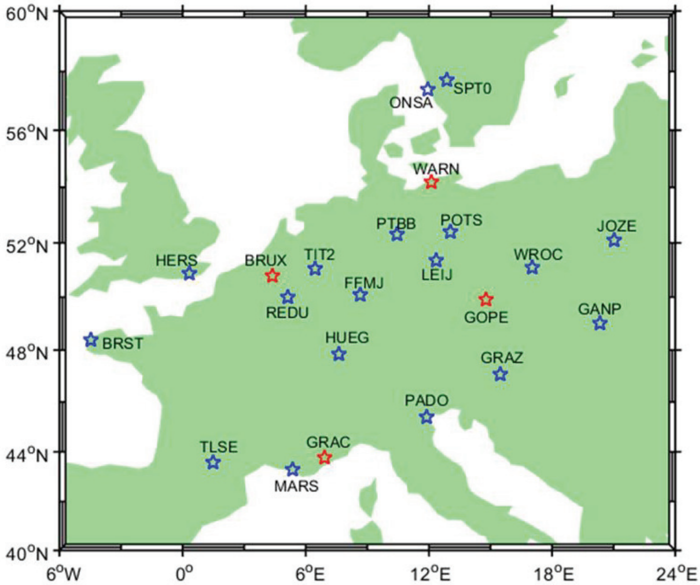


Figure 1. Distribution of the selected 22 MGEX stations in Europe.

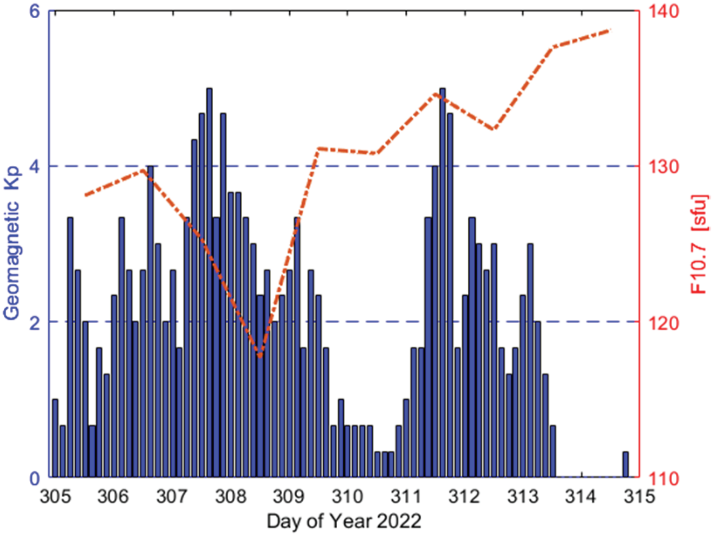


Figure 2. Geomagnetic Kp values and F10.7 index during the testing period.

3.2. Processing Strategies

In the processing of multi-GNSS RT SF-PPP, both satellite positions and clocks are fixed by using broadcast ephemeris and CNES (Centre National d'Etudes Spatiales) state space representation (SSR) corrections [2,22]. The satellite DCB can be corrected using the code bias products of CNES SSR in real-time. The weight of observations for different satellites is set using the elevation-dependent weighting model, and the priori precision of code and phase observations are set to 0.3 and 0.003 m, respectively [8]. Considering the lower accuracy of SSR orbits and clocks for BDS-3 inclined geosynchronous orbit (IGSO) satellites (i.e., C38-40), the weight of IGSO observations needs to be set as 1/2 of other medium earth orbit (MEO) satellites [23]. In order to accurately evaluate the positioning accuracy, the precise coordinates of all selected MGEX stations within accuracy of a few mm can be obtained from the SINEX file. Table 1 summarizes some key processing strategies and correction models. In addition, some minor corrections such as relativistic effect and tidal errors are also considered in the positioning domain [24]. The phase wind-up is corrected by using the method of literature [25].

Table 1. Processing strategies and correction models of the multi-GNSS RT SF-PPP.

Items	Strategies or Models
Observation	GPS: L1; Galileo: E1; BDS-3: B1I
Sampling rate	30 s
Elevation cutoff angle	10°
Antenna correction	Corrected by igs14_2233.atx
Dry part of tropospheric delay	Corrected by GPT2w + SAAS + VMF models [26]
Wet part of tropospheric delay	Estimated as random-walk noise
Ionospheric delay	Corrected by GPS Klobuchar model
Residual of ionospheric delay	Estimated as random-walk noise [19]
Positioning estimator	Kalman filter
Receiver coordinates	Estimated as white noise
Receiver clocks	Estimated as white noise
ISB	Estimated as random-walk noise [27,28]
Phase ambiguities	Estimated as float constant

4. Results and Discussion

Before establishing regional RT between-satellite SD ionospheric model, it is necessary to investigate the current positioning accuracy of the multi-GNSS RT SF UU-PPP that can be achieved. Next, both internal and external accord accuracies of the regional RT between-satellite SD ionospheric model need to be evaluated. At last, using the high-precision GRAPHIC (GRoup And PHase Ionospheric Correction) SF-PPP model as a reference, the advantages of the SD ionospheric-constrained RT SF-PPP in terms of convergence and positioning accuracy are analyzed and discussed.

4.1. Performance of the Multi-GNSS Real-Time Single-Frequency UU-PPP in Static Mode

The multi-GNSS observations of 4 red stations shown in Figure 1 for 10 consecutive days (DoY 305-314, 2022) are selected for conducting the GPS + Galileo + BDS-3 static RT SF UU-PPP based on the GPS Klobuchar constraints. Figure 3 gives the time series of RMS positioning errors during the first 3 h, as well as the positioning accuracy in the north (N), east(E), and up(U) directions after 3 h of convergence. It should be noted that the RMS positioning errors for each epoch are calculated from the results of 4 red stations over a period of 10 days. We can see that the vertical (i.e., U direction) RMS positioning error can converge to 10 cm within about 60 min, while for the horizontal component (i.e., the combined error of N and E directions), it takes at least 90 min to reach the same level. After 90 min of convergence, the vertical RMS positioning error can be stabled at approximately 8 cm, but there is still a downward trend in the horizontal component. The horizontal RMS positioning error can be decreased to 6 cm after 180 min of convergence. When conducting RMS statistics on the positioning errors of 3–24 h for each station in consecutive 10 days, the optimal positioning accuracy can be less than 2 cm in horizontal and 4 cm in vertical. The average RMS positioning accuracy of all used stations can reach 2.6, 2.6 and 5.8 cm in the N, E, and U directions, respectively. This indicates that multi-GNSS static RT SF-PPP has the ability to provide reliable cm-level positioning accuracy at present. If the monitoring station positions are fixed to the SINEX precise coordinates rather than estimated as unknown parameters, the higher precision ionospheric observables can be achieved using the multi-GNSS RT SF-PPP technology.

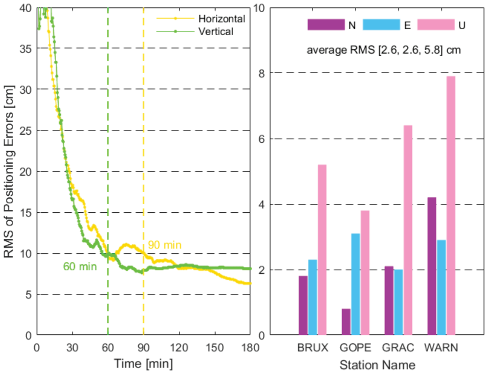


Figure 3. RMS of positioning errors during the first 3 h and positioning accuracy of each station after 3 h convergence in multi-GNSS static RT SF UU-PPP (DoY 305-314, 2022).

4.2. Internal Accord Accuracy of the Regional Real-Time between-Satellite Single-Differenced Ionospheric Model

To evaluate the performance of the regional RT between-satellite SD ionospheric model proposed in this study, its internal accord accuracy as a key indicator needs to be computed. We first extract the slant ionospheric observables of all visible satellites using SF UU-PPP method. Then, the between-satellite SD slant ionospheric delays of each satellite can be obtained after choosing the reference satellite with highest elevation, which are regarded as reference values. If the between-satellite SD slant ionospheric delays are directly derived from the regional SD ionospheric model established by 22 monitoring stations, these results are called model values. The difference between the above reference values and model values can reflect the internal accord accuracy of the regional RT between-satellite SD ionospheric model. Figure 4 shows the time series of between-satellite SD slant ionospheric errors for GPS, BDS-3, and Galileo satellites in DoY 305, 2022. It should be noted that the different colored points represent the results of different satellites. Considering the slow convergence time of the RT SF-PPP, only the extracted ionospheric observables after 3 h are used to establish the regional RT between-satellite SD ionospheric model. It can be seen that the internal accord accuracy of Galileo satellites is better than that of GPS and BDS-3 satellites for all selected stations, its more than half of errors can be lower than 0.1 m. However, the majority of between-satellite SD slant ionospheric errors for both GPS and BDS-3 satellites only maintained within 0.2 m. The proportion of GPS internal accord accuracy exceeding 0.4 m is larger than that of other constellations. This is reasonable that the number of available GPS satellites is significantly more than other constellations, and more results can be displayed in the time series of Figure 4.

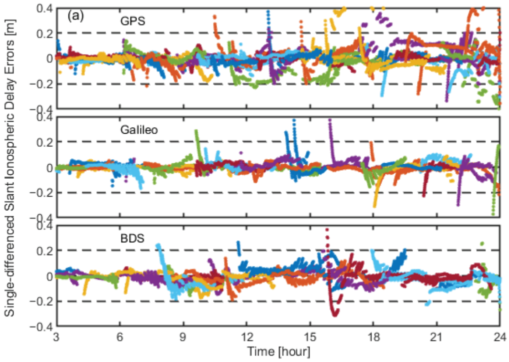


Figure 4. *Cont.*

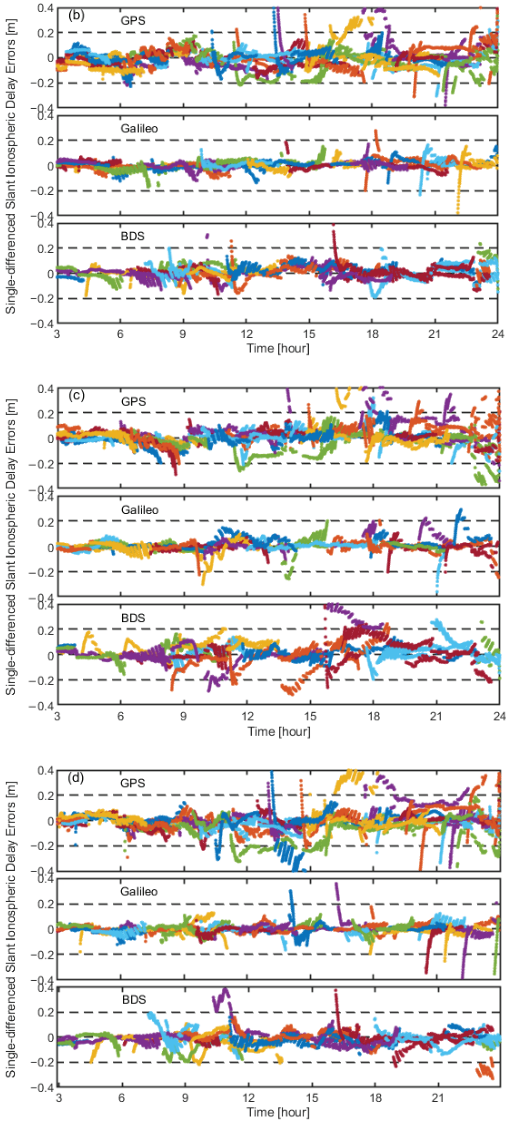


Figure 4. Time series of between-satellite SD slant ionospheric errors for different GNSS satellites at (**a**) BRUX, (**b**) GOPE, (**c**) GRAC, and (**d**) WARN stations in DoY 305, 2022.

The RMS of between-satellite SD slant ionospheric errors calculated from results of 4 selected stations on 10 days are shown in Figure 5. There is a significant difference in the internal accord accuracy of different GPS satellites. The RMS of between-satellite SD slant ionospheric errors for most GPS satellites is lower than 8 cm, while for G05 satellite, its RMS can be up to 15 cm. In contrast, the difference of internal accord accuracy for Galileo and BDS-3 satellites is not much, its RMS is maintained at around 5 cm for Galileo satellites and can be lower than 8 cm for BDS-3 satellites. The average RMS of between-satellite SD slant ionospheric errors for all GPS, Galileo, and BDS-3 satellites are 7.0, 5.1, 7.2 cm, respectively. This proves that the internal accord accuracy of the regional RT between-satellite SD ionospheric model using multi-GNSS SF observations can be better than 0.5 TECU (1 TECU $\approx$ 15.6 cm).

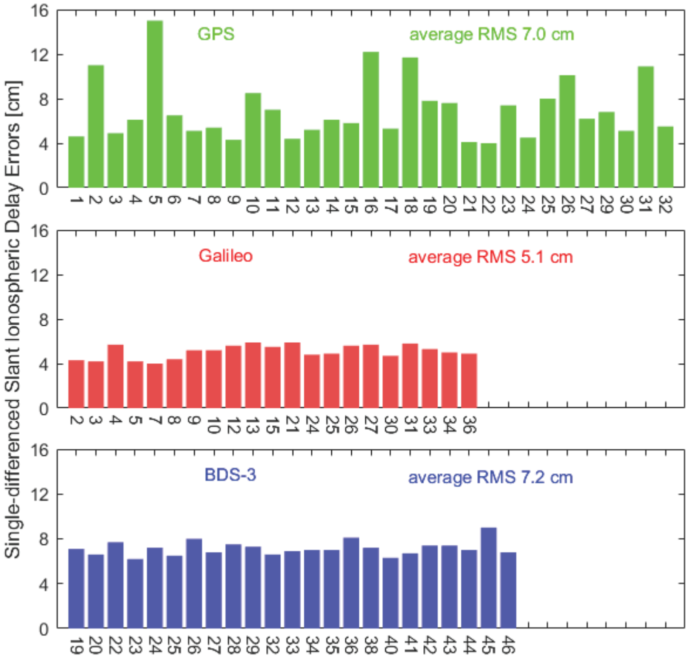


Figure 5. RMS of between-satellite SD slant ionospheric errors for different GNSS satellites (4 red stations in DoY 305-314, 2022).

4.3. External Accord Accuracy of the Regional Real-Time between-Satellite Single-Differenced Ionospheric Model

Although the internal accord accuracy of the regional RT between-satellite SD ionospheric model has excellent performance, its true accuracy still needs to be validated by external accord accuracy. Considering that the differential STEC (dSTEC) derived from epoch-differenced geometry-free (GF) combinations of phase observations have mm-level accuracy, the variation of dSTEC can be further obtained from the between-satellite SD algorithm [29]. This between-satellite SD dSTEC can ebe regarded as reference value. For the regional RT between-satellite SD ionospheric model, the between-satellite SD slant ionospheric delay can be directly calculated, and then the model value of dSTEC variation was obtained using the epoch-differenced method [19]. Similarly, the above model value of dSTEC variation can be calculated from the GPS Klobuchar, BDGIM, and CNES SSR VTEC ionospheric models. Therefore, the difference between the model and reference values can be defined as an indicator of external accord accuracy. Figure 6 shows the differences in dSTEC variation of 4 red stations using GPS, Galileo, and BDS-3 satellites. It is worthing that the calculate interval of external accord accuracy was set to 5 min and the results of elevation exceeding 10 degrees are removed in this assessment. The abbreviation for "GPS-Klo" and "SD-VTEC" are represented as the GPS Klobuchar model and the regional RT between-satellite SD ionospheric model, respectively. We can see that the dSTEC variation differences of different stations have similar distribution, and their values decrease with the increase of elevation. The dSTEC variation differences of the GPS Klobuchar model have the maximum range and even exceed ±5 TECU. There is not much difference in external accord accuracy among the GPS Klobuchar, BDGIM and SSR VTEC models. Compared with the above three models, the variation range of the regional RT between-satellite SD ionospheric model in external accord accuracy is mainly concentrated in ±2 TECU and can be lower than 1 TECU at high elevation. This indicates that the regional RT between-satellite SD ionospheric model has higher accuracy as expected.

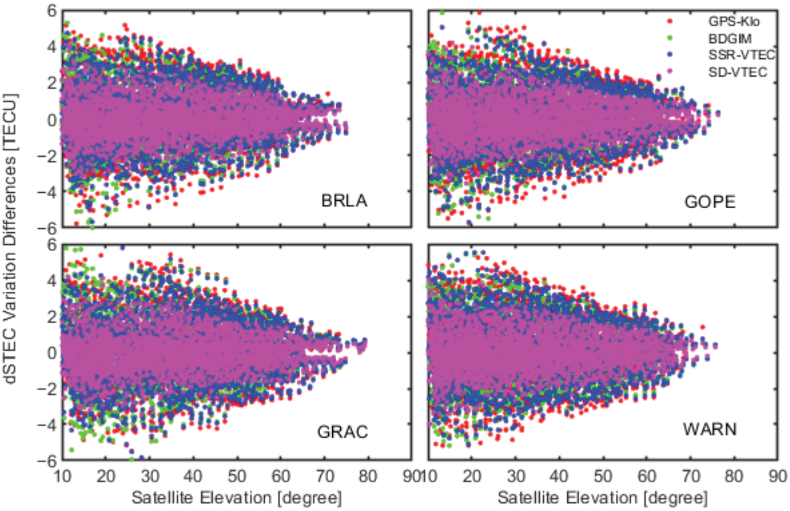


Figure 6. Differences in dSTEC variation of GPS, Galielo, and BDS-3 satellites at 4 red stations (DoY 305, 2022).

The RMS of dSTEC variation differences for all selected ionospheric models using 4 red stations throughout the entire testing period is given in Figure 7. Except for the regional RT between-satellite SD ionospheric model with RMS value of below 1.0 TECU, the RMS of dSTEC variation differences for all other ionospheric models is more than 1.3 TECU and even up to 1.6 TECU. The average RMS values of all stations in external accord accuracy are 1.60, 1.44, 1.39, and 0.96 TECU for the GPS Klobuchar, BDGIM, SSR VTEC, and regional RT between-satellite SD ionospheric models, respectively. This result demonstrates the feasiblilty of modeling the high-precision ionospheric delay using multi-GNSS SF observations.

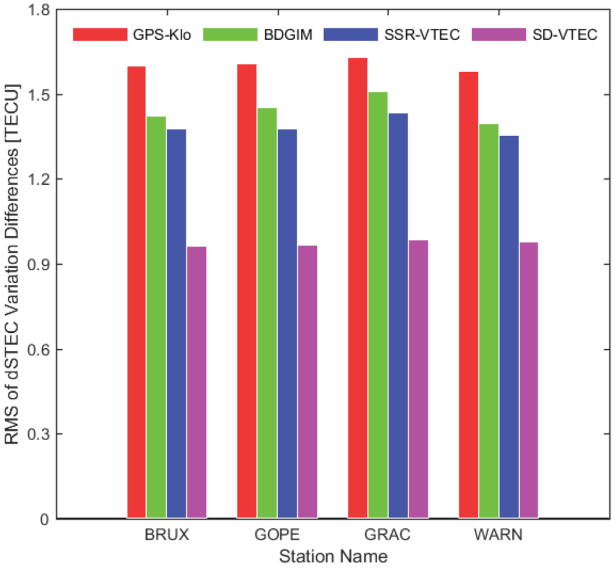


Figure 7. RMS of dSTEC variation differences using different ionospheric models at 4 red stations (DoY 305-314, 2022).

4.4. Multi-GNSS Real-Time Single-Frequency PPP Based on the Regional between-Satellite Single-Differenced Ionospheric Constraints

As long as the accuracy of external ionospheric constraints is high enough, the positioning accuracy and (re-)convergence of the UU-PPP can be improved in theory [30]. In order to explore the advantages of the SD ionospheric constrained multi-GNSS RT SF-PPP, the results of high-precision GRAPHIC RT SF-PPP can be used as a reference. Due to the significant influence of code observation noise on the GRAPHIC model, an RT phase smoothing code observation method named CNMC (Code Noise and Multipath Correction) filter was adopted to further improve the performance of the GRAPHIC RT SF-PPP in this study [31]. Figure 8 shows the daily solutions of the GPS + Galileo + BDS-3 RT SF-PPP in kinematic mode for 2 red stations on DoY 305, 2022. It should be noted that all available satellites are reinitialized with the interval of 6 h to simulate GNSS signal interruption or missing. It is very clear to see that the positioning errors of the GRAPHIC SF-PPP at reinitialized epoch is significantly larger than that of the SD ionospheric-constrained (SDIC) SF-PPP and even exceed 1 m. No matter what SF-PPP models, the positioning accuracy of horizontal component is better than that of vertical component. After convergence, the horizontal and vertical positioning errors can be stable within around 0.2 and 0.5 m, respectively. This indicates that both GRAPHIC and SDIC RT SF-PPP models have the ability to achieve dm-level positioning accuracy in kinematic mode at present.

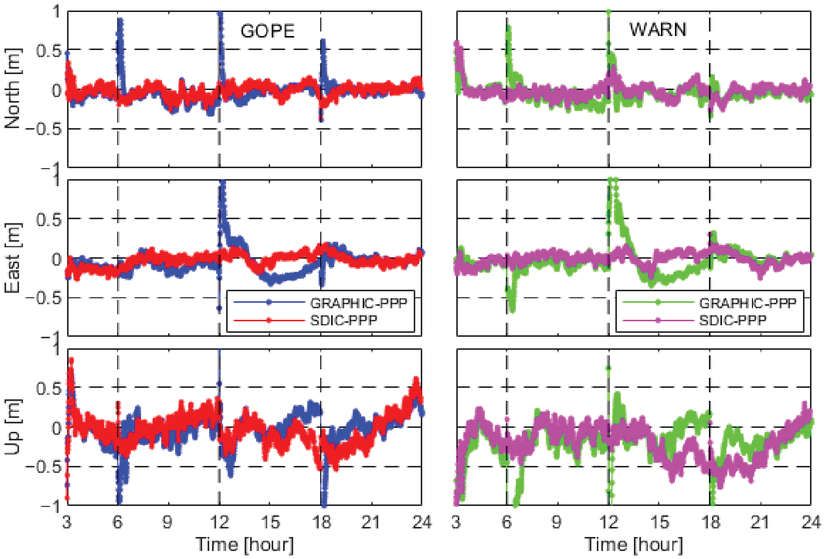


Figure 8. Kinematic positioning errors of both GRAPHIC and SDIC RT SF-PPP using GPS + Galileo + BDS-3 observations at GOPE and WARN stations (DoY 305, 2022).

The convergecne curve of the GPS + Galileo + BDS-3 RT SF-PPP in kinematic mode at 68% confidence level is displayed in Figure 9. The absolute positioning errors of 4 red stations during the whole testing period are sorted from small to large at each epoch, and the 68th% of absolute positioning errors for each epoch is selected as representative to measure the convergence performance [20]. We can see that the horizontal positioning errors of the GRAPHIC and SDIC RT SF-PPP remain consistent in the initialization period, but the vertical positioning errors of the SDIC RT SF-PPP are always smaller than those of the GRAPHIC model during the first 50 min. In the case of setting 0.2 m as the convergence criterion of positioning errors, the convergence time of the SDIC RT SF-PPP with 37 min is less than that of the GRAPHIC RT SF-PPP with 47 min, and its improvement of convergecne speed can up to 21.3%. In the period of re-convergence, due to the introduction of the regional RT bettween-satellite SD ionospheric model, both horizontal and vertcial positioning errors of the SDIC RT SF-PPP can be maintained within 0.2 m. Such excellent positioning performance is significantly superior to the GRAPHIC model, which takes at least 75 min to converge to 0.2 m. This indicates that the positioning errors of the SDIC RT SF-PPP model hardly generate fluctuations when some GNSS signals missing or blocked in the complex kinematic environments.

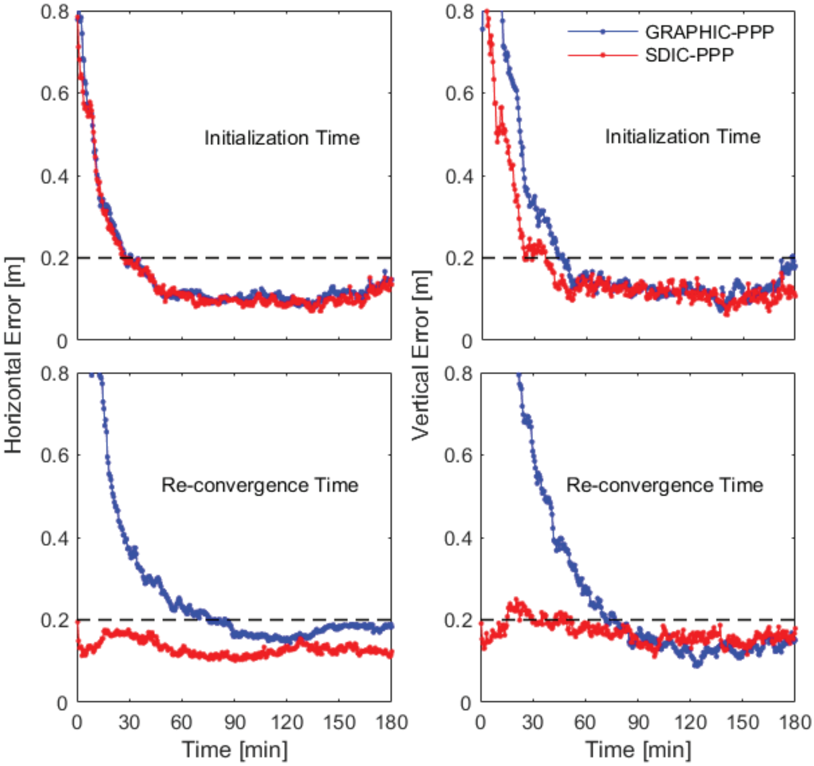


Figure 9. Convergence curve of GPS + Galileo + BDS-3 RT SF-PPP in kinematic mode at 68% confidence level during the first 3 h (4 red stations in DoY 305-314, 2022).

Figure 10 shows the RMS of positioning errors of GPS + Galileo + BDS-3 kinematic RT SF-PPP after convergence of 3 h, and these values are calculated from results of 4 red stations on 10 days. It can be seen that both N and E positioning accuracies of the SDIC RT SF-PPP is better than that of the GRAPHIC model, and its corresponding improvements are 17.0% and 45.7% respectively. However, compared with the GRAPHIC model, the vertical positioning accuracy of the SDIC RT SF-PPP is decreased from 0.21 to 0.26 m, which is caused by the limited accuracy of the regional RT between-satellite SD ionospheric model. If the multi-GNSS DF observations rather than SF observations are used to establish this regional SD ionospheri model, the vertical positioning accuracy of the SDIC RT SF-PPP can be improved like results of [19]. Therefore, this novel multi-GNSS RT SF-PPP model proposed in this study is mainly benefit to improving the horizontal positioning accurcy, with an improvement rate of 37.9%. From the perspective of 3D comprehensive error, the SDIC RT SF-PPP has slighly better accuracy and can reach within 0.3 m.

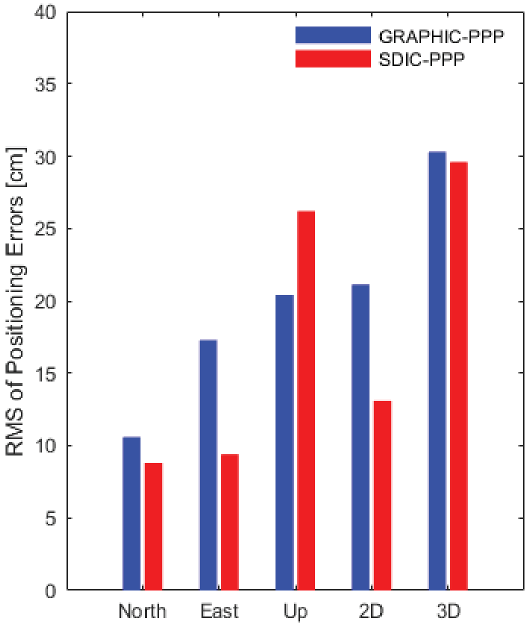


Figure 10. RMS positioning accuracy of GPS + Galileo + BDS-3 RT SF-PPP in kinematic mode after convergence of 3 h (4 red stations in DoY 305-314, 2022).

5. Conclusions

Due to the limited accuracy of ionospheric observables extracted by the CCL method, the UU-PPP technology has gradually become an important solution for extracting high-precision slant ionospheric delays in recent years. Different from the previse research using GNSS DF or multi-frequency (MF) observations, the multi-GNSS SF UU-PPP was used to extract the ionospheric observables in this study. To completely remove the negative effects of receiver DCB with short-term time-varying characteristics, the between-satellite SD method can be adopted for establishing the regional RT ionospheric model. In this contribution, we evaluate the performance of the regional RT between-satellite SD ionospheric model and explore its contribution to the multi-GNSS RT SF-PPP.

The internal accord accuracy of the SD slant ionospheric delays derived from the regional RT between-satellite SD ionospheric model can be better than 0.5 TECU, and its average RMS of GPS, BDS-3, and Galileo satellites are 7.0, 7.2, and 5.1 cm, respectively. Using the dSTEC variation calculated from between-satellite SD and epoch-differenced GF phase observations as a refence, the external accord accuracy of this ionospheric model proposed in this study is significantly better than that of the GPS Klobuchar, BDGIM, and CNES SSR VTEC models, and its average RMS of less than 1.0 TECU can be achieved.

With the introduction of the regional RT between-satellite SD ionospheric model into the multi-GNSS RT SF-PPP in kinematic mode, its initialization speed of vertical positioning errors can be improved by 21.3% to 37 min in comparison with the CNMC-smoothed GRAPHIC model. More importantly, both horizontal and vertical positioning errors of the SDIC SF-PPP after reinitialization can be maintained within 0.2 m. This means that proposed SDIC SF-PPP model has the ability to enhance the continuity and stability of kinematic positioning in case of some satellites missing or blocked. After convergence, the horizontal positioning accuracy of the SDIC SF-PPP can be improved by 37.9% to 0.13 m compared to the GRAPHIC model, but its vertical positioning accuracy may be decreased. With the increase of the number of monitoring stations in the region or the improvement of RTS products quality, the spatial-temporal resolution and accuracy of extracted ionospheric observables can be improved, and result in higher accuracy of the SD ionospheric model. As long as the SD ionospheric model is accurate enough, the vertical positioning accuracy of the SD ionospheric-constrained RT SF-PPP can be improved. In summary, the 3D positioning accuracy of the SDIC SF-PPP can be better than 0.3 m and is superior to other widely used SF-PPP models. To verify the effectiveness of the proposed algorithm on real low-cost SF-PPP users, the truly SF observations collected from the affordable SF devices or chips need to be used for carrying out RT SF-PPP tests in our future research.

Author Contributions: Conceptualization, A.W.; methodology, A.W. and X.L.; software, A.W. and Y.Z.; data curation, A.W. and X.L.; validation, X.L. and H.W.; formal analysis, A.W., Y.Z. and X.L.; writing—original draft preparation, A.W.; writing—review and editing, Y.Z., X.L. and H.W.; supervision, J.C.; funding acquisition, A.W., J.C., X.L. and H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 42304023), National Natural Science Foundation of China (No. 42304015), National Natural Science Foundation of China (No. 42274044), China Postdoctoral Science Foundation (No. 2022M723404), Program of Shanghai Academic/Technology Research Leader (No. 20XD1404500); Jiangsu Province Natural Science Foundation of China (No. BK20231087); Open Research Foundation of the State Key Laboratory of Geodesy and Earth's Dynamics (No. SKLGED2024-3-7).

Data Availability Statement: The datasets analyzed in this study are managed by IGS, and data will be made available on request.

Acknowledgments: The authors would like to thank the IGS, CNES, and MGEX for the provision of GNSS observation data, broadcast ephemeris, real-time SSR corrections, GPS Klobuchar and BDGIM model coefficients.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hernandez-Pajares, M.; Juan, J.M.; Sanz, J.; Aragon-Angel, A.; Garcia-Rigo, A.; Salazar, D.; Escudero, M. The ionosphere: Effects, GPS modeling and the benefits for space geodetic techniques. *J. Geod.* **2011**, *85*, 887–907. [CrossRef]
2. Wang, A.; Chen, J.; Zhang, Y.; Meng, L.; Wang, B.; Wang, J. Evaluating the impact of CNES real-time ionospheric products on multi-GNSS single-frequency positioning using the IGS real-time service. *Adv. Space Res.* **2020**, *66*, 2516–2527. [CrossRef]
3. Odolinski, R.; Teunissen, P.J.G. An assessment of smartphone and low-cost multi-GNSS single-frequency RTK positioning for low, medium and high ionospheric disturbance periods. *J. Geod.* **2018**, *93*, 701–722. [CrossRef]
4. Zou, J.; Wang, A.; Wang, J. Single-Frequency Precise Point Positioning Using Regional Dual-Frequency Observations. *Sensors* **2021**, *21*, 2856. [CrossRef]
5. Klobuchar, J.A. Ionospheric time-delay algorithm for single frequency GPS users. *IEEE Trans. Aerosp. Electron. Syst.* **1987**, *23*, 325–331. [CrossRef]
6. Yuan, Y.; Wang, N.; Li, Z.; Huo, X. The BeiDou global broadcast ionospheric delay correction model (BDGIM) and its preliminary performance evaluation results. *Navigation* **2019**, *66*, 55–69. [CrossRef]
7. Bidaine, B.; Warnant, R. Ionosphere modelling for Galileo single frequency users: Illustration of the combination of the NeQuick model and GNSS data ingestion. *Adv. Space Res.* **2011**, *47*, 312–322. [CrossRef]
8. Wang, A.; Zhang, Y.; Chen, J.; Li, S.; Zhang, Z.; Wang, H. Analysis of spatial–temporal characteristics for BDS-3 broadcast ionospheric models (BDS Klobuchar and BDGIM) in multi-GNSS real-time single-frequency precise point positioning. *Measurement* **2024**, *224*, 113958. [CrossRef]
9. Cai, C.; Gong, Y.; Gao, Y.; Kuang, C. An approach to speed up single-frequency PPP convergence with quad-constellation GNSS and GIM. *Sensors* **2017**, *17*, 1302. [CrossRef]
10. Liu, Q.; Hernández-Pajares, M.; Yang, H.; Monte-Moreno, E.; Roma-Dollase, D.; García-Rigo, A.; Li, Z.; Wang, N.; Laurichesse, D.; Blot, A.; et al. The cooperative IGS RT-GIMs: A reliable estimation of the global ionospheric electron content distribution in real time. *Earth Syst. Sci. Data* **2021**, *13*, 4567–4582. [CrossRef]
11. Chen, L.; Yi, W.; Song, W.; Shi, C.; Lou, Y.; Cao, C. Evaluation of three ionospheric delay computation methods for ground-based GNSS receivers. *GPS Solut.* **2018**, *22*, 125. [CrossRef]
12. Li, Z.; Wang, N.; Hernandez-Pajares, M.; Yuan, Y.; Krankowski, A.; Liu, A.; Zha, J.; Garcia-Rigo, A.; Roma-Dollase, D.; Yang, H.; et al. IGS real-time service for global ionospheric total electron content modeling. *J. Geod.* **2020**, *94*, 32. [CrossRef]
13. Zhang, B.; Ou, J.; Yuan, Y.; Li, Z. Extraction of line-of-sight ionospheric observables from GPS data using precise point positioning. *Sci. China Earth Sci.* **2012**, *55*, 1919–1928. [CrossRef]

14. Liu, T.; Zhang, B.; Yuan, Y.; Li, M. Real-time precise point positioning (RTPPP) with raw observations and its application in real-time regional ionospheric VTEC modeling. *J. Geod.* **2018**, *92*, 1267–1283. [CrossRef]
15. Wang, N.; Yuan, Y.; Li, Z.; Montenbruck, O.; Tan, B. Determination of differential code biases with multi-GNSS observations. *J. Geod.* **2016**, *90*, 209–228. [CrossRef]
16. Li, M.; Yuan, Y.; Zhang, X.; Zha, J. A multi-frequency and multi-GNSS method for the retrieval of the ionospheric TEC and intraday variability of receiver DCBs. *J. Geod.* **2020**, *94*, 102. [CrossRef]
17. Zhang, B.; Zhao, C.; Odolinski, R.; Liu, T. Functional model modification of precise point positioning considering the time-varying code biases of a receiver. *Satell. Navig.* **2021**, *2*, 11. [CrossRef]
18. Xiang, Y.; Gao, Y.; Li, Y. Reducing convergence time of precise point positioning with ionospheric constraints and receiver differential code bias modeling. *J. Geod.* **2020**, *94*, 8. [CrossRef]
19. Wang, A.; Zhang, Y.; Chen, J.; Wang, H. Improving the (re-)convergence of multi-GNSS real-time precise point positioning through regional between-satellite single-differenced ionospheric augmentation. *GPS Solut.* **2022**, *26*, 39. [CrossRef]
20. Wang, A.; Zhang, Y.; Chen, J.; Wang, H.; Yuan, D.; Jiang, J.; Zhang, Z. Investigating the contribution of BDS-3 observations to multi-GNSS single-frequency precise point positioning with different ionospheric models. *Adv. Space Res.* **2024**, *73*, 553–570. [CrossRef]
21. Chen, J.; Zhang, Y.; Yu, C.; Wang, A.; Song, Z.; Zhou, J. Models and performance of SBAS and PPP of BDS. *Satell. Navig.* **2022**, *3*, 4. [CrossRef]
22. Kazmierski, K.; Sosnica, K.; Hadas, T. Quality assessment of multi-GNSS orbits and clocks for real-time precise point positioning. *GPS Solut.* **2018**, *22*, 11. [CrossRef]
23. Chen, J.; Wang, J.; Yu, C.; Zhang, Y.; Wang, B. Improving BDS broadcast ephemeris accuracy using ground-satellite-link observations. *Satell. Navig.* **2022**, *3*, 11. [CrossRef]
24. Petit, G.; Luzum, B. *IERS Conventions*; IERS Technical 2010 Note 36; Verlag des Bundesamts für Kartographie und Geodäsie: Frankfurt am Main, Germany, 2010.
25. Wu, J.T.; Wu, S.C.; Hajj, G.A.; Bertiger, W.I.; Lichten, S.M. Effects of antenna orientation on GPS carrier phase. In Proceedings of the AAS/AIAA Astrodynamics Conference, Durango, CO, USA, 1992; pp. 1647–1660.
26. Boehm, J.; Moller, G.; Schindelegger, M.; Pain, G.; Weber, R. Development of an improved empirical model for slant delays in the troposphere (GPT2w). *GPS Solut.* **2015**, *19*, 433–441. [CrossRef]
27. Zhou, F.; Dong, D.; Li, P.; Li, X.; Schuh, H. Influence of stochastic modeling for inter-system biases on multi-GNSS undifferenced and uncombined precise point positioning. *GPS Solut.* **2019**, *23*, 59. [CrossRef]
28. Lu, Y.; Yang, H.; Li, B.; Li, J.; Xu, A.; Zhang, M. Analysis of Characteristics for Inter-System Bias on Multi-GNSS Undifferenced and Uncombined Precise Point Positioning. *Remote Sens.* **2023**, *15*, 2252. [CrossRef]
29. Feltens, J.; Angling, M.; Jackson-Booth, N.; Jakowski, N.; Hoque, M.; Hernandez-Pajares, M.; Aragon-Angel, A.; Orus, R.; Zandbergen, R. Comparative testing of four ionospheric models driven with GPS measurements. *Radio. Sci.* **2011**, *46*, 1–11. [CrossRef]
30. Li, B.; Zang, N.; Ge, H.; Shen, Y. Single-frequency PPP models: Analytical and numerical comparison. *J. Geod.* **2019**, *93*, 2499–2514. [CrossRef]
31. Chang, Z.; Hu, X.; Guo, R.; Cao, Y.; Wu, X.; Wang, A.; Dong, E. Comparison between CNMC and hatch filter & its precision analysis for BDS precise relative positioning. *Sci. Sin. Phys. Mech. Astron.* **2015**, *45*, 079508. (In Chinese)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

Article

Stationary Detection for Zero Velocity Update of IMU Based on the Vibrational FFT Feature of Land Vehicle

Mowen Li [1,2], Wenfeng Nie [1,*], Vladimir Suvorkin [2], Adria Rovira-Garcia [2], Wei Zhang [1], Tianhe Xu [1] and Guochang Xu [1]

[1] Institute of Space Sciences, Shandong University, Weihai 264209, China
[2] Research Group of Astronomy and Geomatics (gAGE), Universitat Politecnica de Catalunya (UPC), 08034 Barcelona, Spain; vladimir.suvorkin@upc.edu (V.S.)
[*] Correspondence: wenfengnie@sdu.edu.cn

Abstract: The inertial navigation system (INS) and global satellite navigation system (GNSS) are two of the most significant systems for land navigation applications. The inertial measurement unit (IMU) is a kind of INS sensor that measures three-dimensional acceleration and angular velocity measurements. IMUs based on micro-electromechanical systems (MEMSs) are widely employed in vehicular navigation thanks to their low cost and small size, but their magnitude and noisy biases make navigation errors diverge very fast without external constraint. The zero-velocity update (ZVU) function is one of the efficient functions that constrain the divergence of IMUs for a stopped vehicle, and the key of the ZVU is the correct stationary detection for the vehicle. When a land vehicle is stopped, the idling engine produces a very stable vibration, which allows us to perform frequency analysis and a comparison based on the fast Fourier transform (FFT) and IMU measurements. Hence, we propose a stationary detection method based on the FFT for a stopped land vehicle with an idling engine in this study. An urban vehicular navigation experiment was carried out with our GNSS/IMU integration platform. Three stops for 10 to 20 min were set to analyze, generate and evaluate the FFT-based stationary detection method. The FFT spectra showed clearly idling vibrational peaks during the three stop periods. Through the comparison of FFT spectral features with decelerating and accelerating periods, the amplitudes of vibrational peaks were put forward as the key factors of stationary detection. For the consecutive stationary detection in the GNSS/IMU integration process, a three-second sliding window with a one-second updating rate of the FFT was applied to check the amplitudes of peaks. For the assessment of the proposed stationary detection method, GNSS observations were removed to simulate outages during the three stop periods, and the proposed detection method was conducted together with the ZVU. The results showed that the proposed method achieved a 99.7% correct detection rate, and the divergence of the positioning error constrained via the ZVU was within 2 cm for the experimental stop periods, which indicates the effectiveness of the proposed method.

Keywords: stationary detection; inertial navigation system (INS); micro-electromechanical system (MEMS); fast Fourier transform (FFT); zero-velocity update (ZVU); integration

Citation: Li, M.; Nie, W.; Suvorkin, V.; Rovira-Garcia, A.; Zhang, W.; Xu, T.; Xu, G. Stationary Detection for Zero Velocity Update of IMU Based on the Vibrational FFT Feature of Land Vehicle. *Remote Sens.* **2024**, *16*, 902. https://doi.org/10.3390/rs16050902

Academic Editor: Giuseppe Casula

Received: 19 January 2024
Revised: 24 February 2024
Accepted: 25 February 2024
Published: 4 March 2024

1. Introduction

An inertial navigation system (INS) measures specific forces and the angular velocity of its carrier using accelerometers and gyroscopes [1,2]. The direct dynamic measurements of the carrier make the INS capable of processing independent navigation and avoiding external interference for the user. The inertial measurement unit (IMU) is a kind of INS sensor that generally combines three accelerometers and three gyroscopes to produce three-dimensional acceleration and angular velocity measurements. Due to comprising small size, lightweight and low-cost IMUs, micro-electromechanical systems (MEMSs) are widely applied in vehicular navigation with the global navigation satellite system (GNSS) and other sensors [3–5]. Nevertheless, magnitude and noisy biases make MEMSs perform poorly in solo-IMU processing, as errors of navigation parameters accumulate quickly via IMU mechanization, i.e., integrating IMU acceleration and angular velocity measurements to update attitude, velocity and position [6].

To apply the independent observations of an IMU without error accumulation in navigation applications, the integration between the INS and other positioning sensors, like the GNSS, camera and lidar, is well studied and practiced [7–9]. In the integrated processing, the GNSS provides absolute geometric observations to align the INS to the Earth-centered Earth-fixed (ECEF) frame and assists with the correction of INS errors [10,11]. However, the GNSS signals are usually weakened and blocked for dynamic users in urban canyons, tunnels or other occluded environments [12–14]. When there is a lack of geometric constraints from other sensors, like the GNSS, the navigation errors produced via the INS diverge rapidly and terminate accurate navigation.

One of the solutions for inhibiting the divergence of IMU errors in the outage of the other sensors is zero-the velocity update (ZVU) constraint [15]. For land vehicles, employing a ZVU-based GNSS/IMU system was found to benefit both ambiguity resolution and positioning accuracy with GNSS-observation gaps in urban canyons [15]. Applying a ZVU and nonholonomic constraint for the land vehicular GNSS/IMU integration system was able to improve the positioning accuracy by 23.3% and 34.3% with GNSS outages of 30 and 60 s, respectively [6]. For a high-grade IMU, the ZVU could also reduce the velocity and position errors to 24.2% and 9.2% in a five-hour solo-IMU vehicular experiment with frequent and short as well as ten-minute stops [16].

The precondition for applying a ZVU constraint is detecting when a vehicle is going to stop. Many efforts that relied on the velocity, acceleration and angular velocity produced via a vehicular IMU have been made to detect the stationary condition for vehicles [10]. Both velocity and angular velocity were suggested to be compared with a specified threshold, while the standard deviation (STD) was proposed to calculate the vehicle acceleration in a sliding window and was used to compare it with the threshold deduced from the stationary data [6,15]. An attitude heading reference system was put forward to detect the stationary states with acceleration data, which achieved an 87% correction rate in a vehicular test [17,18]. A neural network was also trained to use velocity and IMU measurements for stationary detection in GNSS outages [16,19]. A frequency domain method of forward acceleration was proposed to detect the stationary states, and the frequency in the region of common vehicular vibration was eliminated from the method [20,21]. These previous works focus more on the value and STD of the IMU measurements; however, the feature of the acceleration and angular velocity in the frequency domain has not been considered enough. For the stationary detection of land vehicles, this work focuses on the vibration of an idling engine.

The vibrational frequencies of an idling engine have been studied well regarding land vehicles. The vibration of an idling engine was first taken into account for the optimization of engine mounting systems within the frequency range from 6 to 20 Hz [22,23]. The vibrational frequency of an idling engine was found to be mostly determined by the engine's revolutions per minute (RPM) and its number of cylinders [24–26]. Generally, the vibrational frequency of a land vehicle caused by an idling engine ranges from 10 to 50 Hz [24–28]. When the vehicle is in an idling status, the resonant peak at the double dominant frequency is distinct and considerable [24,25]. It is worth noting that the IMU data sampling frequency is usually in the range of 100 to 200 Hz; hence, the vibrational frequency caused by an idling engine can be availably reflected with the IMU [2,6,10].

This paper proposes a novel stationary detection method based on the fast Fourier transform (FFT) and vibrational features of an idling engine. The algorithm of the FFT is introduced in the Section 2. To assess the proposed method, an urban vehicular navigation experiment and the conducted GNSS/IMU integration system are presented in the Experimental Setup. Based on the three stops designed for the vehicular test, the FFT features of a vehicle in stationary, decelerating and accelerating states are analyzed. The stationary detection method is put forward according to the found features and is applied for the three stop periods together with a three-second sliding window. Finally, the FFT-based stationary detection for land vehicles is evaluated using the correct rate and positioning error constrained by the ZVU.

2. Methods

The key of the proposed stationary detection approach is generating the features of stopped and idling vehicles. For a usual land vehicle, the vibrational frequency of an idling engine falls in the range of 10 to 50 Hz [24–28]. Thus, with the assistance of IMU measurements at a high sampling rate, the FFT serves as a powerful algorithm for spectral analysis in the frequency range of an idling vehicle.

2.1. FFT-Based Vibration Frequency Analysis

Fourier analysis transforms a signal from the original time domain into a representation in the frequency domain. Natural signals, such as the vibration of an object, are continuous (i.e., without gaps). Typically, vibrational signals related to acceleration or angular velocity in each direction of the body frame can be written as the sum of several sinusoidal functions:

$$V(t) = \sum_{j=1}^{n} A_j \sin(2\pi f_j t + \varphi_j) = \sum_{j=1}^{n} V_j(t) \tag{1}$$

where $V(t)$ and $V_j(t)$ denote the total and the jth component of vibration, respectively; A_j, f_j and φ_j represent the amplitude, frequency and initial phase of the single vibration component, respectively.

However, the signals recorded by sensors like IMUs are discrete. The Discrete Fourier Transform (DFT) offers a solution for converting a finite sequence of equally spaced samples into a sequence of complex numbers containing the frequency information [29]. The DFT for a measurement sequence of length N is defined as follows:

$$F(k) = \sum_{n=0}^{N-1} e^{-2\pi i \frac{kn}{N}} M(n),\ k = 0 \sim N-1 \tag{2}$$

where $M(n)$ denotes the nth measurement in the real sequence, $F(k)$ represents the kth component of the DFT and i is the imaginary unit.

With the increase in the length of the sequence, the original DFT requires significantly more resources, as the computation complexity of the DFT is $O(N^2)$. To address this, the FFT is proposed to reduce the computation complexity of the DFT to $O(N \log N)$ [30]. The radix-2 decimation-in-time (DIT) FFT derived from the Cooley–Tukey algorithm is one of the most commonly used solutions that has been widely applied in many domains [31,32]. The promoted DIT algorithm can be expressed as follows:

$$F(k) = \sum_{n=0}^{N/2-1} e^{-2\pi i \frac{kn}{N/2}} M(2n) + \sum_{n=0}^{N/2-1} e^{-2\pi i \frac{kn}{N/2}} M(2n+1) \tag{3}$$

$$F(k + \frac{N}{2}) = \sum_{n=0}^{N/2-1} e^{-2\pi i \frac{kn}{N/2}} M(2n) - \sum_{n=0}^{N/2-1} e^{-2\pi i \frac{kn}{N/2}} M(2n+1) \tag{4}$$

In this algorithm, the computation is simplified by dividing the original discrete sequence into even ($2n$) and odd ($2n+1$) parts, each with a length of $N/2$.

For the FFT, the maximum frequency that can be analyzed within one consecutive measurement sequence is determined by its sampling rate, while the frequency resolution depends on the total time span covered by the sequence as expressed in the following equations:

$$\max(f_{FFT}) = \frac{1}{2} f_{meas} \tag{5}$$

$$\Delta f_{FFT} = \frac{1}{T_{meas}} \tag{6}$$

where f_{FFT} is the frequency sequence obtained via the FFT algorithm; f_{meas} and T_{meas} represent the sampling rate and the time length covering all the measurements, respectively.

2.2. ZVU Constraint in GNSS/IMU Tight Integration

In the GNSS/IMU integration process, GNSS provides absolute position information for IMU to correct navigation errors and align to the ECEF frame. In the present contribution, the kinematic Precise Point Positioning (PPP) is employed for processing multi-GNSS dual-frequency Ionosphere-Free (IF) observations. The state parameters for multi-GNSS IF kinematic PPP and the corresponding observation equations within an extended Kalman filter (EKF) are defined as follows [33–35]:

$$\mathbf{X}_{G,k}^{T} = \begin{bmatrix} \delta \mathbf{p}_{1\times 3}^{e} & \delta c_r & \delta \mathbf{ISB}_{1\times(s-1)} & \delta tro_r & \delta \mathbf{N}_{LC,1\times m} \end{bmatrix}_k \tag{7}$$

$$\mathbf{L}_{G,k} = \begin{bmatrix} \boldsymbol{\varphi}_{G,k} \\ \boldsymbol{\rho}_{G,k} \end{bmatrix} = \mathbf{H}_{G,k}\mathbf{X}_{G,k} - \mathbf{V}_{G,k} \tag{8}$$

$$\mathbf{H}_{G,k} = \begin{bmatrix} \mathbf{H}_{\varphi,k} \\ \mathbf{H}_{\rho,k} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{p,k} & 1 & \mathbf{H}_{ISB,k} & \mathbf{H}_{tro,k} & \mathbf{H}_{N,k} \\ \mathbf{H}_{p,k} & 1 & \mathbf{H}_{ISB,k} & \mathbf{H}_{tro,k} & \mathbf{0} \end{bmatrix}_k \tag{9}$$

where δc_r, $\delta \mathbf{ISB}$, δtro and $\delta \mathbf{N}_{LC}$ denote the errors of the receiver clock, intersystem bias (ISB), tropospheric delay and IF ambiguity, respectively. It is important to note that the italic symbols denote values or numbers, whereas bold symbols represent vectors or matrices. s and m represent the number of employed constellations and IF carrier-phase measurements, respectively; the subscript k denotes the epoch index; $\mathbf{L}_{G,k}$ and $\mathbf{V}_{G,k}$ represent the multi-GNSS observation vector and the corresponding residual vector, respectively; $\boldsymbol{\varphi}_{G,k}$ and $\boldsymbol{\rho}_{G,k}$ are carrier-phase and pseudo-range measurements minus the modeled terms, respectively; $\mathbf{H}_{G,k}$ is the PPP design matrix; $\mathbf{H}_{\varphi,k}$, $\mathbf{H}_{\rho,k}$, $\mathbf{H}_{p,k}$, $\mathbf{H}_{ISB,k}$, $\mathbf{H}_{tro,k}$ and $\mathbf{H}_{N,k}$ represent the design matrices of the corresponding measurement or state parameters in regard to their subscripts.

In the GNSS/IMU integration process, the state parameters of IMU for the EKF are defined as follows:

$$\mathbf{X}_{I,k}^{T} = \begin{bmatrix} \delta \boldsymbol{\varepsilon}_{1\times 3}^{e} & \delta \mathbf{v}_{1\times 3}^{e} & \delta \mathbf{p}_{1\times 3}^{e} & \delta \mathbf{ba}_{1\times 3}^{b} & \delta \mathbf{bg}_{1\times 3}^{b} \end{bmatrix}_k \tag{10}$$

where the superscript e stands for the ECEF frame and b stands for the body frame. In this study, the body frame is defined as the vehicular body Right–Forward–Up (RFU) frame with the same origin as the IMU center. Finally, $\delta \boldsymbol{\varepsilon}$, $\delta \mathbf{v}$, $\delta \mathbf{p}$, $\delta \mathbf{ba}$ and $\delta \mathbf{bg}$ represent the error of attitude, velocity, position, accelerometer bias and gyroscope bias, respectively [10].

For a stopped vehicle, ZVU is a common constraint used to inhibit the divergence of IMU errors. The observation equation for the ZVU constraint at epoch k, with respect to the IMU state parameters, is as follows:

$$\mathbf{L}_{ZVUT,k} = -\mathbf{v}_{imu,k}^{e} = \mathbf{H}_{ZVUT,k}\mathbf{X}_{I,k}^{T} - \mathbf{V}_{ZVUT,k} \tag{11}$$

$$\mathbf{H}_{ZVUT,k} = \begin{bmatrix} \mathbf{0}_{3\times 3} & -\mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \end{bmatrix} \tag{12}$$

where $\mathbf{L}_{ZVUT,k}$ and $\mathbf{V}_{ZVUT,k}$ represent the observation vector and the corresponding residual vector, respectively; $\mathbf{H}_{ZVUT,k}$ denotes the design matrix; $\mathbf{v}_{imu,k}^{e}$ is the velocity in the ECEF frame as produced by the IMU mechanization; $\mathbf{I}$ denotes the identity matrix.

By combining (7) to (12), one can derive the integrated equation for the ZVU constraint-based GNSS-PPP/IMU tight integration as follows [10]:

$$\mathbf{X}_{k}^{T} = \begin{bmatrix} \mathbf{X}_{I,k}^{T} & \delta c_r & \delta \mathbf{ISB}_{1\times(s-1)} & \delta tro_r & \delta \mathbf{N}_{LC,1\times m} \end{bmatrix}_k \tag{13}$$

$$\mathbf{L}_k = \begin{bmatrix} \boldsymbol{\varphi}_{G,k} \\ \boldsymbol{\rho}_{G,k} \\ -\mathbf{v}^e_{imu,k} \end{bmatrix} = \mathbf{H}_k\mathbf{X}_k - \mathbf{V}_k \tag{14}$$

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{H}_{TC,\varphi,k} \\ \mathbf{H}_{TC,\rho,k} \\ \mathbf{H}_{ZVUT,k} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{H}_{p,k} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{H}_{ISB,k} & \mathbf{H}_{tro,k} & \mathbf{H}_{N,k} \\ \mathbf{0} & \mathbf{0} & \mathbf{H}_{p,k} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{H}_{ISB,k} & \mathbf{H}_{tro,k} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{15}$$

where $\mathbf{H}_{TC,\varphi,k}$ and $\mathbf{H}_{TC,\rho,k}$ represent the multi-GNSS design matrix in the GNSS-PPP/IMU tightly integrated equation, as the position errors are uniform with the IMU.

In the present work, Equations (13)–(15) are adopted to implement the GNSS-PPP/IMU tight integration when the vehicle is stopped. When the GNSS signals are blocked or simulated to be in outage during the vehicle stop periods, Equations (10)–(12) are employed for ZVU constraint processing.

3. Experimental Setup

3.1. Hardware Platform

A GNSS/IMU hardware platform was established for testing the land vehicular navigation. Figure 1 depicts the configuration and equipment of the hardware platform on the car. The GNSS receiver and IMU fusion instrument, NovAtel PwrPak7-E2, was secured in the middle of the platform with screws. The GNSS antenna was mounted in front of the IMU within the vehicular body frame. The entire platform was tightly hinged to the roof of the car.

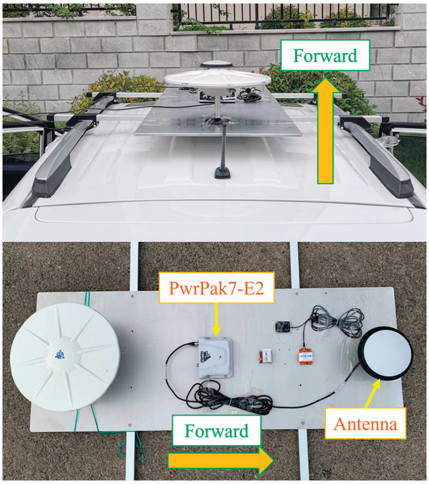


Figure 1. GNSS/IMU hardware platform and its equipment on the land vehicle.

The IMU integrated in the PwrPak7-E2 receiver is the MEMS EG370N with the specification parameters listed in Table 1. The magnitude of gyroscope bias is in the order of hundreds of deg/h with a stability of less than 1 deg/h. The PwrPak7-E2 receiver is capable of tracking multi-GNSS signals from GPS, GLONASS (GLO), Galileo (GAL) and BDS. The produced pseudo-range and carrier-phase measurements were used for the tight integration of the GNSS with the IMU.

Table 1. Specifications of MEMS IMU in PwrPak7-E2.

PwrPak7-E2	Item	Value
Gyroscope	Bias Magnitude	360°/h
	Bias Stability	0.8°/h
	Angle Random Walk	$0.06°/\sqrt{\text{h}}$
Accelerometer	Bias Magnitude	2 mg
	Bias Stability	0.012 mg
	Velocity Random Walk	$0.06\ \text{mg}/\sqrt{\text{Hz}}$

3.2. Experimental Strategy

To evaluate the availability of FFT-based stationary detection with an IMU for an idling car, a vehicular test supported by the GNSS/IMU hardware platform was conducted on a Toyota Prado 2016 with the engine of 7GR-FKS from the GPS time (GPST) 12 October 2021 01:55:20 to 03:50:45. Figure 2 depicts the test trajectory in the urban area of Weihai City, China. The car started moving after the turn-on of the hardware platform and moved forward in a straight line for a while to initialize the yaw angle of the IMU. After aligning the IMU, the vehicle was stopped with the engine idling for three periods lasting 10 to 20 min each. The time spans of these three stops, along with the associated ten-second decelerating and accelerating phases, are detailed in Table 2. The IMU measurements of accelerations and gyroscopes during these three stop periods were used to implement FFT processing to examine the vibrational features of our stationary idling vehicle. Based on the experimental evidence, a three-second sliding window FFT was employed for the stationary detection. To evaluate the proposed detection method, GNSS observations were removed to simulate a complete GNSS outage during each stop period. Then, the proposed detection method was implemented together with the ZVU constraint. Finally, the correct detection rate was calculated, and the positioning errors were compared with the errors of solo-IMU processing without the ZVU constraint.

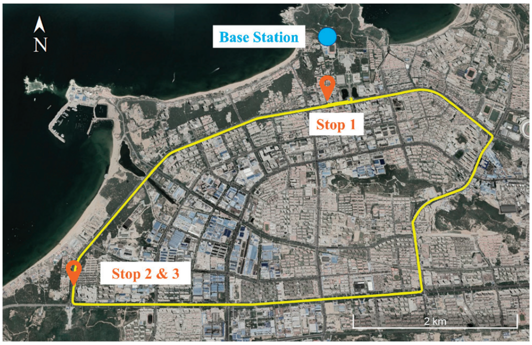


Figure 2. Urban trajectory (yellow), base station (blue) and three stop points in vehicular test.

Table 2. Time range of stop, deceleration and acceleration of the three stop periods.

Period	State	Start	End
1	Stop	02:04:07	02:20:13
	Decelerating	02:03:56	02:04:06
	Accelerating	02:20:14	02:20:24
2	Stop	02:37:31	02:59:32
	Decelerating	02:37:20	02:37:30
	Accelerating	02:59:33	02:59:43
3	Stop	03:21:42	03:41:54
	Decelerating	03:21:31	03:21:41
	Accelerating	03:41:55	03:42:05

The EKF-based GNSS-PPP/IMU tight integration was conducted to obtain the navigation solution for the entire trajectory, with the processing strategy detailed in Table 3. The kinematic PPP was implemented to obtain the multi-GNSS observation equations for the tight integration with the IMU. The IF combination was utilized to eliminate 99.9% of the ionospheric delay in GNSS measurements. Positioning errors, receiver clock errors, tropospheric delay, intersystem bias and carrier-phase ambiguities were set as the filter state parameters to be estimated. The base station equipped with a Septentrio PolaRx5 receiver was used to obtain a high-accuracy reference position for the trajectory, by employing multi-GNSS double-differenced (DD) Real-Time Kinematics (RTK) [33]. The GNSS Doppler velocity was used to initialize the yaw of the IMU, while the roll and pitch were initialized based on the direction of gravity [10]. Note that all accelerometer and gyroscope measurements of the IMU were converted from the IMU frame to the vehicular RFU body frame. Hence, in the present paper, the accelerometer X, Y and Z measurements represent right, forward and up accelerations, respectively, and the gyroscope X, Y and Z measurements account for the angular rotational velocity of pitch, roll and yaw, respectively. The sampling rate of the MEMS IMU is 200 Hz, resulting in the maximum FFT frequency of 100 Hz as derived from (5).

Table 3. Processing strategy of post-GNSS-PPP/IMU tight integration.

Module	Item	Strategy
GNSS	Mode	Post-kinematic PPP
	Observations	IF combination of GPS: C1C-C2W/L1C-L2W; GLO: C1C-C2P/L1C-L2P; Galileo: C1C-C5Q/L1C-L5Q; BDS: C2I-C6I/L2I-L6I
	Elevation cutoff	15°
	Measurement weight	Elevation-dependent weight
	Satellite products	Multi-GNSS precise orbit and clock products from IGS
	Receiver clock offset	White noise process
	ISB	Constant mode
	Tropospheric delay	Estimate the zenith total delay as a random walk process
	Phase windup	Corrected for rover
	Ambiguity resolution	Float
IMU	Alignment	GNSS Doppler velocity
Integration	Solution	Tightly coupled

4. Experimental Results and Discussion

This section describes how the DIT FFT algorithm, as introduced previously in the Methods, was employed to analyze the vibrational features of the idling vehicle. The consecutive measurements from IMU accelerometers and gyroscopes during the three stop periods are addressed using the FFT. In order to compare the results, the decelerating and accelerating periods before and after the three stops are also processed by the FFT algorithm. A stationary detection based on a three-second sliding window method is proposed to extract the FFT features. The present section ends with the artificial removal of GNSS observations during the three stop periods to simulate a complete GNSS outage. Then, the three stop periods are re-assessed using the proposed method and constrained by ZVU. Finally, the correct detection rate is calculated as a reliable indicator of the proposed method, and the results of the IMU constrained by ZVU are presented in comparison with the results without constraints for a complete evaluation.

4.1. FFT Features of Acceleration and Angular Velocity for Stationary Idling Vehicle

Figure 3 depicts the FFT spectrum of acceleration amplitude with respect to frequency for the three stop periods. The IMU acceleration reveals several obvious peaks in all the three directions during the stationary period. Moreover, the two highest peaks in the ranges of 10 to 20 Hz and 20 to 30 Hz show significantly greater amplitude compared to the other frequencies in the forward and up directions. The frequencies of the two peaks correspond to the general vibration frequency range of 10 to 50 Hz for an idling vehicle [24–28]. Although the same apparent peak is observed from 10 to 20 Hz in the right direction, there is no peak within the 20 to 30 Hz range, which is accompanied by more noise than in the other two directions. In contrast, gyroscope measurements presented opposite features, as depicted in Figure 4. The angular velocity presents only one distinct peak within the 10 to 20 Hz range along the X-axis. Even though the gyroscope spectra show the same peaks as the accelerometers within the 10 to 20 Hz and 20 to 30 Hz ranges along the Y- and Z-axes, too many strong chaotic peaks mask these two peaks. The opposite behavior in axes between accelerometers and gyroscopes is caused by the orthogonality of acceleration and angular velocity vectors in periodic vibrations.

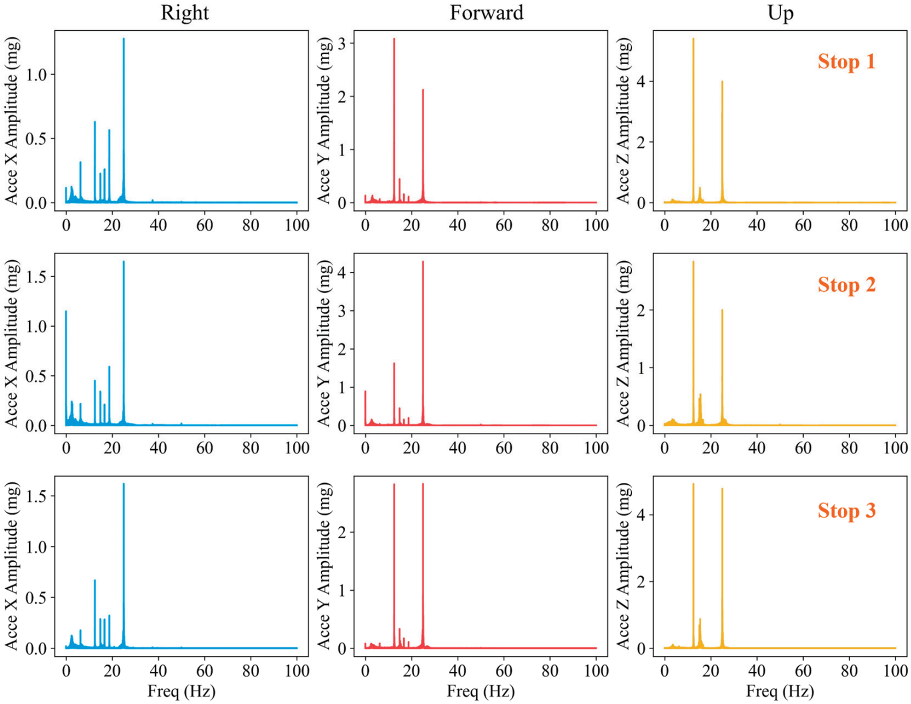


Figure 3. Acceleration FFT spectrum of the Stop 1 (**top row**), Stop 2 (**middle row**) and Stop 3 (**Bottom row**) in the right (**left column**), forward (**middle column**) and up (**right column**) directions.

The FFT spectra of accelerometers and gyroscopes clearly delineate the vibrational peaks within the 10 to 20 Hz and 20 to 30 Hz ranges. To analyze these peaks, the spectra are segmented into three frequency ranges: 0 to 10 Hz, 10 to 20 Hz and 20 to 30 Hz, with the statistics of these peaks summarized in Table 4. The result indicates that the frequencies of the highest peaks within the 10 to 20 Hz and 20 to 30 Hz ranges are 12.495 Hz and 24.991 Hz, respectively. The dominant frequency is 12.495 Hz, which indicates that the idling speed of the engine of the experimental vehicle is approximately 750 RPM [24,25]. For simplicity, the peaks at 12.495 Hz and 24.991 Hz are designated as Peak-1 and Peak-2, respectively, with the frequency of the latter being the twice that of the former. These two frequencies are consistent for both acceleration and angular velocity across the three stop periods, as the STDs are even less than the FFT frequency interval. Note that the FFT frequency intervals derived from (6) for the three stop periods are 1×10^{-3}, 7×10^{-4} and 8×10^{-4} Hz, respectively, corresponding to the duration of each stop. For the pitch angular velocity, only Peak-1 is prominent compared to the other frequencies. The highest peaks within the 0 to 10 Hz range (denoted by Peak-0) exhibit more uncertainty with the STDs of frequencies ranging from 0.1 Hz to several Hz. Overall, the amplitudes of Peak-1 and Peak-2 are 7 and 37 times greater than that of Peak-0 for the forward and up acceleration, respectively, and the amplitude of Peak-1 is 67 times greater than that of Peak-0 for the pitch angular velocity.

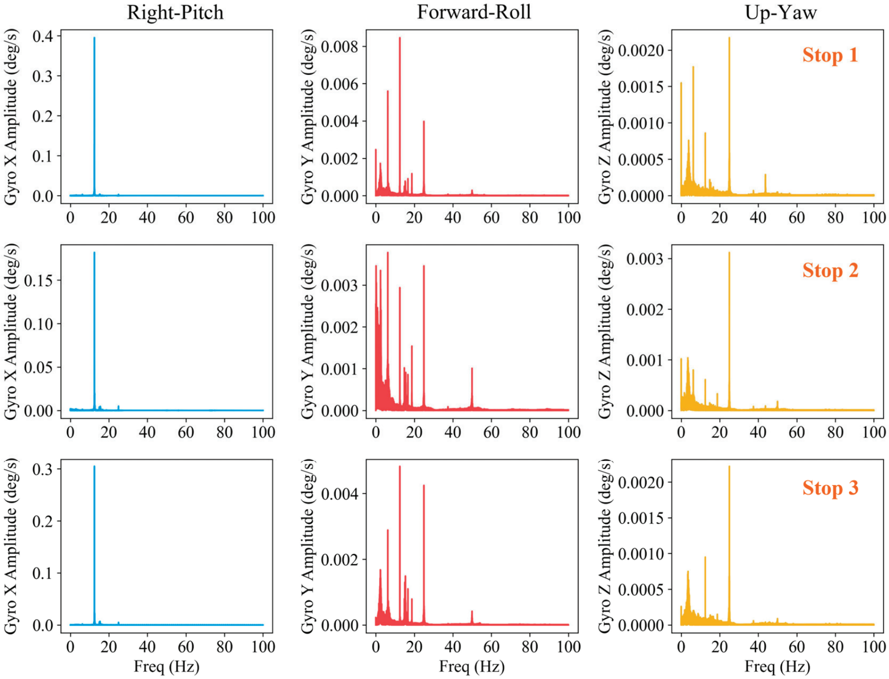


Figure 4. Angular velocity FFT spectrum of Stop 1 (**top row**), Stop 2 (**middle row**) and Stop 3 (**Bottom row**) in the right (**left column**), forward (**middle column**) and up (**right column**) directions.

Table 4. Analysis of the forward/up acceleration and pitch angular velocity with respect to the three frequency ranges (Hz) in the three stops: highest peak, mean value and STD.

Stop Period	Freq Range (Hz)	Acce Y		Acce Z		Gyro X	
		Freq (Hz)	Amp (mg)	Freq (Hz)	Amp (mg)	Freq (Hz)	Amp (deg/s)
1	0~10	3.071	0.133	3.305	0.105	6.247	0.003
	10~20	12.495	3.085	12.495	5.402	12.495	0.395
	20~30	24.991	2.124	24.991	3.994	24.991	0.003
2	0~10	0.001	0.890	3.332	0.107	0.100	0.002
	10~20	12.496	1.622	12.496	2.839	12.496	0.182
	20~30	24.991	4.283	24.991	2.001	24.991	0.005
3	0~10	6.248	0.085	3.573	0.113	6.248	0.002
	10~20	12.496	2.821	12.496	4.927	12.496	0.305
	20~30	24.991	2.829	24.991	4.782	24.991	0.005
Mean	0~10	3.106	0.369	3.404	0.108	4.198	0.002
	10~20	12.495	2.509	12.495	4.389	12.495	0.294
	20~30	24.991	3.078	24.991	3.592	24.991	0.004
STD	0~10	2.550	0.369	0.121	0.003	2.898	5×10^{-4}
	10~20	4×10^{-4}	0.637	4×10^{-4}	1.113	4×10^{-4}	0.087
	20~30	2×10^{-4}	0.899	2×10^{-4}	1.171	2×10^{-4}	0.001

4.2. FFT Features of Acceleration and Angular Velocity in Decelerating and Accelerating

For comparison with the features of a stationary idling vehicle, the FFT is also implemented and analyzed for decelerating and accelerating near the three stop periods; the corresponding periods are exhibited in Table 2. The FFT spectra of forward/up acceleration during the decelerating and accelerating periods are depicted in Figures 5 and 6. For the acceleration, Peak-1 and Peak-2 are still observable in the up direction rather than the forward direction during the decelerating and accelerating period, since the motive force of the vehicle is almost along the forward direction. However, because Peak-0 becomes more striking, the vibrational peaks in the forward and up directions are not as prominent as they are during the stationary period. A similar phenomenon is also observed in the angular velocity, where the amplitude of Peak-0 is increased to the same magnitude as Peak-1.

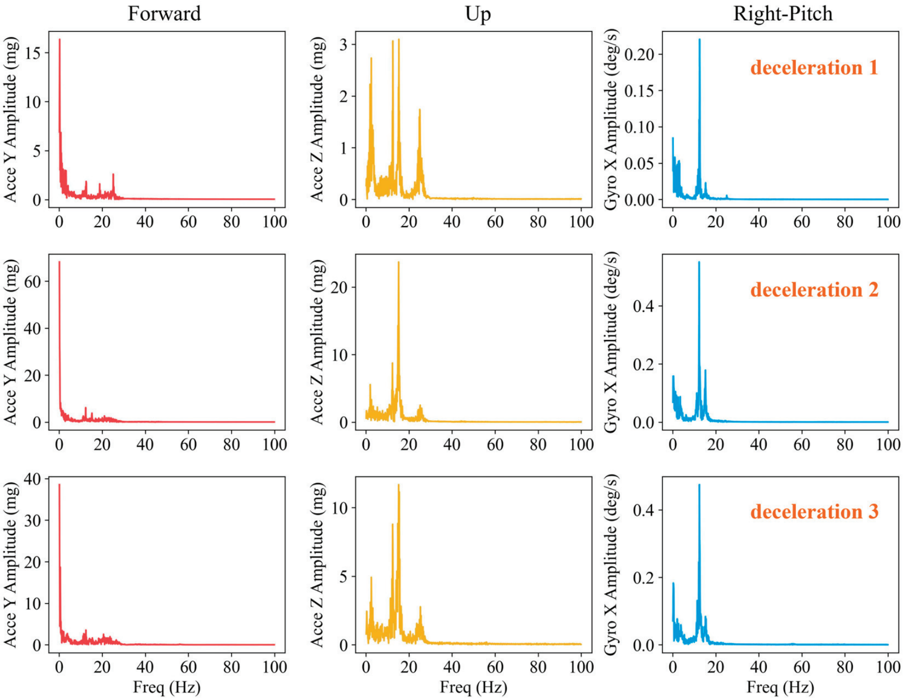


Figure 5. FFT spectrum of forward (**left column**) and up acceleration (**middle column**) and pitch angular velocity (**right column**) during deceleration 1 (**top row**), deceleration 2 (**middle row**) and deceleration 3 (**bottom row**).

Comprehensively taking the FFT features of acceleration and angular velocity during stationary, decelerating and accelerating periods into account, we propose a method of FFT-based stationary detection as depicted by red nodes in Figure 7: a three-second sliding window with a one-second updating rate, i.e., a two-second overlap between the two adjacent windows, is adopted to perform the FFT processing for the forward/up acceleration and the pitch angular velocity. The number of measurements to process with the FFT in a single window is 600 with respect to the 200 Hz sampling rate of the IMU. Once the FFT window is updated, the frequency and amplitude of Peak-0, Peak-1 and Peak-2 are calculated and compared. On one hand, a larger window size might cause the wrong detection of a stationary condition at the start and the end of the vehicle stop, since the window contains more historical dynamic information. On the other hand, a smaller window size could make the detection easily affected by random disturbances or measurement errors. Therefore, a three-second window size is selected as the optimal size of the sliding window.

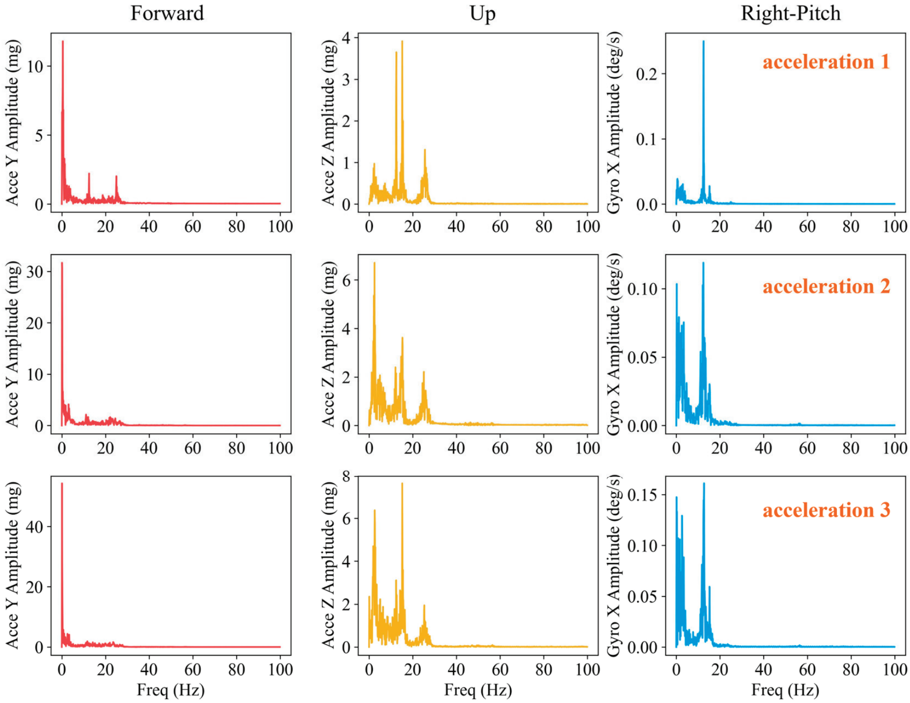


Figure 6. FFT spectrum of forward (**left column**) and up acceleration (**middle column**) and pitch angular velocity (**right column**) during acceleration 1 (**top row**), acceleration 2 (**middle row**) and acceleration 3 (**bottom row**).

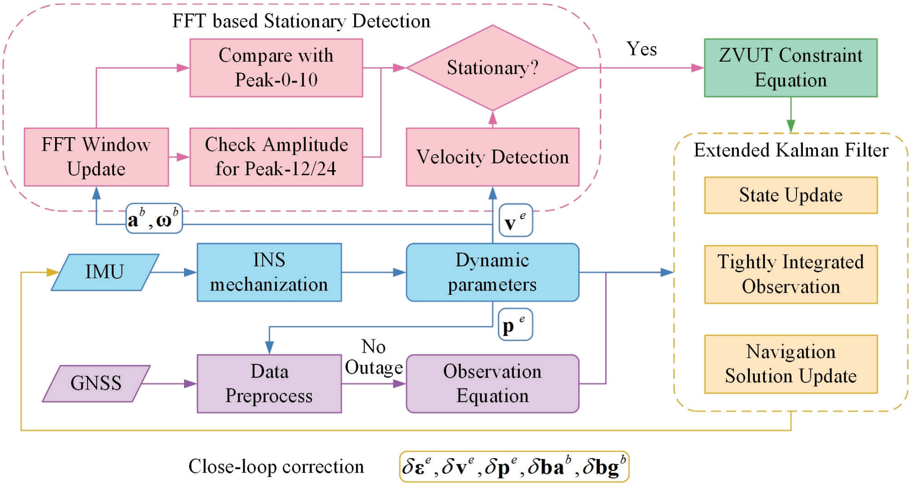

FFT based Stationary Detection
Compare with Peak-0-10
Stationary?
Yes
ZVUT Constraint Equation
FFT Window Update
Check Amplitude for Peak-12/24
Velocity Detection
Extended Kalman Filter
State Update
Tightly Integrated Observation
Navigation Solution Update
$\mathbf{a}^b, \boldsymbol{\omega}^b$
$\mathbf{v}^e$
IMU
INS mechanization
Dynamic parameters
$\mathbf{p}^e$
No Outage
GNSS
Data Preprocess
Observation Equation
Close-loop correction
$\delta\boldsymbol{\varepsilon}^e, \delta\mathbf{v}^e, \delta\mathbf{p}^e, \delta\mathbf{ba}^b, \delta\mathbf{bg}^b$

Figure 7. Flow chart of FFT-based stationary detection and ZVU in the GNSS/IMU tight integration.

The entire window period is considered as stationary if the FFT result satisfies three conditions. First, the vehicular velocity must be below a threshold of 0.5 m/s [6]. Second, the amplitude of all three peaks of acceleration and angular velocity must be lower than the threshold of 15 mg and 0.7 deg/s, respectively. Third, both amplitudes of Peak-1 and Peak-2 of the forward/up acceleration must be higher than the amplitude of Peak-0, and the amplitude of Peak-1 of pitch angular velocity is required to be at least 7.0 times greater than that of Peak-0. In terms of acceleration, the amplitudes of Peak-1 and Peak-2 are multiplied by an amplification factor of 1.5, as there might be some external disturbances, e.g., passenger movements while opening or entering/exiting the car, which may randomly impact the vibration of the vehicle.

The FFT-based stationary detection and ZVU in the GNSS-PPP/IMU tight integration process is depicted in Figure 7 using different colored modules. At the start of every integration epoch, dynamic parameters are produced by IMU mechanization and propagated to other subprocesses in the navigation implementation. For stationary detection, IMU acceleration $\mathbf{a}^b$ and angular velocity $\boldsymbol{\omega}^b$ are adopted to implement the FFT analysis and frequency detection, while IMU velocity $\mathbf{v}^e$ is employed for velocity detection. If all detection nodes return a positive stationary result, a ZVU constraint equation is generated and propagated to the EKF module. Then, the GNSS module receives IMU position $\mathbf{p}^e$ for data preprocessing and outage checking. If no outage is found, GNSS observation equations are calculated and forwarded to the EKF module. Finally, the EKF module updates all state parameters and generates tightly integrated observations to update the navigation solution. The estimated IMU errors are transmitted back to IMU for a closed-loop correction.

4.3. FFT-Based Stationary Detection Results

To assess the performance of the proposed stationary detection method, the GNSS observations during the three stop periods are artificially removed to simulate a complete outage of GNSS. The proposed method is conducted together with ZVU constraints for the IMU navigation processing. The FFT spectra of forward/up acceleration and pitch angular velocity in regard to three-second sliding windows are exhibited in Figure 8. Sliding window results in consecutive ten-second intervals of each stop period are selected as examples. The sliding window FFT result presents consistent features of both acceleration and angular velocity for each entire stationary period. Note that because of the spectral leakage caused by the short window size of three seconds [30,36,37], an extra peak can be seen in Figure 8 besides Peak-1 and Peak-2, especially for Peak-1 of the acceleration in the up component. Although the extra peak is produced from the real Peak-1, only the peak closest to the frequency Peak-1, located at 12.333 Hz, is used in the detection processing, as the extra peak may be contaminated by random noise or unpredictable disturbances.

The statistical results of the FFT-based stationary detection across the three stop periods are presented in Table 5. The correct detection rate is determined as the ratio of the number of seconds that are correctly detected as stationary to the total duration of the stop period. For the three stop periods, all the correct rates exceed 99%, and the aggregated correct rate is 99.7%. The failed detection of some epochs is caused by passenger movements of opening and entering/exiting the car. We intentionally added more of these actions for the second stop period, and Figure 9 shows one case of opening and entering movements before we drove the car at the end of the second stop period. It clearly depicts the influence of the passenger actions in the vehicle on the acceleration and angular velocity.

Table 5. Correct rate of FFT stationary detection during the three stop periods.

Stop Period	Duration (s)	Correct (s)	Rate
1	966	965	99.9%
2	1321	1314	99.5%
3	1212	1208	99.7%
Total	3499	3487	99.7%

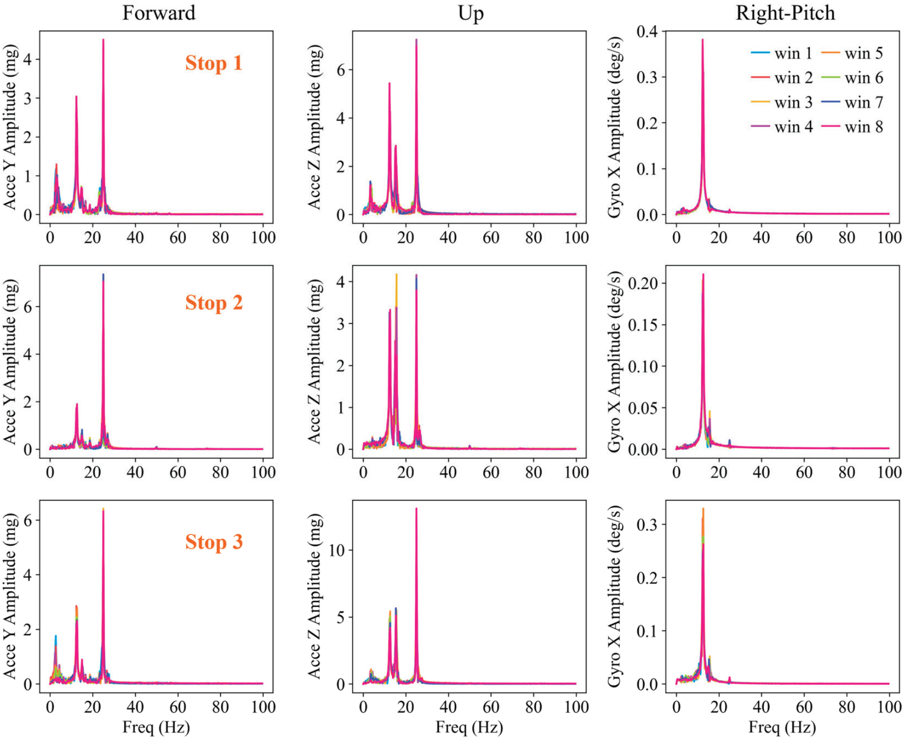


Figure 8. Ten-second examples of three-second sliding window FFT spectrum for Stop 1 (**top row**), Stop 2 (**middle row**) and Stop 3 (**Bottom row**). Each panel contains FTT results of eight consecutive windows.

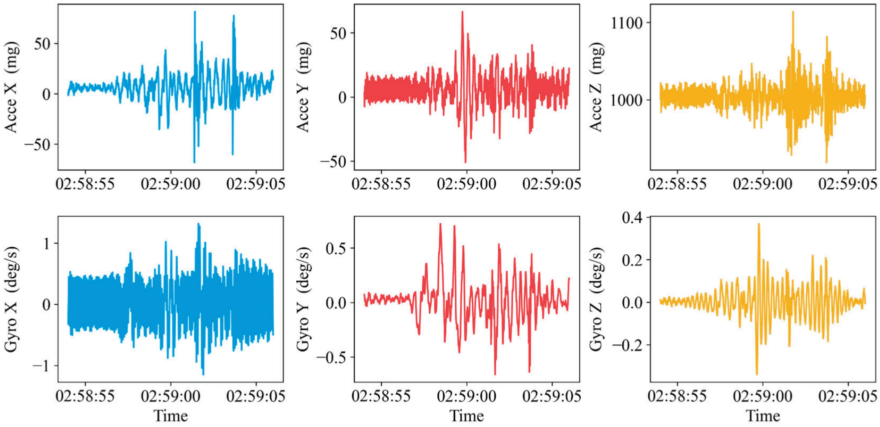


Figure 9. Raw IMU measurements of acceleration (**top row**) and angular velocity (**bottom row**) during the passenger entry into the vehicle registered in the second stop period.

4.4. ZVU-Constrained Positioning Results

In order to further evaluate the performance of the proposed FFT-based stationary detection method, the errors of ZVU-constrained positioning results and no-constraint results were calculated with respect to multi-GNSS DD RTK results as a reference. As mentioned in Section 4.3, all the GNSS observations were artificially removed, and only the

IMU was employed for computing the navigation solution during the three stop periods. The tight GNSS-PPP/IMU integration was implemented individually from the trajectory start to the end of each stationary phase within the stop periods.

The positioning error series without constraints are depicted in Figure 10. Foreseeably, the positioning errors attributable to the IMU mechanization diverge rapidly to tens of kilometers in all directions during the three stop periods. In contrast, Figure 11 presents the positioning errors produced by the ZVU-constrained IMU mechanization. The green lines indicate the start time of every stop period. It is clearly seen that the position is well constrained by the ZVU function, and all the positioning errors are converged within one decimeter.

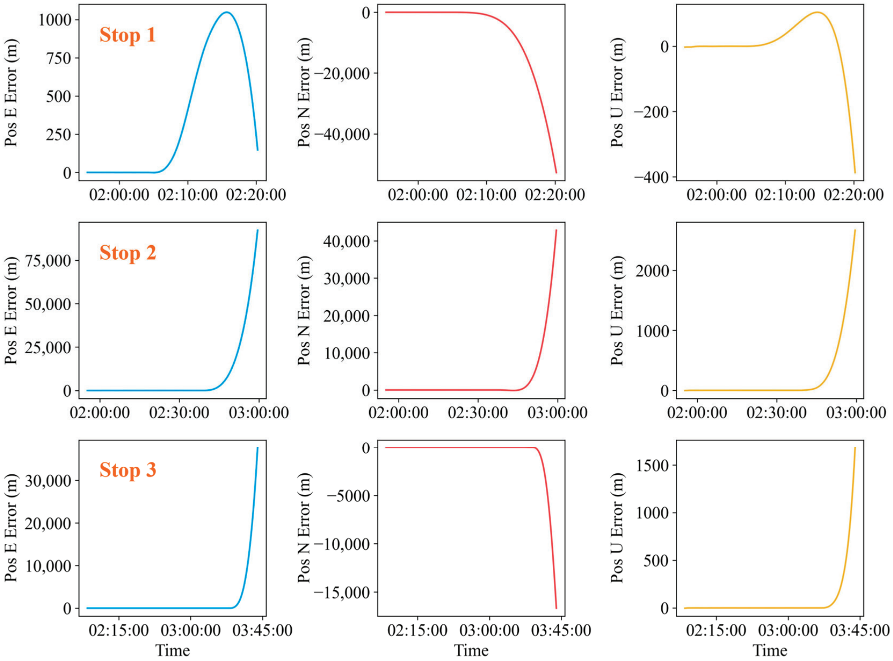


Figure 10. Divergence of positioning error in east (**left column**), north (**middle column**) and up (**right column**) during the three stop periods (in each row) with total GNSS outages.

Table 6 summarizes the position variation from the first stop epoch obtained with the ZVU in terms of the Root Mean Square (RMS), STD and maximum (Max) deviation. The RMS, STD and Max of the three stop periods in all the east, north and up directions are below 3 cm, 1 cm and 4cm, respectively. Moreover, the average RMS, STD and Max of position variation for the experimental stop periods are 1.12 cm, 0.51 cm and 2.29 cm, respectively. The statistical results illustrate that the proposed stationary detection method is efficient in supporting the ZVU in inhibiting the divergence of positioning error.

Table 6. RMS, STD and Max of position variation with respect to the initial position of each stop period.

Stop Period	East (cm)			North (cm)			Up (cm)		
	RMS	STD	Max	RMS	STD	Max	RMS	STD	Max
1	0.99	0.98	2.92	1.39	0.92	3.16	0.31	0.29	0.87
2	1.00	0.44	2.17	0.70	0.42	2.16	2.24	0.21	2.72
3	1.29	0.47	2.45	0.73	0.63	2.12	1.36	0.25	2.08
Average	1.09	0.63	2.51	0.94	0.66	2.48	1.30	0.25	1.89

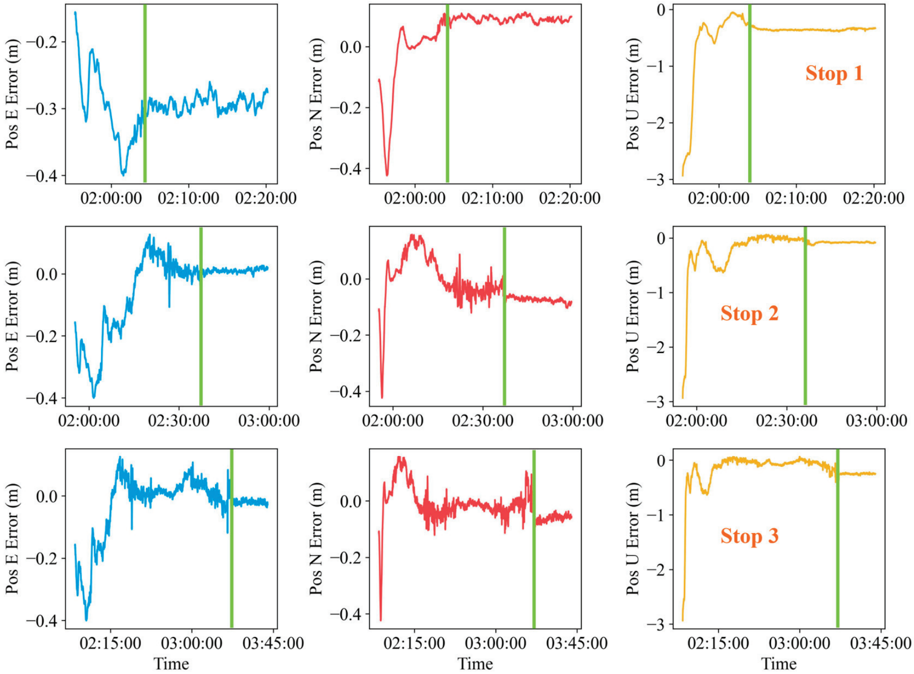


Figure 11. ZVU-constrained positioning error in east (**left column**), north (**middle column**) and up (**right column**) during the three stop periods (in each row) with total GNSS outages. The green line indicates the start time of the stop period.

5. Conclusions

The MEMS IMU is a significant piece of equipment for vehicles, allowing them to produce consecutive navigation information in an integrated system with GNSS and other sensors. MEMS IMU errors are prone to accumulate over time, and the ZVU is recognized as an efficient method for constraining the divergence of an IMU. The key to implementing ZVU is detecting the stop of the vehicle correctly. Hence, we propose a method based on FFT that has been tailored for land vehicles with idling engines. The feasibility of the proposed method was evaluated by an urban vehicular test with three stops with durations ranging from 10 to 20 min. The conclusions of our study are as follows:

The frequency peak caused by the vibration of an idling engine can be distinguished in the FFT spectra of forward/up acceleration and pitch angular velocity when the vehicle is stopped. In contrast, the vibrational peak is masked by the peaks within the 0 to 10 Hz range during the decelerating and accelerating periods. Therefore, we propose implementing FFT based on a three-second sliding window for IMU measurements per second. Then, we compare the amplitude of the vibrational frequency with the 0 to 10 Hz range frequency peak to detect if the vehicle is in a stationary condition.

To evaluate the performance of the proposed stationary detection method, we implemented the ZVU constraint under the simulated GNSS outages over three stop periods. As a result, the proposed detection method achieved an overall correct detection rate of 99.7% for the experimental stop periods. Based on the stationary detection method, the ZVU constrains the divergence of positioning error within centimeters. The results demonstrate the feasibility of the proposed stationary detection based on IMU measurements and the FFT feature of idling vibration tailored for land vehicles.

Author Contributions: Conceptualization, M.L.; methodology, M.L., V.S. and W.N.; software, M.L. and W.Z.; validation, T.X., A.R.-G. and G.X.; formal analysis, M.L.; investigation, M.L.; resources, T.X.; data curation, M.L. and W.Z.; writing—original draft preparation, M.L.; writing—review and editing, M.L., W.N., V.S., A.R.-G. and T.X.; visualization, M.L.; supervision, T.X., A.R.-G. and G.X.; project administration, T.X., W.N. and A.R.-G.; funding acquisition, T.X., W.N., A.R.-G. and V.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Key Research and Development Program of China (2020YFB0505800 and 2020YFB0505804), the Young Scholars Program of Shandong University, Weihai and Youth Innovation Team Development Project of Higher School in Shandong Province (2023KJ143), Key Laboratory of Smart Earth (KF2023YB02-11), State-funded postdoctoral researcher program (GZC20231482), the Spanish AEI/MCIN project (CNS2022-135383 and PID2022-138485OB-I00) within the European Community FEDER and industrial PhD grant number 2020 DI 108 from the Generalitat de Catalunya.

Data Availability Statement: The datasets utilized in this work are available from the corresponding author on reasonable request.

Acknowledgments: Thanks to the International GNSS Service (IGS) for providing the multi-GNSS precise orbit and clock products. Thanks to Shili Yang, Junting Wang, Long Li and Lei Zhang for their contribution to the construction of GNSS/IMU hardware. Thanks to Yongqing Lin for the driving service under the experimental requirements.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Titterton, D.; Weston, J.L.; Weston, J. *Strapdown Inertial Navigation Technology*; IET: London, UK, 2004; Volume 17.
2. Lawrence, A. *Modern Inertial Technology: Navigation, Guidance, and Control*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012.
3. Abd Rabbou, M.; El-Rabbany, A. Tightly coupled integration of GPS precise point positioning and MEMS-based inertial systems. *GPS Solut.* **2015**, *19*, 601–609. [CrossRef]
4. Li, T.; Zhang, H.; Gao, Z.; Niu, X.; El-sheimy, N. Tight Fusion of a Monocular Camera, MEMS-IMU, and Single-Frequency Multi-GNSS RTK for Precise Navigation in GNSS-Challenged Environments. *Remote Sens.* **2019**, *11*, 610. [CrossRef]
5. Kempe, V. *Inertial MEMS: Principles and Practice*; Cambridge University Press: Cambridge, UK, 2011.
6. Aggarwal, P. *MEMS-Based Integrated Navigation*; Artech House: Norwood, MA, USA, 2010.
7. Li, X.; Wang, H.; Li, S.; Feng, S.; Wang, X.; Liao, J. GIL: A tightly coupled GNSS PPP/INS/LiDAR method for precise vehicle navigation. *Satell. Navig.* **2021**, *2*, 26. [CrossRef]
8. Li, X.; Chang, H.; Wang, X.; Li, S.; Zhou, Y.; Yu, H. An Optimization-Based Tightly-Coupled Integration of PPP, INS and Vision for Precise and Continuous Navigation. *IEEE Trans. Veh. Technol.* **2023**, 1–14. [CrossRef]
9. Li, T.; Zhang, H.; Niu, X.; Gao, Z. Tightly-Coupled Integration of Multi-GNSS Single-Frequency RTK and MEMS-IMU for Enhanced Positioning Performance. *Sensors* **2017**, *17*, 2462. [CrossRef] [PubMed]
10. Groves, P.D. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 2nd ed.; Artech House: Norwood, MA, USA, 2013; p. 776.
11. Grewal, M.S.; Weill, L.R.; Andrews, A.P. *Global Positioning Systems, Inertial Navigation, and Integration*; John Wiley & Sons: Hoboken, NJ, USA, 2007.
12. Lima Filho, V.C.; Moraes, A. Modeling multifrequency GPS multipath fading in land vehicle environments. *GPS Solut.* **2020**, *25*, 3. [CrossRef]
13. Wang, X.; Li, X.; Shen, Z.; Li, X.; Zhou, Y.; Chang, H. Factor graph optimization-based multi-GNSS real-time kinematic system for robust and precise positioning in urban canyons. *GPS Solut.* **2023**, *27*, 200. [CrossRef]
14. Quan, Y.; Lau, L.; Roberts, G.W.; Meng, X.; Zhang, C. Convolutional Neural Network Based Multipath Detection Method for Static and Kinematic GPS High Precision Positioning. *Remote Sens.* **2018**, *10*, 2052. [CrossRef]
15. Grejner-Brzezinska, D.A.; Yi, Y.; Toth, C.K. Bridging GPS Gaps in Urban Canyons: The Benefits of ZUPTs. *Navigation* **2001**, *48*, 216–226. [CrossRef]
16. Li, Q.; Li, K.; Liang, W. A zero-velocity update method based on neural network and Kalman filter for vehicle-mounted inertial navigation system. *Meas. Sci. Technol.* **2023**, *34*, 045110. [CrossRef]
17. Amin, M.S.; Reaz, M.B.I.; Nasir, S.S.; Bhuiyan, M.A.S. Attitude Heading Reference System based vehicle stationary state detection. In Proceedings of the 2014 International Conference on Electrical Engineering and Information & Communication Technology, Mirpur, Pakistan, 10–12 April 2014; pp. 1–4.
18. Amin, M.S.; Reaz, M.B.I.; Nasir, S.S.; Bhuiyan, M.A.S.; Ali, M.A.M. A Novel Vehicle Stationary Detection Utilizing Map Matching and IMU Sensors. *Sci. World J.* **2014**, *2014*, 597180. [CrossRef] [PubMed]

19. Lu, S.; Gong, Y.; Luo, H.; Zhao, F.; Li, Z.; Jiang, J. Heterogeneous Multi-Task Learning for Multiple Pseudo-Measurement Estimation to Bridge GPS Outages. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 8500916. [CrossRef]
20. Ramanandan, A.; Chen, A.; Farrell, J.A.; Suvarna, S. Detection of Stationarity in an Inertial Navigation System. In Proceedings of the 23rd International Technical Meeting of the Satellite Division of the Institute-of-Navigation (ION GNSS-2010), Portland, OR, USA, 21–24 September 2010; pp. 238–244.
21. Ramanandan, A.; Chen, A.; Farrell, J.A. Inertial Navigation Aiding by Stationary Updates. *IEEE Trans. Intell. Transp. Syst.* **2012**, *13*, 235–248. [CrossRef]
22. Johnson, S.R.; Subhedar, J.W. Computer optimization of engine mounting systems. In Proceedings of the 3rd International Conference on Vehicle Structural Mechanics, Troy, MI, USA, 1 February 1979.
23. Lee, J.M.; Yim, H.J.; Kim, J.-H. Flexible Chassis Effects on Dynamic Response of Engine Mount Systems. In Proceedings of the International Conference On Vehicle Structural Mechanics & CAE, Troy, MI, USA, 4–6 April 1995.
24. Iyer, G.; Prasanth, B.; Wagh, S.; Hudson, D. Idle Vibrations Refinement of a Passenger Car. In Proceedings of the 2011 Symposium on International Automotive Technology, Pune, India, 19–21 January 2011.
25. Balla, C.K.; Naidu, S.; Ambardekar, M.N. Reduction of Idle Shake in a Small Commercial Vehicle. In Proceedings of the SAE 2015 Noise and Vibration Conference and Exhibition, Tokyo, Japan, 22–25 June 2015.
26. Qatu, M.S.; Abdelhamid, M.K.; Pang, J.; Sheng, G. Overview of automotive noise and vibration. *Int. J. Veh. Noise Vib.* **2009**, *5*, 1–35. [CrossRef]
27. Wang, X. *Vehicle Noise and Vibration Refinement*; Woodhead Publishing: Cambridge, UK, 2010. [CrossRef]
28. Kalla, B.; Patil, S.; Kumbhar, M. Idle Shake Simulation and Optimization through Digital Car Model. In Proceedings of the SAE 2015 Noise and Vibration Conference and Exhibition, Tokyo, Japan, 22–25 June 2015.
29. Heideman, M.; Johnson, D.; Burrus, C. Gauss and the history of the fast fourier transform. *IEEE ASSP Mag.* **1984**, *1*, 14–21. [CrossRef]
30. Smith, S. *Digital Signal Processing: A Practical Guide for Engineers and Scientists*; Newnes: London, UK, 2003.
31. Cooley, J.W.; Tukey, J.W. An algorithm for machine calculation of complex fourier series. *Math. Comput.* **1965**, *19*, 297–301. [CrossRef]
32. Cooley, J.; Lewis, P.; Welch, P. The finite Fourier transform. *IEEE Trans. Audio Electroacoust.* **1969**, *17*, 77–85. [CrossRef]
33. Sanz Subirana, J.; Juan Zornoza, J.M.; Hernández-Pajares, M. GNSS Data Processing. Fletcher, K., Ed.; European Space Agency, ESA Communications: Noordwijk, The Netherlands, 2013; Volume I.
34. Leick, A.; Rapoport, L.; Tatarnikov, D. *GPS Satellite Surveying*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
35. Xu, G.; Xu, Y. *GPS: Theory, Algorithms and Applications*, 3rd ed.; Springer: Berlin/Heidelberg, Germany, 2016. [CrossRef]
36. Harris, F.J. On the use of windows for harmonic analysis with the discrete Fourier transform. *Proc. IEEE* **1978**, *66*, 51–83. [CrossRef]
37. Rohling, H.; Schuermann, J. Discrete time window functions with arbitrarily low sidelobe level. *Signal Process.* **1983**, *5*, 127–138. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

A Robust Position Estimation Method in the Integrated Navigation System via Factor Graph

Sihang Quan [1], Shaohua Chen [2], Yilan Zhou [1], Shuai Zhao [1], Huizhu Hu [1,2,*] and Qi Zhu [2]

[1] State Key Laboratory of Modern Optical Instrumentation, Zhejiang University, Hangzhou 310058, China; 12130067@zju.edu.cn (S.Q.); naliyuohz@zju.edu.cn (Y.Z.); 12030011@zju.edu.cn (S.Z.)
[2] Institute of Intelligent Perception, Zhejiang Lab, Hangzhou 311500, China; chenshaohua@zhejianglab.com (S.C.); riggda@zju.edu.cn (Q.Z.)
* Correspondence: huhuizhu2000@zju.edu.cn; Tel.: +0571-87952781

Abstract: Achieving higher accuracy and robustness stands as the central objective in the navigation field. In complex urban environments, the integrity of GNSS faces huge challenges and the performance of integrated navigation systems can be significantly affected. As the proportion of faulty measurements rises, it can result in both missed alarms and false positives. In this paper, a robust method based on factor graph is proposed to improve the performance of integrated navigation systems. We propose a detection method based on multi-conditional analysis to determine whether GNSS is anomalous or not. Moreover, the optimal weight of GNSS measurement is estimated under anomalous conditions to mitigate the impact of GNSS outliers. The proposed method is evaluated through real-world road tests, and the results show the positioning accuracy of the proposed method is improved by more than 60% and the missed alarm rate is reduced by 80% compared with the traditional algorithms.

Keywords: integrated navigation system; factor graph; robustness; GNSS outliers

Citation: Quan, S.; Chen, S.; Zhou, Y.; Zhao, S.; Hu, H.; Zhu, Q. A Robust Position Estimation Method in the Integrated Navigation System via Factor Graph. *Remote Sens.* **2024**, *16*, 562. https://doi.org/10.3390/rs16030562

Academic Editors: Giuseppe Casula, Zhetao Zhang, Guorui Xiao, Zhixi Nie and Vagner Ferreira

Received: 5 January 2024
Revised: 26 January 2024
Accepted: 27 January 2024
Published: 31 January 2024

1. Introduction

As the main source of global referenced positioning for Intelligent Transportation Systems (ITS), the Global Navigation and Positioning System (GNSS) can provide all-weather position information in outdoor scenarios [1]. Unfortunately, the positioning accuracy of GNSS is greatly affected by the environment. Interference, spoof, and outage occur frequently due to the multipath effects and non-line-of-sight (NLOS) receptions [2], which limit positioning accuracy. In contrast, the inertial navigation system (INS) can autonomously provide position information at a high output frequency with little dependence on the external environment but the navigation error will accumulate over time. GNSS/INS integrated navigation systems [3,4] make use of their advantages, which can provide high-precision and high-frequency navigation results. The popular existing GNSS/INS integration methods can be divided into filter frameworks and optimization frameworks.

Extended Kalman filter (EKF) based on filtering frameworks has been widely used to integrate different sensors due to its simple calculations and real-time performance [5]. However, from the mathematical interpretation, EKF can only iterate in a single step based on the first-order Markov assumption, which does not make full use of the historical information and is unable to perform correct repeated linearization. To solve this problem, unscented Kalman filter [6], and cubature Kalman filter [7] are proposed to improve the adaptability of nonlinear systems through more precise modeling. Furthermore, the iterated Kalman filter [8] is proposed to achieve multiple iterations, which significantly mitigates the linearization error. However, all these solutions did not fully utilize historical information. In addition, adding and removing sensors in filtering frameworks requires reconfiguration of the system, which increases complexity and time consumption.

The recently proposed factor graph optimization (FGO) [9] scheme provides a new view for multi-sensor fusion. Factor graphs have been widely used in Simultaneous Localization and Mapping (SLAM) to integrate diverse sensor measurements. In recent years, FGO has attracted much attention in the field of integrated navigation due to the advantages of multiple iterations, re-linearization, and the use of historical information [10]. The factor-graph-based multi-source fusion algorithms have been validated in simulations or experiments on several platforms, including aircraft, vehicles, and ships [11–14], which indicates that the FGO-based method is superior to the EKF-based method.

Although the potential of factor graph algorithms for positioning has been explored, it has also been shown that improved methods based on FGO are better adapted to the error distribution of multi-sensor fusion [15]. However, in complex environments, such as urban canyons, multipath, and NLOS, interference occurs frequently, which creates a greater impact on the accuracy of integrated navigation systems. A lot of research has been conducted on how to obtain more reliable and accurate GNSS positioning results, including detecting cycle slips by Melbourne-Wübbena [16] and geometry-free [17], which have significantly improved the data quality control. Nevertheless, there are still roughness and model-free errors in the observation that affect the localization accuracy.

Typically, methods to solve this problem can be classified into two categories: one is probabilistic statistics-based fault detection and isolation, which usually employ binary hypothesis tests based on residuals to detect faults then isolate them using the plug-and-play nature of the factor graph framework. The work described in [18] proposed a GNSS pseudorange fault detection method, which employs a chi-square test to detect faults and isolate them directly using factor graph. In [19], an iRAIM method is proposed where faults are detected in a fixed time window. The study in [20] combines the advantages of the sliding window and chi-square to allow faults to be estimated repeatedly to improve GNSS integrity. Similarly, the work in [21] uses the history information within a sliding window to fit new observations, reducing the effect of coarseness. However, all these solutions have a certain degree of time delay [22]. The other category is robust model-aided fault suppression. These strategies usually use a robust weight function to adjust the cost function to reduce the weight of the fault measurements, including the Huber kernel function [23], switchable constraints [24], dynamic covariance [25], etc. Wei et al. [26] proposed an improved factor graph fusion algorithm with enhanced robustness, which achieves the weight assignment and robust adjustment before the fusion of sensors, effectively reducing the adverse effect of sensor failures on the navigation results. Unfortunately, the weight function depends on the confidence distance and is restricted to [0, 1], which is not suitable for least squares approaches that require a continuous domain [27]. Yi et al. [28] proposed a robust loss function through parameter learning using a differentiable optimization. They use an end-to-end approach for learning state estimator modeled as the factor-graph-based smoother, which has a significant improvement over existing baselines. However, the use of neural networks for parameter estimation introduces a large computational effort. Nam et al. [29] built a robust adaptive state estimation framework based on the type-2 fuzzy inference system, which can learn the uncertainty by using particle swarm optimization, improving the robustness of the system. However, the increase in robustness usually implies a loss in accuracy; hence, the obtained solution is not optimal.

To overcome the limitations above, we propose an innovative IMU/GNSS/ODO loosely coupled integrated navigation framework based on factor graph. Firstly, inspired by the work in [30,31], to reduce the Inertial Measurement Unit (IMU) drift, an odometer (ODO) is derived to provide auxiliary measurements into the graph by constructing an IMU-ODO preintegration factor. Subsequently, we put forward a GNSS anomaly detection method based on multi-conditional analysis with a weighted fusion algorithm for adaptive covariance estimation. The original weighting estimate is used when the detection is normal and the adaptive covariance estimation is used for weighting when the test is anomalous, reducing the negative influence of faults. In addition, the impact of the delay in parameter estimation on the system accuracy is reduced by autonomously switching between the two weighting modes. The main contributions of the article are as follows:

- An IMU/GNSS/ODO integrated navigation framework based on factor graph is described. We accommodate the key parameters of each sensor and add ODO into preintegration to construct the IMU-ODO preintegration factor, which reduces the drift of IMU.
- To enhance the accuracy and robustness, an adaptive weighting estimation (AWE) algorithm is proposed, which can estimate the state and covariance simultaneously. We derive the principle of AWE from the maximum a posteriori (MAP) perspective.
- We put forward a GNSS anomaly detection method based on multi-conditional analysis. Through anomaly detection, the system achieves autonomous switching between two modes, original weighting estimation or adaptive weighting estimation, significantly reducing the missed alarm rate and time delays in fault recovery due to parameter estimation.

The remainder of this paper is organized as follows. Section 2 introduces the factor graph algorithm and the formulation in the IMU/GNSS/ODO integrated navigation system. Section 3 presents a robust factor graph optimization based on adaptive weighting estimation with multi-conditional analysis. The experiment results and discussion are presented to validate the effectiveness of the proposed algorithms in Section 4. Finally, the conclusion of this study is composed.

2. Factor Graph Algorithm and the Formulation in IMU/GNSS/ODO Integrated Navigation System

2.1. MAP Estimation and Factor Graph Algorithm

Factor graph is a bipartite undirected graphical model based on Bayesian networks [32], which encodes the links between variable nodes and measurements. The factor graph model solves the joint probability distribution of multivariate global functions based on the MAP estimation theory [33] to achieve information fusion.

The location issue can be described as an MAP problem:

$$\hat{\boldsymbol{X}} = \underset{\boldsymbol{X}}{\operatorname{argmax}} P(\boldsymbol{X}|\boldsymbol{Z}, \boldsymbol{u}) \tag{1}$$

where $\hat{\boldsymbol{X}}$ denotes the optimal estimation of the navigation-state variable and $\boldsymbol{X}$ denotes the set of all navigation states. $\boldsymbol{Z}$ represents all the measurements received by the system from the beginning. $\boldsymbol{u}$ is the motion excitation or the output of the motion sensors and $\boldsymbol{u}_k \in \boldsymbol{u}$.

Since the inputs $\boldsymbol{u}_k$ and measurements z_k at each moment are independent of each other, according to Bayes' law, the joint probability distribution function in Equation (1) can be factorized as:

$$\hat{\boldsymbol{X}} = \underset{\boldsymbol{X}}{\operatorname{argmax}} \prod_k P(z_k|\boldsymbol{X}_k) \prod_k P(\boldsymbol{X}_k|\boldsymbol{X}_{k-1}, \boldsymbol{u}_k) \tag{2}$$

where z_k is the measurements at the time t_k. Equation (2) can be expressed as a product of individual factors. The optimal solution can be obtained by constructing and solving factor graphs.

A factor graph $\boldsymbol{G}$=$(\boldsymbol{F},\boldsymbol{V},\boldsymbol{\varepsilon})$ [32,34] represents the joint probability distribution function of random variables where $\boldsymbol{V}$ represents a set of variable nodes encoded by the navigation state, $\boldsymbol{F}$ is a set of factor nodes, and $\boldsymbol{\varepsilon}$ represents a set of all edges connecting nodes. It consists of nodes and edges; nodes are divided into factor nodes $f_j \in \boldsymbol{F}$ and variable nodes $\boldsymbol{X}_k \in \boldsymbol{V}$. The edges connecting the two nodes represent the error function $\boldsymbol{e}_{k,j}(\cdot) \in \boldsymbol{\varepsilon}$, and each edge corresponds to a measurement z_k. The factor graph is defined as:

$$G(\boldsymbol{X}) = \prod_k \boldsymbol{f}_k(\boldsymbol{X}_k) \tag{3}$$

Assuming that the measurements are consistent with a Gaussian model [35]:

$$\boldsymbol{f}_k(\boldsymbol{X}_k) \propto \exp\left(-\frac{1}{2}\|\boldsymbol{e}(\boldsymbol{X}_k, z_k)\|^2_{\boldsymbol{\Sigma}_k}\right) = \exp\left(-\frac{1}{2}\|h_k(\boldsymbol{X}_k) - z_k\|^2_{\boldsymbol{\Sigma}_k}\right) \tag{4}$$

where $\boldsymbol{e}(\cdot)$ is the error function, which defines the error between the measurements z_k and the observation $h_k(\cdot)$, $\|\boldsymbol{e}\|^2_{\boldsymbol{\Sigma}_k}$ denotes the Mahalanobis distance, while $\boldsymbol{\Sigma}_k$ is the measurement noise covariance matrix, and $h_k(\cdot)$ denotes some known functions about state vectors. Based on Equations (3) and (4), the MAP estimation can be transformed into a nonlinear least squares problem:

$$\begin{aligned}\hat{\boldsymbol{X}} &= \underset{\boldsymbol{X}}{\operatorname{argmin}}\sum_k \|h_k(\boldsymbol{X}_k) - z_k\|^2_{\boldsymbol{\Sigma}_k} \\ &= \underset{\boldsymbol{X}}{\operatorname{argmin}}\sum_k \boldsymbol{e}(\boldsymbol{X}_k, z_k)^{\mathrm{T}}\boldsymbol{\Sigma}_k{}^{-1}\boldsymbol{e}(\boldsymbol{X}_k, z_k) \\ &= \underset{\boldsymbol{X}}{\operatorname{argmin}}\ g(\boldsymbol{X}_k)\end{aligned} \tag{5}$$

Up until now, the inference of the factor graph is to find the optimal estimation $\hat{\boldsymbol{X}}$ by minimizing the objective function $g(\boldsymbol{X}_k)$ above.

The optimal estimation of $\hat{\boldsymbol{X}}$ can be solved by Gauss–Newton (G-N), Levenberg–Marquardt (L-M), dog-leg iteration [36], etc. To solve the least squares problem, we make the gradient of the objective function decrease by giving initial values and continuously updating the optimization variables through iterations, which transforms the problem of solving for a derivative of 0 into a problem of solving for a falling increment $\Delta\boldsymbol{X}$. In addition, the incremental smoothing algorithm [14,37] is used to reduce the computational effort. Through the linearization process by nonlinear optimization methods, the corrected $\Delta\boldsymbol{X}$ is obtained based on the initial estimate. According to L-M iteration, the increment $\Delta\boldsymbol{X}$ meets the following equation [38]:

$$\Delta\hat{\boldsymbol{X}} = \underset{\Delta\boldsymbol{X}}{\operatorname{argmin}}\left\|\boldsymbol{J}(\boldsymbol{X}_k)^T\Delta\boldsymbol{X} + \boldsymbol{e}(\boldsymbol{X}_k, z_k)\right\|^2,\ s.t.\ \|\ \boldsymbol{D\Delta X}\ \|^2 \le \rho \tag{6}$$

where $\boldsymbol{J}(\boldsymbol{X}_k)$ represents the Jacobian matrix, $\boldsymbol{D}$ donates the coefficient matrix and usually takes the unit matrix $\boldsymbol{I}$, while ρ is the radius of the trust domain.

Solving and updating $\Delta\boldsymbol{X}$ requires upper triangular decomposition of $\boldsymbol{J}(\boldsymbol{X}_k)$ by QR or Cholesky. Solving for (6) is equivalent to solving the incremental equation [34]:

$$(\boldsymbol{J}^T\boldsymbol{J} + \lambda\boldsymbol{I})\Delta\boldsymbol{X}_k = -\boldsymbol{J}\cdot\boldsymbol{e}(\boldsymbol{X}_k, z_k) \tag{7}$$

where λ is the Lagrange operator. The Jacobi matrix is equivalent to linearizing the factor graph, and its block structure determines the structure of the factor graph. The sparsity of the matrix is maintained by determining a specific order of variable elimination [39], and each update will only compute a small portion of the topology that has changed, effectively improving the real-time performance of the algorithm.

2.2. Factor Graph Formulation in Integrated Navigation System

From the principle of the factor graph, it can be noted that the selection of system-state variables and the establishment of error functions are the main factors affecting the performance of the algorithm. Considering the key parameters of each sensor and adding ODO into preintegration, the factor formulation for measurement models is deduced.

A. Formulation

During the navigation process, the parameters of each sensor change over time [40]. Accuracy can be improved by expanding the dimension of the state variables and adding the key parameters into the optimized estimation. Considering the ODO scale factor error, the state variables of the integrated INS/GNSS/ODO navigation system are selected as:

$$\boldsymbol{X}(k) = \begin{bmatrix} (\boldsymbol{p}^{w}_{wb_k})^T & (\boldsymbol{v}^{w}_{wb_k})^T & (\boldsymbol{q}^{w}_{wb_k})^T & (\boldsymbol{\varepsilon}_{\mathrm{g}_k})^T & (\nabla_{a_k})^T & c_k \end{bmatrix} \tag{8}$$

where the state $\boldsymbol{X}(k)$ includes the position $\boldsymbol{p}^{w}_{wb_k}$, velocity $\boldsymbol{v}^{w}_{wb_k}$, and attitude $\boldsymbol{q}^{w}_{wb_k}$ in the world frame (w-frame), along with the gyroscope biases $\boldsymbol{\varepsilon}_{\mathrm{g}_k}$, accelerometer biases ∇_{a_k}, and ODO scale factor c_k. b_k is the body frame (b-frame) at k.

B. IMU-ODO Preintegration Factor

Odometer information is typically used as the forward speed to provide an aid; however, in reality, the raw measurement information of an odometer is the mileage, and obtaining an accurate auxiliary speed relies on accurate motion modeling. The preintegration method is utilized to convert the odometer mileage into displacement, forming the IMU-ODO preintegration model instead of the traditional IMU preintegration model.

The MEMS-INS mechanization [41] can be expressed as:

$$\begin{aligned} \boldsymbol{v}^{w}_{wb_k} &= \boldsymbol{v}^{w}_{wb_{k-1}} + \boldsymbol{C}^{w}_{b_{\mathrm{k}-1}} \boldsymbol{v}^{b_k}_{b_{k-1}} + \boldsymbol{g}^{w} \Delta t_k \\ \boldsymbol{p}^{w}_{wb_k} &= \boldsymbol{p}^{w}_{wb_{k-1}} + \boldsymbol{v}^{w}_{wb_{k-1}} \Delta t_k + \tfrac{1}{2} \boldsymbol{g}^{w} \Delta t_k^2 + \tfrac{1}{2} \boldsymbol{C}^{w}_{b_{k-1}} \boldsymbol{v}^{b_k}_{b_{k-1}} \Delta t_k \\ \boldsymbol{q}^{w}_{wb_k} &= \boldsymbol{q}^{w}_{wb_{k-1}} \otimes \boldsymbol{q}^{b_k}_{b_{k-1}} \end{aligned} \tag{9}$$

where $\boldsymbol{C}^{w}_{b_{\mathrm{k}-1}}$ denotes the rotation matrix from the b-frame to the w-frame; $\boldsymbol{g}^{w}$ is the gravity in the w-frame; $\boldsymbol{v}^{w}_{wb_{k-1}}$ and $\boldsymbol{p}^{w}_{wb_{k-1}}$ are the velocity and position of the b-frame relative to the w-frame projected on the w-frame; $\boldsymbol{q}^{w}_{wb_k}$ is the quaternion; and $\boldsymbol{q}^{b_k}_{b_{k-1}}$ and $\boldsymbol{v}^{b_k}_{b_{k-1}}$ are the increments of attitude and velocity from the $k-1$ to the k, respectively:

$$\begin{aligned} \boldsymbol{v}^{b_k}_{b_{k-1}} &\approx \Delta \boldsymbol{v}^{b}_{f,k} + \tfrac{1}{2} \Delta \boldsymbol{\theta}_k \times \Delta \boldsymbol{v}^{b}_{f,k} \\ \boldsymbol{q}^{b_k}_{b_{k-1}} &\approx \begin{bmatrix} \cos \left\| \frac{1}{2} \Delta \boldsymbol{\theta}_k \right\| \\ \frac{\sin \left\| \frac{1}{2} \Delta \boldsymbol{\theta}_k \right\|}{\left\| \Delta \boldsymbol{\theta}_k \right\|} \Delta \boldsymbol{\theta}_k \end{bmatrix} \\ \Delta \boldsymbol{v}^{b}_{f,k} &= \int_{t_{k-1}}^{t_k} (\boldsymbol{f}^{b} - \nabla_{a_k} - \boldsymbol{n}_a) dk \\ \Delta \boldsymbol{\theta}_k &= \int_{t_{k-1}}^{t_k} \left[\boldsymbol{\omega}^{b} - \boldsymbol{\varepsilon}_{g_k} - \boldsymbol{n}_g \right] dk \end{aligned} \tag{10}$$

where $\boldsymbol{f}^{b}$ and $\boldsymbol{\omega}^{b}$ are the specific force and angular rate in the b-frame. $\Delta \boldsymbol{v}^{b}_{f,k}$ and $\Delta \boldsymbol{\theta}_k$ are the velocity and angle of the integration of accelerometer and gyroscope outputs. $\boldsymbol{n}_a$ and $\boldsymbol{n}_g$ represent the gyroscope and accelerometer white noise, respectively.

When ODO observations are introduced, considering the nonholonomic constraints, the velocity from the b-frame to the vehicle co-ordinate (v-frame) can be expressed as follows [30]:

$$\boldsymbol{v}^{w}_{wv_k} = \boldsymbol{v}^{w}_{wb_k} + \boldsymbol{C}^{w}_{b_k} \left[\boldsymbol{\omega}^{b_k}_{wb_k} \times \right] \boldsymbol{I}^{b}_{odo} \tag{11}$$

where $\boldsymbol{I}^{b}_{odo}$ donates the lever arm for the odometer.

The mileage increment derived from velocity is:

$$\boldsymbol{s}^{b_k}_{b_{k-1}} \approx \boldsymbol{C}^{b}_{v} \boldsymbol{s}^{v_k}_{v_{k-1}} - \boldsymbol{C}^{b_k}_{b_{k-1}} \boldsymbol{I}^{b}_{\mathrm{odo}} + \boldsymbol{I}^{b}_{\mathrm{odo}} \tag{12}$$

where $\boldsymbol{C}^{b}_{v}$ is the mounting angle of the IMU, $\boldsymbol{s}^{v_k}_{v_{k-1}}$ is the increment of ODO, denoted as $\boldsymbol{s}^{v_k}_{v_{k-1}} = s^{v}(1 + c_k)$ and $\boldsymbol{s}^{v}$ represents the output of the ODO, and $\boldsymbol{C}^{b_k}_{b_{k-1}}$ is the direction cosine matrix of $\boldsymbol{q}^{b_k}_{b_{k-1}}$.

Therefore, the position can be updated by the IMU and ODO, represented as:

$$\boldsymbol{p}^w_{wb_k} = \boldsymbol{p}^w_{wb_{k-1}} + \boldsymbol{C}^w_{b_{k-1}}[\boldsymbol{C}^b_v \boldsymbol{s}^{v_k}_{v_{k-1}} + (1 - \boldsymbol{C}^{b_k}_{b_{k-1}})\boldsymbol{l}^b_{\text{odo}}] \tag{13}$$

According to (9) and (13), the IMU-ODO preintegration factor is constructed as:

$$f^{\boldsymbol{IMU-ODO}}(\boldsymbol{X}_{k+1}, \boldsymbol{X}_k, \boldsymbol{\varepsilon}_{g_k}, \nabla_{a_k}, c_k) \triangleq d\left(\boldsymbol{X}_{k+1} - h^{INS}_k(\boldsymbol{X}_k, \boldsymbol{\varepsilon}_{g_k}, \nabla_{a_k}, c_k, z_k)\right) \tag{14}$$

The equivalent IMU factor [9] updates in the b-frame, which leads to large errors when the motion changes rapidly. Therefore, the classical INS mechanization [42,43] is used for updating during the preintegration period, $\boldsymbol{X}^{\boldsymbol{INS-ODO}}_{\boldsymbol{k}}$ is the estimated value of the navigation state variable obtained from the IMU-ODO mechanization, and INS-ODO factor node is:

$$f^{\boldsymbol{INS-ODO}}(\boldsymbol{X}_k) \triangleq d\left(\boldsymbol{X}_k - \boldsymbol{X}^{\boldsymbol{INS-ODO}}_k\right) \tag{15}$$

Suppose that the bias $\boldsymbol{k}_k = (\boldsymbol{\varepsilon}_{g_k}, \nabla_{a_k}, c_k)$, the bias node can be expressed as:

$$f^{\boldsymbol{bias}}(\boldsymbol{k}_k, \boldsymbol{k}_{k-1}) \triangleq d(\boldsymbol{k}_k - u(\boldsymbol{k}_k)) \tag{16}$$

where $u(\boldsymbol{k}_k)$ is the stochastic wandering model with bias parameters.

Since the bias of the inertial navigation system does not change significantly in a short period, it can be inserted into the factor graph model at a lower frequency, which can reduce the amount of computation without loss of accuracy.

C. GNSS Positioning Factor

The GNSS positioning results and covariance $\boldsymbol{\Sigma}^{\boldsymbol{GNSS}}_i$ can be obtained from the GNSS receiver. Hence, the measurement equation considering the lever-arm effect can be expressed as:

$$\boldsymbol{p}^w_{\boldsymbol{GNSS},k} = \boldsymbol{p}^w_{wb_k} + \boldsymbol{R}^w_{b_k}\boldsymbol{l}^b + \boldsymbol{n}^{\boldsymbol{GNSS}}_k \tag{17}$$

where $\boldsymbol{p}^w_{\boldsymbol{GNSS},k}$ is the GNSS position in the w-frame, $\boldsymbol{n}^{\boldsymbol{GNSS}}_k$ is the measurement position noise, and $\boldsymbol{l}^b$ denotes the lever-arm error of GNSS antenna. $\boldsymbol{R}^w_{b_k}$ is the rotation matrix from the b-frame to the w-frame. Then, the GNSS factor node can be expressed as:

$$f^{\boldsymbol{GNSS}}(\boldsymbol{p}^w_{wb_k}) \triangleq d\left(\boldsymbol{p}^w_{wb_k} + \boldsymbol{R}^w_{b_k}\boldsymbol{l}^b - \boldsymbol{p}^w_{\boldsymbol{GNSS},k}\right) \tag{18}$$

D. IMU-ODO/GNSS factor graph model

In summary, we derive the IMU-ODO preintegration model and the GNSS measurement model to form the IMU-ODO/GNSS integrated navigation system. When the GNSS measurement information arrives, the corresponding factor node is connected to the INS node constructed by IMU-ODO. The factor graph model of the whole navigation system is shown in Figure 1.

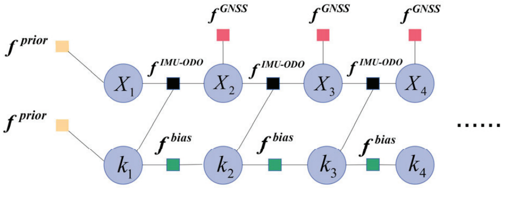


Figure 1. Factor graph model of IMU/GNSS/ODO.

We minimize the sum of the prior and Mahalanobis distances for all residuals to obtain the MAP estimate:

$$\min_{\boldsymbol{X}} \left\{ \begin{array}{c} \left\| \boldsymbol{r}_0 - \boldsymbol{h}_0 \boldsymbol{X} \right\|^2 + \sum\limits_{k \in [1,n]} \left\| \boldsymbol{r}_{\boldsymbol{IMU-ODO}} \left(z_{k-1,k}^{IMU-ODO}, \boldsymbol{X} \right) \right\|_{\boldsymbol{\Sigma}_{k-1,k}^{IMU-ODO}}^2 \\ + \sum\limits_{i \in [0,m]} \left\| \boldsymbol{r}_{\boldsymbol{GNSS}} \left(z_i^{GNSS}, \boldsymbol{X} \right) \right\|_{\boldsymbol{\Sigma}_i^{GNSS}}^2 \end{array} \right\} \tag{19}$$

where $\{\boldsymbol{r}_0, \boldsymbol{h}_0\}$ represents the priori information and $\boldsymbol{r}_{\boldsymbol{IMU-ODO}}$ and $\boldsymbol{r}_{\boldsymbol{GNSS}}$ are the residual of the IMU-ODO preintegrating factors and GNSS factors, respectively. m, n are the number of GNSS and IMU-ODO preintegrating factors, respectively. Ceres solver is used to solve this problem.

3. Robust Factor Graph Optimization Based on Adaptive Weighting Estimation with Multi-Conditional Analysis

In this section, we will introduce a novel GNSS anomaly detection method and an adaptive covariance estimation algorithm to enhance accuracy and robustness. An overview of the method proposed is shown in Figure 2. The FGO integrates multi-sensor measurements aided by the weighting from the switching mode options. The whole process is called switching adaptive weighting estimation factor graph optimization (SAWE-FGO) in this paper.

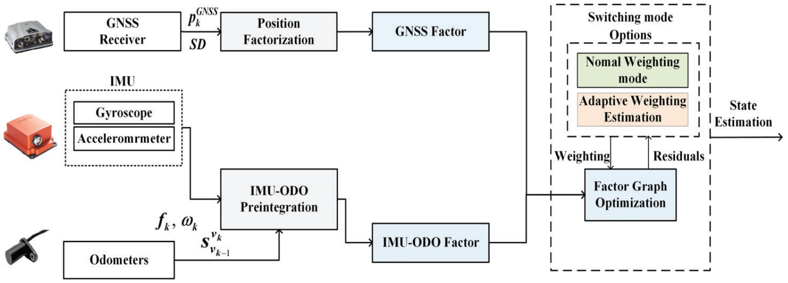


Figure 2. Overview of the proposed IMU/GNSS/ODO integration via SAWE-FGO. The light blue area represents the factor graph optimization part. The green and orange area denotes the adaptive weighting estimation based on switching modes.

3.1. GNSS Anomaly Detection Based on Multi-Conditional Analysis

Based on Equation (17), the residual of the GNSS factor can be obtained as follows:

$$\boldsymbol{r}_{\boldsymbol{GNSS}} \left(z_k^{GNSS}, \boldsymbol{X} \right) = \boldsymbol{p}_{\mathrm{wb}_k}^{\mathrm{w}} + \boldsymbol{R}_{\mathrm{b}_k}^{\mathrm{w}} \boldsymbol{l}^b - \boldsymbol{p}_{\boldsymbol{GNSS},k}^{\mathrm{w}} \tag{20}$$

Assuming that the observation model and the IMU-ODO model do not contain errors, this yields the following conditional density on the measurement z_k:

$$P(z_k | \boldsymbol{X}_k) = \mathcal{N}(z_k; h_k(\boldsymbol{X}_k), \boldsymbol{\Sigma}_0) = \frac{1}{(2\pi)^{k/2} \sqrt{|\boldsymbol{\Sigma}_0|}} \exp\left\{ -\frac{1}{2} \| h_k^{GNSS}(\boldsymbol{X}_k) - z_k \|_{\boldsymbol{\Sigma}_0}^2 \right\} \tag{21}$$

If the GNSS equipment fails, the statistical nature of the measurement noise changes and the mean of the residual $\boldsymbol{r}_{\boldsymbol{GNSS}}(k)$ will no longer be equal to 0, as described below.

According to the binary hypothesis to $\boldsymbol{r}_{\boldsymbol{GNSS}}(k)$, H_0 indicates that there are no faults:

$$E\{\boldsymbol{r}_{\boldsymbol{GNSS}}(k)\} = 0, \; E\left\{ \boldsymbol{r}_{\boldsymbol{GNSS}}(k) \boldsymbol{r}_{\boldsymbol{GNSS}}(k)^T \right\} = \boldsymbol{\Sigma}_0 \tag{22}$$

H_1 indicates a malfunction:

$$E\{\boldsymbol{r_{GNSS}}(k)\} = \mu,\ E\left\{(\boldsymbol{r_{GNSS}}(k) - \mu)(\boldsymbol{r_{GNSS}}(k) - \mu)^T\right\} = \boldsymbol{\Sigma}_0 \tag{23}$$

Since the residuals have three dimensions, latitude, longitude, and height, the same judgment criterion can be used in each dimension. The residual criterion in one dimension is as follows:

$$\begin{cases} \text{ }if\ r_{\boldsymbol{GNSS}}(k) > r_1, \text{ fault} \\ if\ r_{\boldsymbol{GNSS}}(k) \le r_1, \text{ nomal} \end{cases} \tag{24}$$

where r_1 is the threshold for the judgment. $\boldsymbol{r_{GNSS}}(k)$ represents the residual between the measured value and the predicted value; thus, the larger $\boldsymbol{r_{GNSS}}(k)$ is, the less reliable the measurement information is. The selection of r_1 will affect the results of fault detection. If r_1 is too large, it will lead to a high rate of missed alarms. Inversely, if r_1 is too small, it will not be able to meet the statistical characteristics. The threshold value can be set according to the accuracy and performance of different sensors, but the basic criterion is to set it between the maximum probability value of the independent variable and the mean of the chi-square distribution. In this paper, we have chosen the value of r_1 to be 0.2.

Using the Mahalanobis distance of residuals, the chi-square test is simple in operation and small in computation compared to sequential methods with the same number of samples. However, this approach introduces missed alarms and time delays for slow-change faults. Therefore, the GNSS and IMU-ODO preintegrated incremental error criterion is provided to assist the residuals in determining whether there is an anomaly in the standard deviations (SD) of latitude, longitude, and altitude measurement.

During a GNSS epoch, the IMU-ODO preintegration is recursed in increments, with mileage increments $\boldsymbol{S}_{k-1}^{k}$. Obtained from the IMU-ODO mechanization, $\boldsymbol{X}_{\mathrm{k-1}}^{\boldsymbol{INS-ODO}}$ and $\boldsymbol{X}_{\mathrm{k}}^{\boldsymbol{INS-ODO}}$ are the navigation-state estimates for two neighboring moments without the GNSS fusion. $\boldsymbol{L}_{k-1}^{k}$ represents displacement increments from the GNSS result of the previous epoch. Since relative changes are described, to accurately reflect the incremental changes, we align $\boldsymbol{GNSS}_{k-1}$ with $\boldsymbol{X}_{\mathrm{k-1}}^{\boldsymbol{INS-ODO}}$ in position and use vector decomposition in the b-frame, as shown in Figure 3.

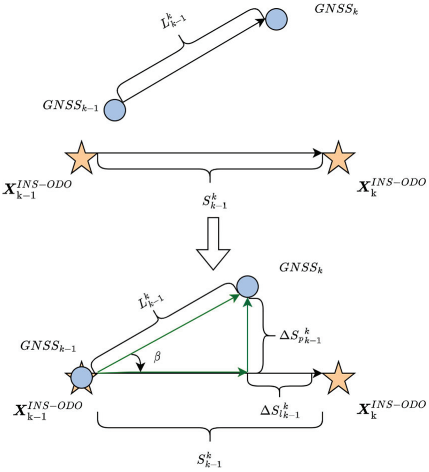


Figure 3. Vector decomposition representation of GNSS and IMU-ODO preintegrated incremental error.

The longitudinal and lateral incremental error can be expressed as:

$$\begin{array}{l} \Delta \boldsymbol{S}_{lk-1}^{k} = \boldsymbol{S}_{k-1}^{k} - \boldsymbol{L}_{k-1}^{k} \cos \beta \\ \Delta \boldsymbol{S}_{pk-1}^{k} = \boldsymbol{L}_{k-1}^{k} \sin \beta \end{array} \tag{25}$$

Thus, the incremental error of IMU-ODO preintegration with GNSS is:

$$\Delta S_{k-1}^{k} = \sqrt{\left(\Delta \boldsymbol{S}_{p}{}_{k-1}^{\ k}\right)^{2} + \left(\Delta \boldsymbol{S}_{l}{}_{k-1}^{\ k}\right)^{2}} \tag{26}$$

The incremental error criterion is:

$$\begin{cases} if\ \Delta S_{k-1}^{k} > s_1, \text{ fault} \\ if\ \Delta S_{k-1}^{k} \le s_1, \text{ nomal} \end{cases} \tag{27}$$

Under normal GNSS operation, the displacement increments of GNSS and the mileage increment of IMU-ODO are in the same trend for a short period and the incremental error between them is very small. When the GNSS is abnormal, the GNSS increment will be much larger than the IMU-ODO mileage increment, which generates a large incremental error. The threshold s_1 is set to the maximum possible probability value of the independent variable; here, we choose 0.45.

The detection method combines residual judgment, SD judgment, GNSS, and IMU-ODO incremental error judgment. It is specifically divided into four categories as Table 1 shown.

Table 1. Combined failure diagnosis rule.

Case No.	Residual	SD	Incremental Error	Diagnosis Result	Mode Judgment Result
1	T	T	/	T	T
2	F	F	/	F	T
3	T	F	/	F	F
4	F	T	F	F	F
			T	T	T

In cases 1 or 2, the result is judged to be a normal condition when residual and SD methods detect it identically. In case 3, the combination of judgments has a lower probability because false positives are small probability events in navigation systems. In case 4, which is a priority for us, the SD shows GNSS signals are normal at this moment, while the residual method displays exceptions, known as missed alarms of SD. The original SD values become implausible under this condition. Thus, using the original SD values as covariates will increase the percentage of fault measurement in integrated navigation systems. Since the residual is more sensitive to faults, extra time is required for the system to return to normal as faults accumulate. We introduce the incremental displacement errors of GNSS and IMU-ODO preintegration to assist in determining the end of the failure event. If the incremental error determines that the system has returned to normal, we choose to use the original SD from then on.

Therefore, the switchable GNSS factor error function can be written as:

$$\parallel \boldsymbol{e}_{i}^{GNSS} \parallel^{2} = \begin{cases} \left(z_i - h^{GNSS}(X_i)\right)^{T} \Sigma_0 \left(z_i - h^{GNSS}(X_i)\right), \text{ mode judgment result} = \text{T} \\ \left(z_i - h^{GNSS}(X_i)\right)^{T} \Sigma_i \left(z_i - h^{GNSS}(X_i)\right), \text{ mode judgment result} = \text{F} \end{cases} \tag{28}$$

When the results are judged normal, it means that the edges associated with the variables are added to the graph entirely and the GNSS covariance is computed using the original SD; when the results are judged abnormal, the proportion of the edges added to the graph is decided by the weight obtained from adaptive covariance estimation. The optimization can be defined on *R* instead of [0, 1] in existing methods.

3.2. Adaptive Weighting Estimation

If the results are judged to be abnormal, the GNSS SD at this time can no longer correctly reflect the degree of data correlation. Therefore, when the result is abnormal, the factor graph optimization becomes both a state and covariance estimation problem. Then, the cost function becomes:

$$J(\boldsymbol{X},\boldsymbol{\Sigma}) = \frac{1}{2}\sum_{i=1}^{k} \boldsymbol{e}(X_i,z_i)^{\mathrm{T}}\boldsymbol{\Sigma}_i{}^{-1}\boldsymbol{e}(X_i,z_i) \tag{29}$$

where $\boldsymbol{\Sigma} = \{\Sigma_1,\Sigma_2\cdots\Sigma_k\}$ is the unknown covariance matrix.

The MAP problem can be modified to:

$$\left\{\hat{X},\hat{\boldsymbol{\Sigma}}\right\} = \arg\min_{\{X,\Sigma\}} J(\boldsymbol{X},\boldsymbol{\Sigma}) = \arg\min_{\{X,\Sigma\}}\{-\ln P(\boldsymbol{X},\boldsymbol{\Sigma}|z)\} \tag{30}$$

where $\hat{\boldsymbol{\Sigma}}$ is the estimate of the covariance matrix.

The decomposition of the posterior probability can be obtained:

$$P(\boldsymbol{X},\boldsymbol{\Sigma}|z) = P(\boldsymbol{X}|z,\boldsymbol{\Sigma})P(\boldsymbol{\Sigma}) = \prod_{i=1}^{k} P(\boldsymbol{X}_i|z_i,\boldsymbol{\Sigma}_i)P(\boldsymbol{\Sigma}_i) \tag{31}$$

According to Bayes' rule, decomposing the posterior probability density $P(\boldsymbol{X}_i|z_i,\boldsymbol{\Sigma}_i)$ is obtained:

$$P(\boldsymbol{X}_i|z_i,\boldsymbol{\Sigma}_i) = \frac{P(z_i|\boldsymbol{X}_i,\boldsymbol{\Sigma}_i)P(\boldsymbol{X}_i)}{P(z_i)} \propto P(z_i|\boldsymbol{X}_i,\boldsymbol{\Sigma}_i)P(\boldsymbol{X}_i) \tag{32}$$

where $P(\boldsymbol{X}_i)$ and $P(z_i)$ are the a priori probability densities for prediction $\boldsymbol{X}_i$ and measurement z_i, respectively. $P(z_i)$ has nothing to do with $\boldsymbol{X}_i$ and z_i, so it can be ignored. Since there is no priori covariance, the conditional probability density $p(z_i|X_i,\Sigma_i)$ is denoted [44]:

$$P(z_i|\boldsymbol{X}_i,\boldsymbol{\Sigma}_i) = \frac{1}{\sqrt{(2\pi)^k \det(\boldsymbol{\Sigma}_i)}}\exp\left\{-\frac{1}{2}\boldsymbol{e}(\boldsymbol{X}_i,z_i)^{\mathrm{T}}\boldsymbol{\Sigma}_i{}^{-1}\boldsymbol{e}(\boldsymbol{X}_i,z_i)\right\} \tag{33}$$

where $\det(\boldsymbol{\Sigma}_i)$ is the determinant of the matrix $\boldsymbol{\Sigma}_i$. However, if there is no a priori knowledge of the $\boldsymbol{\Sigma}_i$, an accurate estimate of the state $\boldsymbol{X}_i$ is not available in practice. Then, we obtain only an approximation of the state $\hat{\boldsymbol{X}}_i$ and the likelihood probability density will change to:

$$P(z_i|\boldsymbol{X}_i,\boldsymbol{\Sigma}_i) = \frac{1}{\sqrt{(2\pi)^k \det(\boldsymbol{R}_i)}}\exp\left\{-\frac{1}{2}\boldsymbol{e}(\hat{\boldsymbol{X}}_i,z_i)^{\mathrm{T}}\boldsymbol{R}_i^{-1}\boldsymbol{e}(\hat{\boldsymbol{X}}_i,z_i)\right\} \tag{34}$$

Combining Equations (30), (32), and (34), the objective function can be written as:

$$J(\boldsymbol{X},\boldsymbol{\Sigma}) \propto \frac{1}{2}\sum_{i=1}^{k}\left[\boldsymbol{e}(\hat{\boldsymbol{X}}_i,z_i)^{\mathrm{T}}\boldsymbol{R}_i^{-1}\boldsymbol{e}(\hat{\boldsymbol{X}}_i,z_i) - \ln(\det(\boldsymbol{R}_i^{-1})) - k\ln(2\pi)\right] \tag{35}$$

where $\boldsymbol{R}_i$ denotes the residual covariance matrix after the i observation. Our strategy is to find the optimal $\boldsymbol{\Sigma}_i$ to eliminate it from the expression. Assuming that the covariance matrices of the same observations are the same. The derivation of Equation (35) with respect to $\boldsymbol{R}_i^{-1}$ and letting the derivative be 0 yields R_i:

$$\boldsymbol{R}_i = \frac{1}{k}\sum_{i=1}^{k}\boldsymbol{e}(\hat{\boldsymbol{X}}_i,z_i)\boldsymbol{e}(\hat{\boldsymbol{X}}_i,z_i)^{T} \tag{36}$$

The residual sequence can be approximated as:

$$\boldsymbol{e}(\hat{\boldsymbol{X}}_i) = z_i - h(\hat{\boldsymbol{X}}_i) \approx z_i - \boldsymbol{\phi}_i\hat{\boldsymbol{X}}_i = z_i - \boldsymbol{\phi}_i(\boldsymbol{X}_i + \delta\hat{\boldsymbol{X}}_i) = \boldsymbol{n}_i - \boldsymbol{\phi}_i\delta\hat{\boldsymbol{X}}_i \tag{37}$$

where $\boldsymbol{\phi}_i$ is the Jacobian of the measurement function and $\delta\hat{\boldsymbol{X}}_i$ is the estimation error of the navigation state. $\boldsymbol{n}_i$ is the measurement noise.

The propagation process of the covariance is derived below:

$$\begin{aligned} \boldsymbol{R}_i &= \mathrm{E}(\boldsymbol{e}_i(\hat{\boldsymbol{X}}_i)\boldsymbol{e}_i(\hat{\boldsymbol{X}}_i)^T) \\ &= E[(\boldsymbol{n}_i - \boldsymbol{\phi}_i\delta\hat{\boldsymbol{X}}_i)(\boldsymbol{n}_i - \boldsymbol{\phi}_i\delta\hat{\boldsymbol{X}}_i)^T] \\ &= E(\boldsymbol{n}_i\boldsymbol{n}_i{}^T) + \boldsymbol{\phi}_i E(\delta\hat{\boldsymbol{X}}_i\delta\hat{\boldsymbol{X}}_i{}^T)\boldsymbol{\phi}_i{}^T - E(\boldsymbol{\phi}\delta\hat{\boldsymbol{X}}_i)\underbrace{E(\boldsymbol{n}_i{}^T)}_{0} - \underbrace{E(\boldsymbol{n}_i)}_{0}E(\delta\hat{\boldsymbol{X}}_i{}^T\boldsymbol{\phi}^T) \\ &= E(\boldsymbol{n}_i\boldsymbol{n}_i{}^T) + \boldsymbol{\phi}_i E(\delta\hat{\boldsymbol{X}}_i\delta\hat{\boldsymbol{X}}_i{}^T)\boldsymbol{\phi}_i{}^T \\ &= \boldsymbol{\Sigma}_i + \boldsymbol{\phi}_i Q_{\hat{\boldsymbol{X}}_i|z_i}\boldsymbol{\phi}_i{}^T \end{aligned} \tag{38}$$

where $\boldsymbol{Q}_{\hat{X}_i|z_i}$ is a posteriori covariance matrix updated with the IMU preintegrated error model:

$$\boldsymbol{Q}_{\hat{\boldsymbol{X}}_{i+1}|z_i} = \boldsymbol{Q}_{\boldsymbol{X}_{i+1,i}} - \boldsymbol{Q}_{\boldsymbol{X}_{i+1},z_i}\boldsymbol{\Sigma}_{\boldsymbol{i}}{}^{-1}\boldsymbol{Q}_{z_i,\boldsymbol{X}_{i+1}} \tag{39}$$

$$\begin{cases} \boldsymbol{Q}_{\boldsymbol{X}_{i+1,i}} = \boldsymbol{M}_i\boldsymbol{Q}_{\boldsymbol{X}_i}\boldsymbol{M}_i^{\mathrm{T}} + \boldsymbol{N}_i\boldsymbol{\Sigma}^a\boldsymbol{N}_i^{\mathrm{T}} + \boldsymbol{F}_i\boldsymbol{\Sigma}^g\boldsymbol{F}_i^{\mathrm{T}} \\ \boldsymbol{Q}_{\boldsymbol{X}_{i+1},z_i} = \boldsymbol{Q}_{\boldsymbol{X}_{i+1,i}}\boldsymbol{H}_i^{\mathrm{T}} \\ \boldsymbol{Q}_{z_i,\boldsymbol{X}_{i+1}} = \boldsymbol{H}_k\boldsymbol{Q}_{\boldsymbol{X}_{i+1,i}} \end{cases} \tag{40}$$

where $\boldsymbol{M}_i$ is the derivation of Equation (9) with respect to position, velocity, and attitude. $\boldsymbol{N}_i$ and $\boldsymbol{F}_i$ are the derivations of the equations of motion with respect to force and angular velocity. $\boldsymbol{\Sigma}^a$ and $\boldsymbol{\Sigma}^g$ are the measurement noise of the accelerometer and gyroscope, respectively.

Thus, the estimation of the measurement covariance can be obtained:

$$\boldsymbol{\Sigma}_i = \boldsymbol{R}_i - \boldsymbol{\phi}_i\boldsymbol{Q}_{\hat{X}_i|z_i}\boldsymbol{\phi}_i^T \tag{41}$$

From the derivation, it can be seen that the estimation $\boldsymbol{\Sigma}_i$ requires a priori $\boldsymbol{R}_i$, where the a priori value is averaged over the residual covariance of all observations before switching to adaptive weight estimation.

4. Experiment Results and Discussion

To evaluate the performance of the proposed method, we built the testbed for the IMU/GNSS/ODO integrated navigation system. The data acquisition vehicle is shown in Figure 4. The sensors used are all commercial and the specific parameters are shown in Table 2. The sampling frequency of the MEMS-IMU SCHA634-D03 is set at 100 Hz and the frequency of the Bynav X1 GNSS receiver is at 5 Hz. We use RTK mode to obtain the positioning results in the experiments. The ODO is mounted on the rear wheel bearing. In addition, forward and backward filtering postprocessing is used to provide the ground truth. We run the algorithms on a desktop computer equipped with an Intel i7-12700 at 2.10 GHz and 32 GB RAM and the size of the sliding window is set to 1 s.

The vehicle-mounted test was conducted on a viaduct in the Daxing district, Beijing, with an average speed of 16 km/h and a duration of 1190 s. The specific trajectory is shown in Figure 5a. The experimental environment includes up and down the viaduct and stable driving. However, in complex urban environments, GNSS signals are highly susceptible to interference; thus, the SD output from the GNSS board is not completely reliable. We choose #1 and #2, which have no faults in raw data then add faults manually to simulate the situation where there are faults in the GNSS signals but the SD judgment is completely ineffective. To verify the effectiveness and robustness of the proposed algorithm, three sections of the trajectory including two typical fault types are selected for verification:

(1) #1 and #2 are selected during the stable driving process in an open environment, which means that the original data in these segments are error-free. The complete failure of SD detection, which is the worst-case scenario, can be simulated by adding errors. To simulate the ramp and step fault of the satellite, we add slow-growth faults at 400 s to 580 s (#1), lasting 180 s, and step faults at 700 s to 730 s (#2), lasting 30 s, to the latitude and longitude of the satellite. The GNSS anomaly detection algorithm is validated using all segments including simulation errors.

(2) #1 and #3 are selected to verify the effectiveness of SAWE-FGO. #3 (from 1153 s to 1162 s) is a winding road where the vehicle passes through the staggered elevation before the turn, with continuous random jump points in the raw data (from 1138 s to 1152 s).

Table 2. Parameters of relevant navigation sensors.

Sensors	Parameter	Value
IMU	Gyro Bias instability	0.9°/h
	Accelerometer Bias instability	0.01 mg
	Accelerometer RMS noise	0.5 mg
	Gyro RMS noise	0.007°/s
GNSS RTK	Positional accuracy	1.5 cm + 1 ppm
Odometer	Positional accuracy	1% × Distance

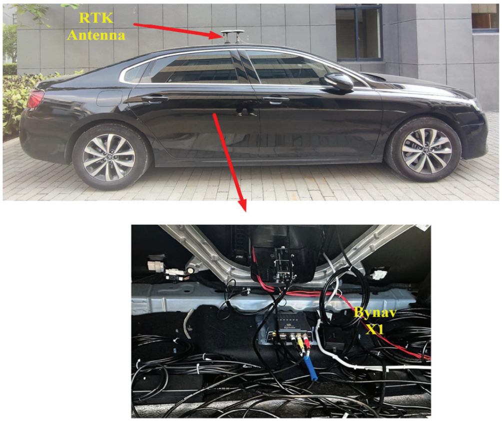
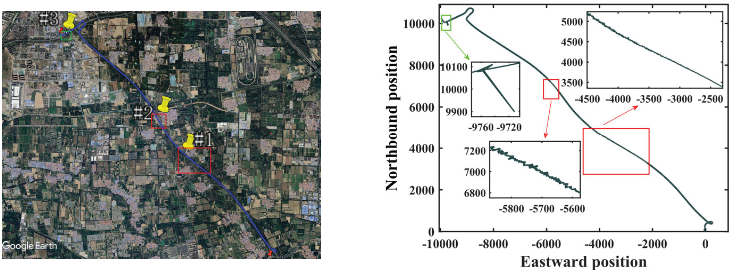


Figure 4. Data collection scheme.



(**a**) (**b**)

Figure 5. The left figure (**a**) shows the trajectory in field tests. The right figure (**b**) shows the GNSS output trajectory.

The GNSS output trajectory after adding the faults is shown in Figure 5b, with the distance information relative to the start point (0 s).

4.1. Validation of the GNSS Anomaly Detection Algorithm

In the simulation, the whole duration is about 180 s in #1 and 30 s in #2; the GNSS measurements are available during these two periods of time. Slow-growth faults and step faults are added to the latitude and longitude, respectively. The size of the added faults in #1 (group A) and #2 (group B) is shown in Figure 6.

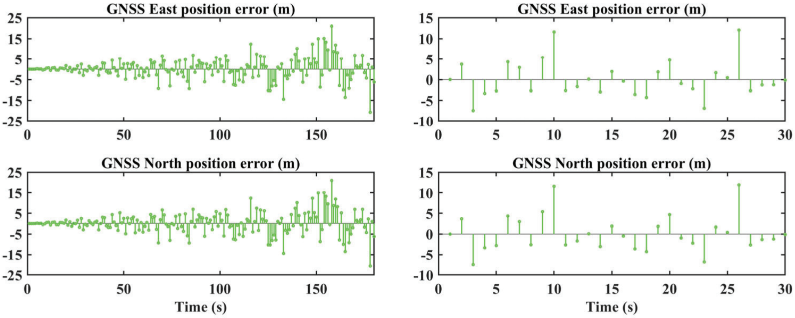


Figure 6. GNSS error in group A (**left**) and group B (**right**).

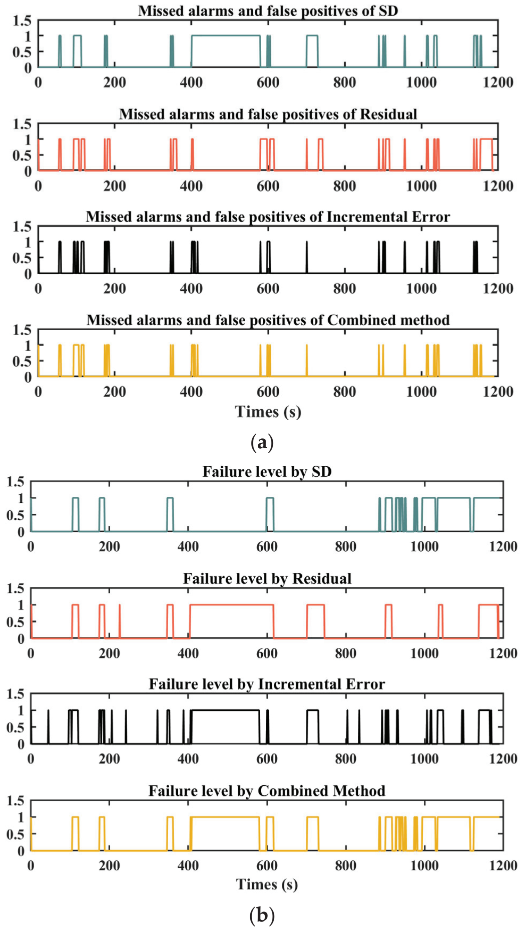
We apply the proposed GNSS anomaly detection method to the entire trajectory segment, which includes both real and simulated faults, for comprehensive evaluation. To illustrate the specific results of typical failure types, we select two representative segments of the trajectory (#1 and #2).

The missed alarms and false positives of different methods are demonstrated in Figure 7a. For the judgment curves, a value of 1 means missed alarms or false positives and a value of 0 means normal detection. Meanwhile, the failure level by different methods is shown in Figure 7b. For the judgment curves, a value of 1 signifies that the detection method detects a perceived fault and a value of 0 denotes no fault.

As Figure 7a illustrates, the SD statistics have a long duration during 400 s to 580 s, while the residual statistics identify the missed detection. However, a 17 s duration is required for the residual statistics parameter to fall under the fault alarm shoulder. Meanwhile, a new fault appears in the raw data at 597 s. As shown in Figure 7b, the residual detection is unable to distinguish between the two adjacent faults and it is straightforward to determine that all of them are abnormal from 404 s to 618 s. The proposed method is composed of an SD test unit, a residual test unit, and an incremental error unit, which significantly reduces missed alarms. As Table 3 shows, the missed alarm rate of the combined method is reduced by 81.7% compared with the SD criterion and the false alarm rate of the combined method is reduced by 85.7% compared with the residual criterion.

Table 3. Statistical results of missed alarms and false positives.

Error Level	SD Criterion	Residuals Criterion	Incremental Error Criterion	Combined Method
Number of false detections	276	151	69	74
Number of missed alarms	272	38	48	50
Number of false positives	4	113	21	24
Missed alarm rate	0.93	0.13	0.16	0.17
False alarm rate	0.009	0.14	0.02	0.02



(**a**)

(**b**)

Figure 7. (**a**) Missed alarms and false positives of different methods for the whole trajectory. (**b**) Fault detection of different methods for the whole trajectory.

In addition, to represent the performance of the use of the incremental error criterion to assist the residual criterion in the case of complete failure of SD detection, #1 and #2 are selected to evaluate the advantages of the proposed method by comparing the detection and recovery ability for different types of faults, as shown in Figures 8 and 9. Figure 8 shows the results for group A in ramp failure, while Figure 9 presents the results for group B in step failure. The statistical curves of residual, incremental error, and combined fault detection methods are compared in Figures 8a and 9a where the pink line represents the threshold. Meanwhile, the judgment results of these methods are presented in Figures 8b and 9b. For the judgment curves, a value of 1 means that the detection method detects a perceived fault and a value of 0 means normal. The fault detection delays for different detection methods are listed in Table 4.

Among the results presented in Table 4, the following conclusions can be easily found:

(1) For step faults, all methods are able to detect the start of the fault quickly. For ramp failure, the combined fault detection method chooses the minimum detection time due to the sensitivity of the residual method.

(2) The proposed method can recover after the failure termination, with a 92–97% reduction in delay compared to the residual method.

(3) In group A, it is not hard to see original faults appearing after the end of adding faults. In Figure 8, the residual method cannot distinguish between different faults, which is not possible for multiple-failure detection. Instead, the proposed method overcomes the drawback of the residual method in failure recovery, which can detect second faults.

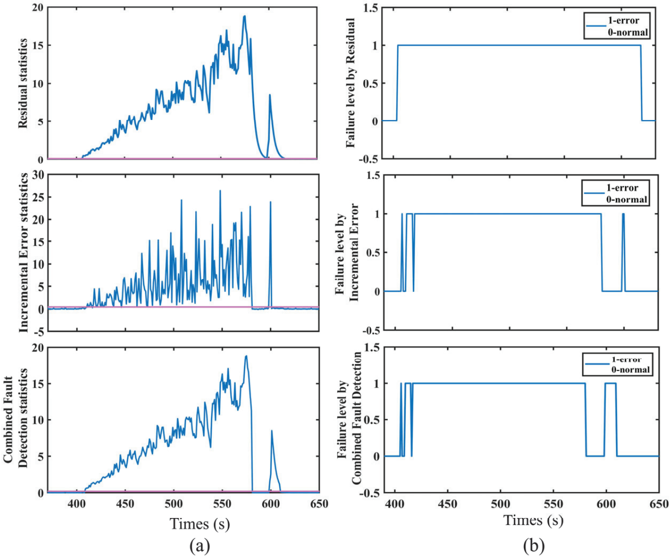


Figure 8. Fault detection results for ramp failure (group A). (**a**) The statistical curves of different methods. (**b**) The judgment results of different methods.

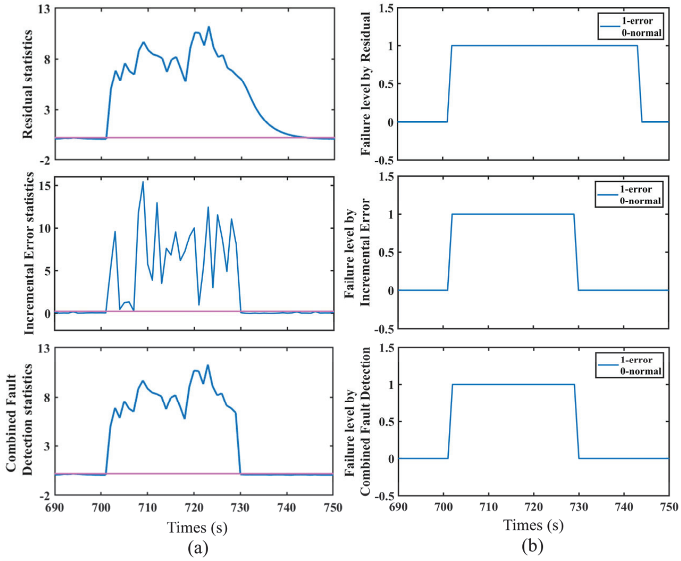


Figure 9. Fault detection results for step failure (group B). (**a**) The statistical curves of different methods. (**b**) The judgment results of different methods.

Table 4. Delayed time of fault detection and recovery.

Failure Type	Group	Residuals Criterion	Incremental Error Criterion	Combined Method
Slope Failure	A detection	4 s	9 s	4 s
	A recovery	38 s	1 s	1 s
Step Failure	B detection	1 s	2 s	1 s
	B recovery	13 s	0 s	0 s

Fault delay detection and recovery usually adversely affect the parameter estimation, making the AWE not optimal. Therefore, the proposed detection algorithm is used to solve this problem in the next part, which can quickly and accurately determine the moment when the system returns to normal and then switch to the normal weighting mode.

4.2. Comparison of Performance between Different Information Fusion Methods

To verify the effectiveness of the switching adaptive weighting estimation (SAWE), #1 is selected for verification. We compared our methods with the EKF method, which is commonly used in engineering applications, and the traditional FGO method. The elapsed time of all data for the FGO algorithm with the equipped CPU is about 18 s. Figures 10–12 demonstrate the localization results produced by each algorithm.

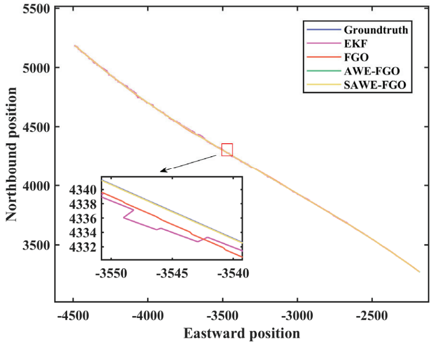


Figure 10. Positioning trace of different integrated navigation algorithms in #1.

The traces of the different algorithms are shown in Figure 10. It can be seen that, after the addition of the damp faults, there are obvious fluctuations in EKF. The smoothing effect of FGO is better than that of the EKF; however, there are still some large errors here. Instead, there is not a significant decrease in the navigation accuracy of the AWE-FGO and SAWE-FGO. Figures 11 and 12 show the position errors and velocity errors, respectively. We can see that the position and velocity errors are remarkably reduced using AWE-FGO and SAWE-FGO. The position results are shown in Table 5.

For the eastward position, the STD values of FGO and AWE-FGO reach 1.304 m and 0.509 m, respectively, while EKF reaches 2.155 m. The STD of FGO improved by 39.4% compared to EKF. Similarly, for the northward position, the STD improves by 19.9%. It can be seen that the positioning accuracy of EKF and FGO is comparable. After applying AWE to the FGO, the errors reduce noticeably in Table 5, which shows the effectiveness of the proposed AWE method. Moreover, with the help of SAWE via SAWE-FGO, the position errors decrease to 0.489 m and 0.208 m, respectively, resulting in a further improvement in accuracy from AWE-FGO. RMSE can reflect the deviation of the estimated value from the reference value. The use of RMSE can assess the impact of outliers on navigation and reflect the navigation accuracy and robustness. It can be seen that the RMSE of AWE-FGO is significantly smaller than that of EKF and FGO, with reductions of 76.3% and 57.2% in eastward and northward positions, respectively. When the traditional factor graph algorithm is enhanced by SAWE-FGO, the position accuracy improves by 80%. The higher accuracy and robustness of the proposed method depend on the fact that the SAWE-FGO can recognize the end time of faults and adjust the corresponding covariance according to the error changes, which makes the weighting more accurate. In contrast, the traditional factor graph model does not take into account the exceptions of sensor measurements in the information fusion process, which leads to a decrease in navigation accuracy.

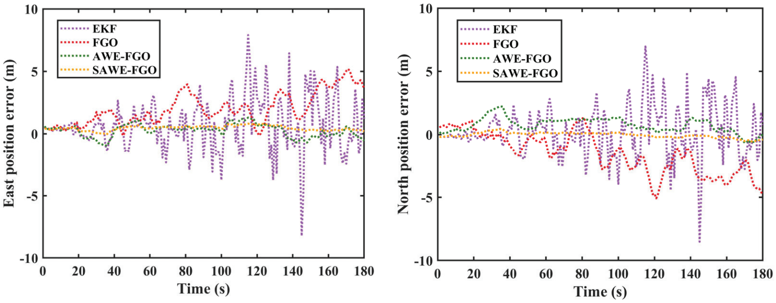


Figure 11. Position errors of different integrated navigation algorithms in #1.

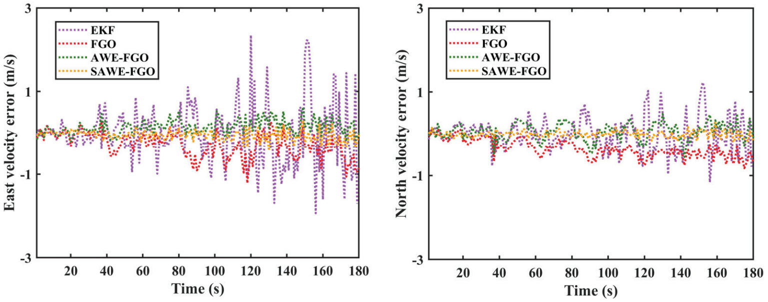


Figure 12. Velocity errors of different integrated navigation algorithms in #1.

In addition, we interestingly discovered that the fault detection delay has a much greater effect on the turnaround than the straight line. Therefore, we choose #3 for additional validation. It can be seen in Figure 5b that there are random jump points in the GNSS signals before passing a turnaround section (#3) as the road crosses the viaduct for a while. As shown in Figure 13, we use the SAWE-FGO method compared with full use of the AWE-FGO method for the selected trajectory (#3). It can be seen that the AWE-FGO method causes an error of about 1 m in the northward position, while the SAWE-FGO method has a maximum of only 0.1 m, which validates the effectiveness of SAWE.

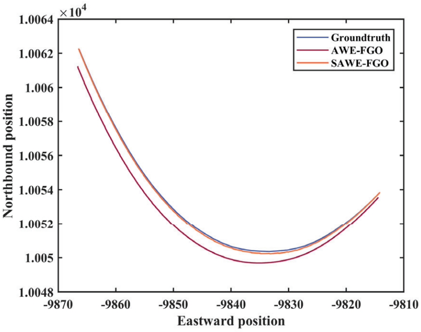


Figure 13. The comparison of trajectory in #3.

Table 5. STD and RMSE of position and velocity errors for different algorithms in #1.

Error	Method	STD	RMSE
East position error (m)	EKF	2.155	2.248
	FGO	1.304	2.396
	AWE-FGO	0.509	0.570
	SAWE-FGO	0.189	0.489
North position error (m)	EKF	2.051	2.062
	FGO	1.642	2.227
	AWE-FGO	0.586	0.954
	SAWE-FGO	0.206	0.208
East velocity error (m/s)	EKF	0.721	0.720
	FGO	0.297	0.410
	AFGO	0.174	0.206
	SAWE-FGO	0.122	0.132
North velocity error (m/s)	EKF	0.393	0.392
	FGO	0.210	0.409
	AFGO	0.168	0.206
	SAWE-FGO	0.131	0.078

4.3. Experimental Validation Using an Open-Source Dataset

To challenge the performance of the proposed fault detection algorithm and adaptive weight estimation strategy, we test the proposed algorithms on the open-source dataset collected in the Odaiba districts of Tokyo Urban Canyons [45], which has denser buildings and multiple viaducts compared with the experimental environment in which we collected data. In other words, the selected open-source dataset is collected in a typical urban canyon environment with a higher percentage of multipath/NLOS effects.

Figure 14 shows the vehicle traveling trajectory provided by the high-precision INS/RTK positioning results. It can be seen that the road condition is quite complicated, which not only contains straight lines and turning sections but also passes through dense buildings and several viaducts.

In order to validate the effectiveness of the proposed algorithm, we performed fault detection validation and adaptive weight estimation validation on this dataset, respectively. The missed alarms and false positives of different methods are demonstrated in Figure 15. For the judgment curves, a value of 1 means missed alarms and false positives and a value of 0 means normal detection.

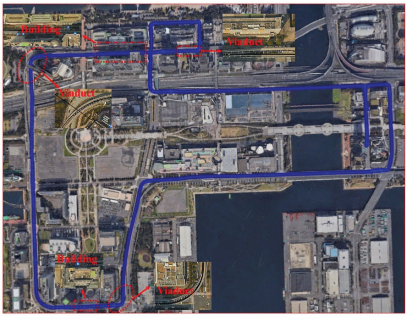


Figure 14. The vehicle traveling trajectory.

Figure 15 shows that, in typical urban canyon environments, GNSS signals are significantly disturbed by multipath/NLOS effects, resulting in frequent anomalies in the SD values of GNSS outputs.

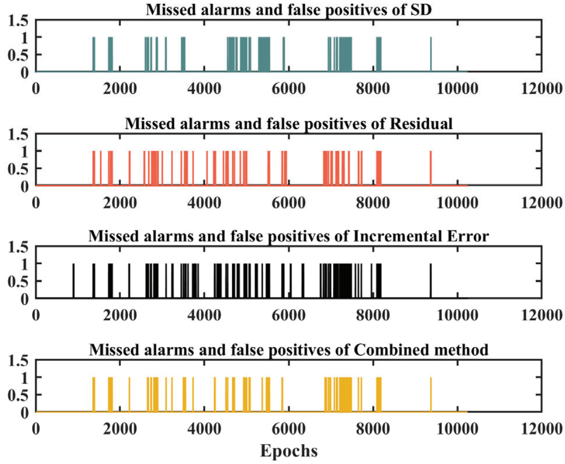


Figure 15. Missed alarms and false positives of different methods for the whole trajectory.

As Table 6 shows, the missed alarm rate of the combined method is reduced by 41% compared with the SD criterion and the false alarm rate of the combined method is reduced by 62% compared to the residual criterion.

Table 6. Statistical results of missed alarms and false positives.

Error Level	SD Criterion	Residuals Criterion	Incremental Error Criterion	Combined Method
Number of false detections	1220	1100	1470	980
Number of missed alarms	1210	430	900	720
Number of false positives	10	670	570	260
Missed alarm rate	0.63	0.22	0.46	0.37
False alarm rate	0.001	0.08	0.07	0.03

A comparison of performance between different information fusion methods is shown in Figures 16 and 17.

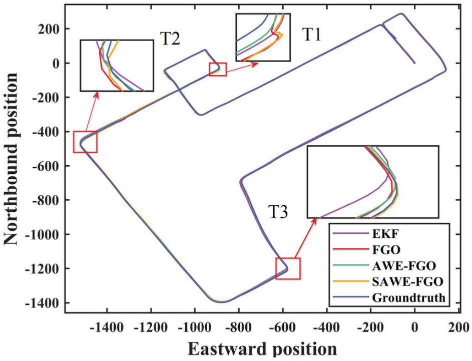


Figure 16. Positioning trace of different integrated navigation algorithms.

The traces of the different algorithms are shown in Figure 16. The three zoomed-in trajectories (T1, T2, and T3) correspond to three elevated segments. With the AWE method, the trajectories at the corners are significantly smoother and the corresponding errors are smaller than those of EKF and FGO. When the system is operating in switchable modes, there is a further reduction in error at corners, except for section T1, which occurs because the fault detection algorithm fails to detect partial faults in the segment, failing to switch up to AWE mode.

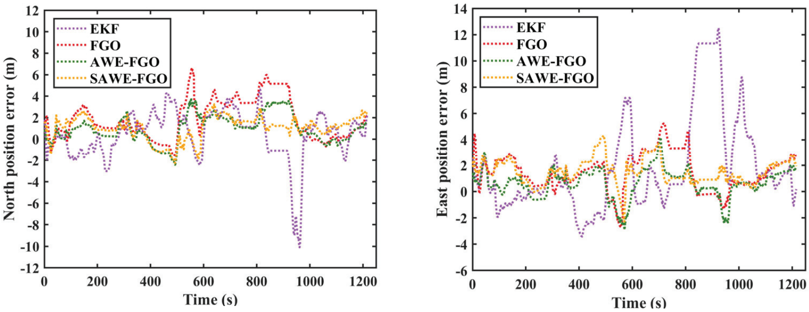


Figure 17. Position errors of different integrated navigation algorithms.

Figure 17 shows the position error. It can be seen that the EKF has a large error and FGO has a relatively reduced error, while the AWE-FGO significantly reduces the error. The position results are shown in Table 7.

Table 7. STD and RMSE of position errors for different algorithms.

Error	Method	STD	RMSE
East position error (m)	EKF	3.656	4.115
	FGO	1.438	2.057
	AWE-FGO	1.108	1.282
	SAWE-FGO	0.906	1.805
North position error (m)	EKF	2.320	2.354
	FGO	1.858	2.631
	AWE-FGO	1.273	1.605
	SAWE-FGO	1.088	1.509

For the eastward position, the STD values of AWE-FGO and SAWE-FGO reach 1.108 m and 0.906 m, respectively. The position accuracy of AWE-FGO improved by 37.7% compared with FGO. Relative to EKF, the AWE-FGO errors are reduced by 68.8%. Similarly, for the northward position, the position accuracy of AWE-FGO improves by 39.0% compared with FGO. Compared to EKF, the AWE-FGO errors are reduced by 45.1%. While SAWE-FGO significantly reduces errors in the northward position, it does not perform as well in the eastward position due to the failure of fault detection at T1.

In short, the errors are reduced significantly with the help of AWE. Moreover, the proposed fault detection method based on multi-conditional analysis is also validated on the challenging dataset. With the aid of fault detection, the SAWE exhibits less positional error. Since the selection of the threshold values directly affects the effectiveness of fault detection, we need to continue to improve and explore the criteria for threshold selection in the future.

5. Conclusions

To improve the accuracy and robustness of the integrated navigation system in complex urban environments, this paper proposes a robust factor graph optimization method with switchable adaptive weight estimation. Aiming at the shortcomings of the existing factor graph algorithms, an adaptive covariance estimation factor graph navigation algorithm is proposed, which effectively suppresses the impact of GNSS faults on the accuracy of the navigation system. In addition, we also propose a GNSS anomaly detection strategy based on multi-conditional analysis, which improves the system fault detection capability and effectively improves the robustness of the system. The superiority and robustness of the proposed method relative to other methods have been verified in vehicle experiments. The positioning accuracy of the proposed AWE-FGO method is improved by more than 60% compared to the traditional FGO algorithm. Moreover, the proposed fault detection method is effective in detecting the end of the fault and greatly reduces the effect of delay on parameter estimation. Therefore, SAWE-FGO can be used to produce superior navigational results. Furthermore, we verified the performance of the algorithm on a real urban canyon dataset and the results show that the proposed method improves the navigation accuracy by more than 30% compared to the FGO algorithm. In more complex environments, the threshold selection of faults is more difficult and the improvement of threshold selection criteria will be one of the focuses of future work.

Author Contributions: Conceptualization, Y.Z.; Methodology, S.Q.; Software, S.C.; Formal analysis, S.Z.; Investigation, H.H.; Resources, Q.Z.; Writing—original draft preparation, S.Q.; Writing—review and editing, Y.Z.; supervision, H.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key R&D Program of China, grant number 2022YFB3903800.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to acknowledge Xiaoji Niu and the Integrated and Intelligent Navigation (i2Nav) group from Wuhan University for providing the OB_GINS software that was used in the paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Groves, P. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*; Artech: Morristown, NJ, USA, 2007.
2. Inside GNSS. *Multipath vs. NLOS Signals*; Inside GNSS—Global Navigation Satellite Systems Engineering, Policy, and Design: Red Bank, NJ, USA, 2013.
3. Boguspayev, N.; Akhmedov, D.; Raskaliyev, A.; Kim, A.; Sukhenko, A. A Comprehensive Review of GNSS/INS Integration Techniques for Land and Air Vehicle Applications. *Appl. Sci.* **2023**, *13*, 4819. [CrossRef]
4. Mu, M.; Zhao, L. A GNSS/INS-Integrated System for an Arbitrarily Mounted Land Vehicle Navigation Device. *GPS Solut.* **2019**, *23*, 112. [CrossRef]
5. Lou, T.-S.; Chen, N.-H.; Chen, Z.-W.; Wang, X.-L. Robust Partially Strong Tracking Extended Consider Kalman Filtering for INS/GNSS Integrated Navigation. *IEEE Access* **2019**, *7*, 151230–151238. [CrossRef]
6. Allotta, B.; Caiti, A.; Costanzi, R.; Fanelli, F.; Fenucci, D.; Meli, E.; Ridolfi, A. A New AUV Navigation System Exploiting Unscented Kalman Filter. *Ocean Eng.* **2016**, *113*, 121–132. [CrossRef]
7. Arasaratnam, I.; Haykin, S. Cubature Kalman Filters. *IEEE Trans. Autom. Control* **2009**, *54*, 1254–1269. [CrossRef]
8. Bell, B.M.; Cathey, F.W. The Iterated Kalman Filter Update as a Gauss-Newton Method. *IEEE Trans. Autom. Control* **1993**, *38*, 294–297. [CrossRef]
9. Indelman, V.; Williams, S.; Kaess, M.; Dellaert, F. Information Fusion in Navigation Systems via Factor Graph Based Incremental Smoothing. *Robot. Auton. Syst.* **2013**, *61*, 721–738. [CrossRef]
10. Wen, W.; Bai, X.; Kan, Y.C.; Hsu, L.-T. Tightly Coupled GNSS/INS Integration via Factor Graph and Aided by Fish-Eye Camera. *IEEE Trans. Veh. Technol.* **2019**, *68*, 10651–10662. [CrossRef]
11. Wen, W.; Pfeifer, T.; Bai, X.; Hsu, L.-T. It Is Time for Factor Graph Optimization for GNSS/INS Integration: Comparison between FGO and EKF. *arXiv* **2020**, arXiv:2004.10572.
12. Zeng, Q.; Chen, W.; Liu, J.; Wang, H. An Improved Multi-Sensor Fusion Navigation Algorithm Based on the Factor Graph. *Sensors* **2017**, *17*, 641. [CrossRef] [PubMed]
13. Dai, J.; Liu, S.; Hao, X.; Ren, Z.; Yang, X. UAV Localization Algorithm Based on Factor Graph Optimization in Complex Scenes. *Sensors* **2022**, *22*, 5862. [CrossRef] [PubMed]
14. Xu, J.; Yang, G.; Sun, Y.; Picek, S. A Multi-Sensor Information Fusion Method Based on Factor Graph for Integrated Navigation System. *IEEE Access* **2021**, *9*, 12044–12054. [CrossRef]
15. Dehghannasiri, R.; Esfahani, M.S.; Dougherty, E.R. Intrinsically Bayesian Robust Kalman Filter: An Innovation Process Approach. *IEEE Trans. Signal Process.* **2017**, *65*, 2531–2546. [CrossRef]
16. Zhao, D.; Hancock, C.M.; Roberts, G.W.; Jin, S. Cycle Slip Detection during High Ionospheric Activities Based on Combined Triple-Frequency GNSS Signals. *Remote Sens.* **2019**, *11*, 250. [CrossRef]
17. Yuan, H.; Zhang, Z.; He, X.; Xu, T.; Xu, X.; Zang, N. Real-Time Cycle Slip Detection and Repair Method for BDS-3 Five-Frequency Data. *IEEE Access* **2021**, *9*, 51189–51201. [CrossRef]

18. Sun, K.; Zeng, Q.; Liu, J.; Wang, S. Fault Detection of Resilient Navigation System Based on GNSS Pseudo-Range Measurement. *Appl. Sci.* **2022**, *12*, 5313. [CrossRef]
19. Roysdon, P.F.; Farrell, J.A. GPS-INS Outlier Detection & Elimination Using a Sliding Window Filter. In Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017; pp. 1244–1249.
20. Wen, W.; Meng, Q.; Hsu, L.-T. Integrity Monitoring for GNSS Positioning via Factor Graph Optimization in Urban Canyons. In Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021), St. Louis, MO, USA, 20–24 September 2021; pp. 1508–1515.
21. Hu, Y.; Li, H.; Liu, W. Robust Factor Graph Optimisation Method for Shipborne GNSS/INS Integrated Navigation System. *IET Radar Sonar Navig.* **2023**, *17*, 1–17. [CrossRef]
22. Zhang, C.; Zhao, X.; Pang, C.; Li, T.; Zhang, L. Adaptive Fault Isolation and System Reconfiguration Method for GNSS/INS Integration. *IEEE Access* **2020**, *8*, 17121–17133. [CrossRef]
23. Su, S.; Dai, H.; Cheng, S.; Lin, P.; Hu, C.; Lv, B. A Robust Magnetic Tracking Approach Based on Graph Optimization. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 7933–7940. [CrossRef]
24. Sünderhauf, N.; Protzel, P. Switchable Constraints for Robust Pose Graph SLAM. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 1879–1884.
25. Pfeifer, T.; Lange, S.; Protzel, P. Dynamic Covariance Estimation—A Parameter Free Approach to Robust Sensor Fusion. In Proceedings of the 2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Daegu, Republic of Korea, 16–18 November 2017; pp. 359–365.
26. Wei, X.; Li, J.; Zhang, D.; Feng, K. An Improved Integrated Navigation Method with Enhanced Robustness Based on Factor Graph. *Mech. Syst. Signal Process.* **2021**, *155*, 107565. [CrossRef]
27. Lesouple, J.; Robert, T.; Sahmoudi, M.; Tourneret, J.-Y.; Vigneau, W. Multipath Mitigation for GNSS Positioning in an Urban Environment Using Sparse Estimation. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 1316–1328. [CrossRef]
28. Yi, B.; Lee, M.A.; Kloss, A.; Martín-Martín, R.; Bohg, J. Differentiable Factor Graph Optimization for Learning Smoothers. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021.
29. Nam, D.V.; Gon-Woo, K. Learning Type-2 Fuzzy Logic for Factor Graph Based-Robust Pose Estimation with Multi-Sensor Fusion. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 3809–3821. [CrossRef]
30. Chang, L.; Niu, X.; Liu, T. GNSS/IMU/ODO/LiDAR-SLAM Integrated Navigation System Using IMU/ODO Pre-Integration. *Sensors* **2020**, *20*, 4702. [CrossRef]
31. Bai, S.; Lai, J.; Lyu, P.; Cen, Y.; Ji, B. Improved Preintegration Method for GNSS/IMU/In-Vehicle Sensors Navigation Using Graph Optimization. *IEEE Trans. Veh. Technol.* **2021**, *70*, 11446–11457. [CrossRef]
32. Kschischang, F.R.; Frey, B.J.; Loeliger, H.-A. Factor Graphs and the Sum-Product Algorithm. *IEEE Trans. Inf. Theory* **2001**, *47*, 498–519. [CrossRef]
33. Barfoot, T.D. *State Estimation for Robotics*; Cambridge University Press: Cambridge, UK, 2017; ISBN 1-108-50971-1.
34. Dellaert, F.; Kaess, M. Factor Graphs for Robot Perception. *FNT Robot.* **2017**, *6*, 1–139. [CrossRef]
35. Carlone, L.; Kira, Z.; Beall, C.; Indelman, V.; Dellaert, F. Eliminating Conditionally Independent Sets in Factor Graphs: A Unifying Perspective Based on Smart Factors. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation, ICRA 2014, Hong Kong, China, 31 May–7 June 2014; pp. 4290–4297.
36. Kaess, M.; Ranganathan, A.; Dellaert, F. iSAM: Incremental Smoothing and Mapping. *IEEE Trans. Robot.* **2008**, *24*, 1365–1378. [CrossRef]
37. Loeliger, H.-A.; Dauwels, J.; Hu, J.; Korl, S.; Ping, L.; Kschischang, F.R. The Factor Graph Approach to Model-Based Signal Processing. *Proceedings of the IEEE* **2007**, *95*, 1295–1322. [CrossRef]
38. Moré, J.J. *Levenberg–Marquardt Algorithm: Implementation and Theory*; Springer: Berlin/Heidelberg, Germany, 1977.
39. Kaess, M.; Johannsson, H.; Roberts, R.; Ila, V.; Leonard, J.; Dellaert, F. iSAM2: Incremental Smoothing and Mapping with Fluid Relinearization and Incremental Variable Reordering. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 3281–3288.
40. Tardif, J.-P.; George, M.; Laverne, M.; Kelly, A.; Stentz, A. A New Approach to Vision-Aided Inertial Navigation. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; pp. 4161–4168.
41. Chang, L.; Niu, X.; Liu, T.; Tang, J.; Qian, C. GNSS/INS/LiDAR-SLAM Integrated Navigation System Based on Graph Optimization. *Remote Sens.* **2019**, *11*, 1009. [CrossRef]
42. Wu, Y.; Pan, X. Velocity/Position Integration Formula Part II: Application to Strapdown Inertial Navigation Computation. *IEEE Trans. Aerosp. Electron. Syst.* **2013**, *49*, 1024–1034. [CrossRef]
43. Chiu, H.-P.; Williams, S.; Dellaert, F.; Samarasekera, S.; Kumar, R. Robust Vision-Aided Navigation Using Sliding-Window Factor Graphs. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 46–53.

44. Carlone, L. State Estimation for Robotics [Bookshelf]. *IEEE Control Syst. Mag.* **2019**, *39*, 86–88. [CrossRef]
45. Hsu, L.-T.; Kubo, N.; Wen, W.; Chen, W.; Liu, Z.; Suzuki, T.; Meguro, J. UrbanNav:An Open-Sourced Multisensory Dataset for Benchmarking Positioning Algorithms Designed for Urban Areas. In Proceedings of the 34th International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GNSS+ 2021, St. Louis, MO, USA, 20–24 September 2021; pp. 226–256.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

DLD-SLAM: RGB-D Visual Simultaneous Localisation and Mapping in Indoor Dynamic Environments Based on Deep Learning

Han Yu [1], Qing Wang [1,*], Chao Yan [1,2], Youyang Feng [1], Yang Sun [1] and Lu Li [1]

[1] School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China; 220213656@seu.edu.cn (H.Y.); chaoyan@seu.edu.cn (C.Y.); 230159565@seu.edu.cn (Y.F.); 220213632@seu.edu.cn (Y.S.); 220203621@seu.edu.cn (L.L.)

[2] School of Electrical Engineering and Automation, Changshu Institute of Technology, Changshu 215500, China

* Correspondence: wq_seu@seu.edu.cn

Abstract: This work presents a novel RGB-D dynamic Simultaneous Localisation and Mapping (SLAM) method that improves the precision, stability, and efficiency of localisation while relying on lightweight deep learning in a dynamic environment compared to the traditional static feature-based visual SLAM algorithm. Based on ORB-SLAM3, the GCNv2-tiny network instead of the ORB method, improves the reliability of feature extraction and matching and the accuracy of position estimation; then, the semantic segmentation thread employs the lightweight YOLOv5s object detection algorithm based on the GSConv network combined with a depth image to determine potentially dynamic regions of the image. Finally, to guarantee that the static feature points are used for position estimation, dynamic probability is employed to determine the true dynamic feature points based on the optical flow, semantic labels, and the state in last frame. We have performed experiments on the TUM datasets to verify the feasibility of the algorithm. Compared with the classical dynamic visual SLAM algorithm, the experimental results demonstrate that the absolute trajectory error is greatly reduced in dynamic environments, and that the computing efficiency is improved by 31.54% compared with the real-time dynamic visual SLAM algorithm with close accuracy, demonstrating the superiority of DLD-SLAM in accuracy, stability, and efficiency.

Keywords: visual SLAM; dynamic environments; GCNv2-tiny feature points; lightweight object detection; LK optical flow

Citation: Yu, H.; Wang, Q.; Yan, C.; Feng, Y.; Sun, Y.; Li, L. DLD-SLAM: RGB-D Visual Simultaneous Localisation and Mapping in Indoor Dynamic Environments Based on Deep Learning. *Remote Sens.* **2024**, *16*, 246. https://doi.org/10.3390/rs16020246

Academic Editors: Giuseppe Casula, Vagner Ferreira, Zhetao Zhang, Guorui Xiao and Zhixi Nie

Received: 24 October 2023
Revised: 7 December 2023
Accepted: 15 December 2023
Published: 8 January 2024

1. Introduction

Visual Simultaneous Localisation and Mapping (VSLAM) is capable of accurately sensing the environment and obtaining the position of the robot. The extensive implementation of VSLAM in autonomous vehicles, perception, and robot technology can be attributed to its cost-effectiveness, improved accuracy, and lack of reliance on specialised sensors [1]. In recent years, there have been notable advancements in VSLAM algorithms. Examples of these advancements include ORB-SLAM2 [2], ORB-SLAM3 [3], VINS-Mono [4], SVO [5], and others. The above open source SLAM algorithms have primarily been designed for static environments. However, in dynamic environments, especially when the texture of moving objects is obvious or occupies a large portion of the image, the accuracy and robustness of the system decrease dramatically. The aforementioned issues have garnered interest towards the integration of VSLAM with deep learning technology. As environments with dynamic objects are frequently present in people's practical application, it is of great practical significance to further develop VSLAM algorithms with stronger robustness, adaptability, and practicality in dynamic environments.

In recent years, the emergence of deep learning technology has brought new opportunities for the improvement of VSLAM. Compared with the traditional SLAM algorithms based on geometry and feature points, the incorporation of deep learning and VSLAM can augment their capacity to address some difficulties and challenges in SLAM problems [6]. There are various factors that may seriously impact the accuracy and reliability of feature detection. The feature extraction and matching segment at the front end of most SLAM algorithms is unable to extract reliable and consistent features in complex dynamic environments, which can lead to problems such as lost feature tracking and positioning failure. Deep learning methods, such as DeepFeat [7] and LIFT [8], have the ability to acquire highly discriminative feature points from large amounts of data. These networks can extract feature points that contain more geometric and semantic information, thus enhancing the accuracy and robustness of the SLAM system.

Deep learning techniques have demonstrated substantial advancements in the fields of target detection and semantic segmentation, among others. Deep learning techniques have been able to achieve high-precision target detection that can identify and locate multiple targets in complex scenes [9]. Therefore, deep learning techniques can be used to preliminarily segment the position and semantic information of objects in the dynamic environment and identify potential dynamic objects, with different techniques detecting objects with different accuracies. Target detection can generally obtain the detection frame of an object, while semantic segmentation and instance segmentation can be used to mask the region of an object, and instance segmentation can distinguish different individuals of the same category. As the accuracy increases, the deep learning network becomes more complex, which can impact the real-time performance of the system when applied to the SLAM system. Therefore, the operational efficiency of the algorithm can be maintained at the expense of target detection accuracy, and at the same time, the geometric information can be used to accurately identify the region of dynamic objects and improve detection accuracy. It is also feasible to improve the efficiency of the algorithm and reduce the unnecessary computational burden by lightweighting the neural network. After detecting possible dynamic objects using deep learning techniques, dynamic feature points can be identified and rejected by geometric methods such as optical flow and motion consistency estimation.

This paper proposes the DLD-SLAM algorithm to solve the positioning problem in dynamic environments. Based on the below work, our algorithm runs much more efficiently than open source algorithms with the same accuracy. The main work is as follows:

1. On the basis of the ORB-SLAM3 algorithm, the GCNv2 tiny network replaces the conventional ORB method to achieve the extraction and matching of feature points, which improves the efficiency and robustness of the system.
2. The lightweight GSConv [10] module is applied to the YOLOv5s network model, which reduces the count of parameters in the network to improve the computational efficiency of the target detection algorithm. Then, the target detection algorithm combines with the depth information of the RGB-D camera to obtain the mask of potential dynamic targets, which helps identify areas where dynamic feature points are located.
3. A novel method for rejecting dynamic feature points was designed based on the dynamic feature point rejection strategy. We propose the concept of dynamic probability based on LK (Lucas–Kanade) optical flow, semantic labels, and the state in the last frame which is added to the tracking thread. Using this method, the real dynamic feature points are rejected, and the static feature points are retained for position estimation. This method can effectively solve the problem of interference with positioning by dynamic objects.
4. Experiments are carried out for the above design: Firstly, feature point detection and matching are verified to prove the accuracy and robustness of the system; Then, the training accuracy and detection results of the lightweight target detection network are analysed. Finally, the performance of position estimation in the dynamic environment is verified by the TUM dataset, which has been demonstrated to improve the efficiency of our algorithm and the effectiveness of our approach when dealing with dynamic objects.

2. Related Work

2.1. Visual SLAM Based on Deep Learning

Current research has shown the potential of deep learning in several segments of Simultaneous Localisation and Mapping (SLAM), including front-end feature extraction, loop detection, and mapping. The following is a detailed review of the integration of visual SLAM with deep learning.

Traditional front-end feature extraction methods such as Scale-invariant Feature Transform (SIFT) and Speeded-Up Robust Features (SURF) have been widely used in SLAM systems. However, these methods are not stable enough to meet the application requirements in some complex scenes. In recent years, feature extraction methods based on deep learning have gradually emerged. These methods extract feature points and descriptors from images using convolutional neural network (CNN). MagicPoint [11] is an end-to-end position estimation network based on deep learning, but it is only applicable when partially regular graphics and poor migration capability are acceptable. D2-Net [12] is a robust CNN-based feature extraction method. It is able to extract stable feature points and descriptors in different scenarios by designing a CNN architecture based on local stability and repeatability evaluation. R2D2 [13] is proposed based on the D2-Net network. The R2D2 algorithm trains both keypoints and descriptors, using part of the detection network to compute an accuracy score for the feature points. SuperPoint [14] is a lightweight feature extraction method that employs an end-to-end training approach using a convolutional neural network for feature point detection and description. This method has low computational complexity while maintaining high performance. GCNv2 [15] is a proposed algorithm based on the GCN network, which obtains binarised feature points and descriptors through neural networks. Experimental results demonstrate that the above method is more accurate and stable than traditional feature extraction methods.

Traditional mapping methods frequently rely on point clouds, yet these methods are difficult to handle for complex scenes. There has been a growing popularity in the utilisation of deep learning-based methodologies for mapping in recent years. VINet [16] employs convolutional neural networks to extract the features of keyframes in an image sequence. Then, it obtains a global map using a convolutional neural network-based triangulation method.

A number of approaches have also emerged to improve the performance of SLAM algorithms by improving deep learning networks. Zhang, R [17] used ShuffleNetV2 to improve the YOLOv5 network. Meanwhile, to achieve semantic extraction in the environment, the segmentation head of the pyramid scene analysis network is added to the head of the YOLOv5 network, giving the improved YOLOv5 network both target detection and semantic segmentation capabilities. The use of YOLOv5 has also emerged as an approach to the problem of mapping in dynamic scenes [18].

In addition to the above methods, deep learning can be applied to different parts of the SLAM system. These studies demonstrate that the integration of deep learning and VSLAM enhanced system performance and offers a diverse array of practical applications.

2.2. Dynamic Visual SLAM

Dynamic visual SLAM can be mainly classified into geometry-based and deep learning-based approaches.

Among the geometry-based methods, Kim [19] modelled the background in the environment to eliminate the influence of moving targets and constructed a pixel-level background likelihood function based on the difference method between depth images. The image sequence can effectively separate the moving targets from the background region. It can be embedded into the DVO-SLAM system [20] to achieve real-time results. Fan [21] solved the problem of dense distribution and the high number of iterations of the standard RANSAC algorithm when selecting the inner points, as well as the polar constraints, to effectively filter out the dynamic points in the image. PFD-SLAM [22] uses a grid-based motion statistics method to ensure accurate matching with RANSAC.

It computes homography transformations to extract dynamic regions and uses particle filtering to accurately determine dynamic regions.

Among the deep learning-based approaches, DS-SLAM [23] is based on the ORB-SLAM2 algorithm. SegNet [24], a semantic segment network, combines with optical flow to construct semantic octree graph maps, which reduces the impact of dynamic objects. Detect-SLAM [25] uses SSD networks [26] to achieve semantic segmentation on keyframes and uses SSD networks on a GPU to solve the time efficiency problem. RDS-SLAM [27] adds a semantic thread to ORB-SLAM3, which also performs semantic segmentation on keyframes and updates the motion probability of feature points based on the segmentation results. DynaSLAM [28] is a visual SLAM system with dynamic object recognition and repair that uses Mask-RCNN [29] to detect and eliminate a priori dynamic objects and combines multi-view geometric constraints to locate undetected dynamic objects. However, the system cannot operate in real time due to the operational efficiency of this instance segmentation network. DP-SLAM [30] solves the problem of the inaccurate detection of dynamic points appearing at the boundary edges of the segmentation by determining the probability of moving points based on the previous frames and Bayes' law. YOLO-SLAM [31] extends ORB-SLAM2 with semantic segmentation and dynamic feature selection threads. The segmentation thread uses YOLOv3 to select known dynamic objects. The dynamic feature selection thread uses geometric-depth RANSAC to distinguish between dynamic and static feature points.

Compared with geometry-based dynamic visual SLAM systems, deep learning-based systems have higher accuracy and stability and can be applied to more complex environments. However, due to the deep learning network, it tends to reduce the efficiency of the system. Therefore, there is still ample opportunity for further research in the field of deep learning-based SLAM. To compensate for these shortcomings, deep learning combined with geometrically constrained SLAM methods has emerged [32,33]. To solve these problems, we propose a dynamic SLAM system, DLD-SLAM, based on feature point extraction by GCNv2-tiny, lightweight YOLOv5s, and a dynamic feature point rejection strategy.

3. Methods

DLD-SLAM is improved based on ORB-SLAM3 for RGB-D cameras, as shown in the flow chart in Figure 1. In addition to the original three threads of tracking, local mapping, and loop closing, a semantic segmentation thread is designed to identify dynamic regions.

As depicted in Figure 1, the RGB images and depth images are obtained from the RGB-D camera. The RGB images are simultaneously passed into the semantic segmentation thread for target detection and the tracking thread for feature extraction, while the depth images are only passed into the semantic segmentation thread to achieve the mask of dynamic objects in combination with target detection. In the semantic segmentation thread, target detection is performed using the GS-YOLOv5s algorithm. It preliminarily determines the potential dynamic elements by obtaining the semantic information and combining the depth images to obtain the mask of the dynamic elements. Different from traditional semantic segmentation, it refers to the depth information and can achieve the same effect as the semantic segmentation mask without using complex neural networks. GS-YOLOv5s is the target detection method as it performs lightweight processing. It improves the algorithm's efficiency without affecting its performance.

In the tracking thread, the feature points are extracted and matched using the GCNv2-tiny network. This method can be a good solution to the problem of difficulty with feature point extraction in dynamic environments and the difficulty of matching with large viewpoint changes. Furthermore, the dynamic feature point rejection strategy is designed, and potential dynamic objects are judged to determine whether they are real dynamic elements or not, and the dynamic feature points are rejected. The static feature points are retained for position estimation as well as for back-end optimisation, loop closure, and mapping. Using the strategies described above, we improve the system's accuracy, robustness, and efficiency.

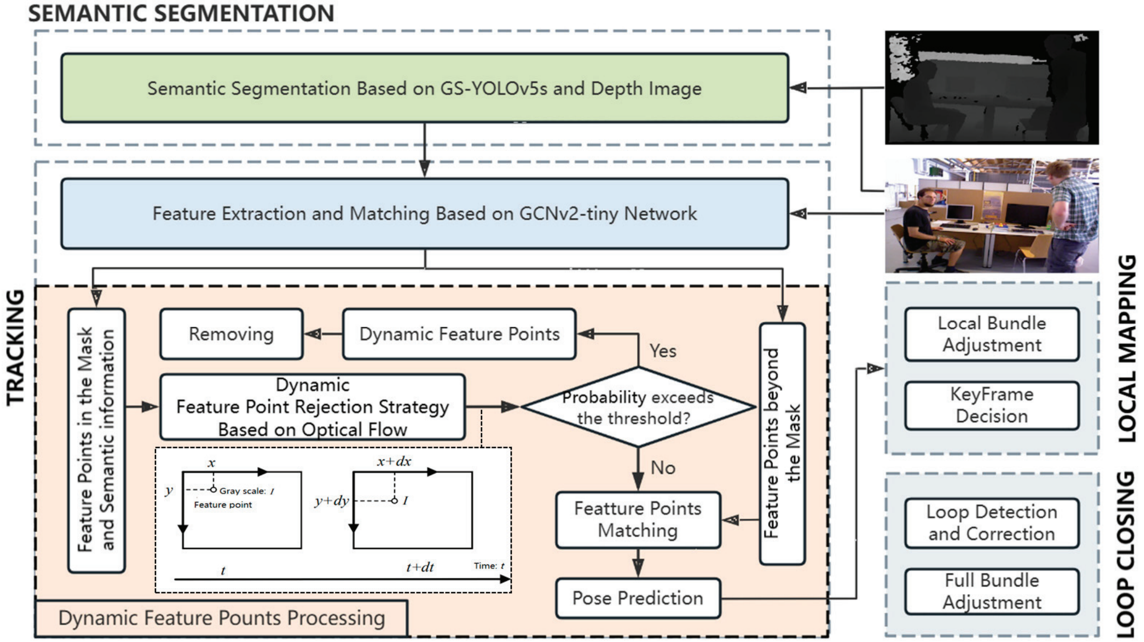

SEMANTIC SEGMENTATION
Semantic Segmentation Based on GS-YOLOv5s and Depth Image
Feature Extraction and Matching Based on GCNv2-tiny Network
TRACKING
Feature Points in the Mask and Semantic information
Removing
Dynamic Feature Points
Yes
Dynamic Feature Point Rejection Strategy Based on Optical Flow
Probability exceeds the threshold?
No
Feature Points beyond the Mask
Featture Points Matching
Pose Prediction
x
x+dx
y
Gray scale: I
Feature point
y+dy
I
t
t+dt
Time: t
Dynamic Feature Pounts Processing
LOCAL MAPPING
Local Bundle Adjustment
KeyFrame Decision
LOOP CLOSING
Loop Detection and Correction
Full Bundle Adjustment

Figure 1. Overview of DLD-SLAM. The algorithmic framework contains four threads: semantic segmentation, tracking, local mapping and loop closing. Section 3.1 is shown in blue; Sections 3.2 and 3.3 are shown in green; and Section 3.4 is shown in orange to express the process of dynamic feature points rejected.

3.1. Feature Extraction and Matching Based on GCNv2-tiny Network

The feature points in the ORB-SLAM3 system are ORB (oriented FAST and rotated BRIEF) feature points. The method of extracting feature points in each frame is as follows: Firstly, an image pyramid of decreasing resolution is constructed by down-sampling. Then, each layer of the image is divided into a 30 × 30 pixel grid, and the feature points are extracted and homogenised in the grid. Finally, the extraction of ORB feature points is repeated for each layer of the image. This method leads to an increase in computation, which reduces the real-time performance of the system. To improve the real-time performance of the system, we employ the GCNv2-tiny network to extract image feature points and descriptors, which replaces the original ORB feature points and is ported to ORB-SLAM3. Figure 2 depicts the process of extracting feature points using the GCNv2-tiny network.

As depicted in Figure 2, firstly, the original input image is segmented into a 16 × 16 pixel grid for separate prediction, which ensures that the feature points are evenly distributed throughout the image. The GCNv2-tiny network shares the convolutional network from conv1 to conv4 to encode image features. Compared to the GCNv2 network, the GCNv2-tiny network halves the number of convolution channels in conv3 and after conv3 while maintaining the same convolution kernel size and step size. This compresses the parameters to reduce the amount of computation, which helps to improve real-time performance. Subsequently, the convF convolutional network is employed for decoding. The sub-pixel convolution is employed to obtain both the position data of the feature points and the probability distribution graph showing the confidence degree. The decoding process employs a convolutional network known as convD. The corresponding descriptors are obtained using the bilinear interpolation method. Finally, the feature points are acquired using the procedure of Non-maximum Suppression (NMS) followed by the identification of corresponding descriptors through indexing.

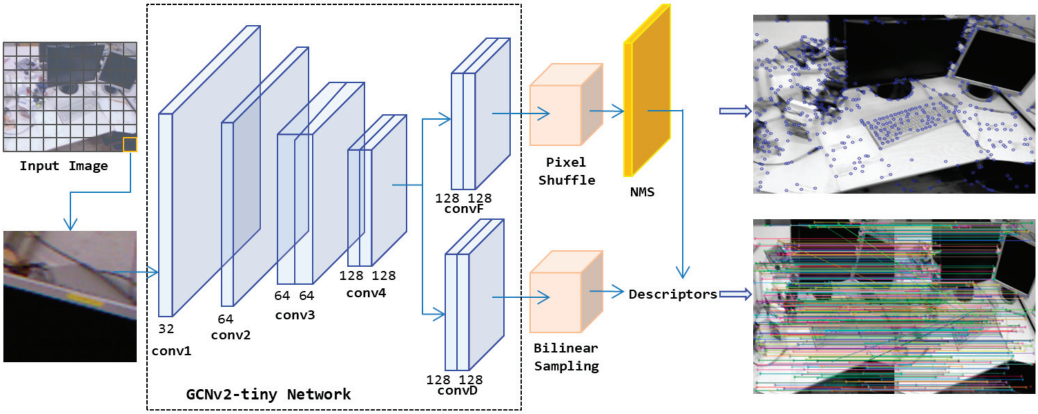


Figure 2. The procedure of extracting feature points based on GCNv2-tiny network. The number under the convolutional layer in the figure is the number of channels. The two figures on the right show the results of feature point extraction and matching.

The GCNv2-tiny network predicts both keypoints and descriptors. The network outputs a probability graph of the keypoint confidence degree and a dense feature graph of descriptors. The probability graph of feature points is binary and the points only take values between 0 and 1. This process becomes a classification problem, which gives better training results. Moreover, this is the same as the feature extraction in ORB-SLAM3, so it can be well ported to ORB-SLAM3. The size of a binary feature vector is set to 256 so that the descriptor has the same bit width as the ORB features and so that the descriptor can be directly embedded into the existing ORB-based visual SLAM. The purpose of the network training aims to match the positions of the points with the value of 1 in the probability graph of the feature points in the final output as closely as possible to the corresponding real feature points in the training dataset. At the same time, a binary activation layer is added at the end of the network to obtain binary descriptors.

By introducing the attention mechanism into the feature encoding stage, more specific feature points can be obtained, which can improve the robustness of feature matching. The GCNv2-tiny network was trained using the SUN3D [34] dataset, which provides information about the camera's motion position. The network uses sample data as well as a loss function, and it contains not only the true positions of the feature points but also the true feature matching relationships obtained from the real position. And the position information in the dataset is the main training base. Therefore, the feature point matching is more stable when the larger view changes, which is also beneficial to the position estimation based on feature points.

The feature point extraction and matching used in this paper are conducive to improving the real-time performance of the system, and the extracted feature points are more suitable for high-precision position estimation. This algorithm is more stable when dealing with large perspective changes and dynamic object disruptions.

3.2. Lightweight YOLOv5 Target Detection Algorithm

YOLO [35] is a one-stage target detection algorithm that has a simple structure and superior performance in both detection speed and detection accuracy. YOLOv5 can be divided into several different architectures depending on the number of convolutional kernels in the network. YOLOv5 contains five architectures: YOLOv5x, YOLOv5l, YOLOv5m, YOLOv5n, and YOLOv5s. The number of floating point operations and the value of model parameters decrease for these five models in the given order. In this paper, we choose YOLOv5s, which has a smaller number of model floating-point operations and model parametric quantities, because this architecture satisfies the system's detection accuracy with less computation required as well as a faster rate of model training. Based on the YOLOv5s algorithm, it is lightweighted in this paper.

The YOLOv5s model is mainly composed of Backbone, Neck, and Head, which includes a large number of convolutional neural network (Conv), spatial pyramid pooling fast (SPPF), and CSP [36] modules. The role of Neck is to perform multi-scale feature fusion processing on the feature graph and then to pass the feature graph to the prediction layer, which requires a large number of parameters and computations. In order to reduce the computational effort of the model and build a lightweight network, the convolution network GSConv is added to the CSP2_x module, which is mainly used in the Neck stage. GSConv replaces the standard convolution network with depthwise convolution (DWConv), which reduces the number of parameters and computation by about half compared to the standard convolution. The network improves the efficiency of training and running the model while maintaining a comparable level of performance.

The original input image should be transformed into a multi-layer feature map in Backbone and subjected to feature extraction. Different layers of features are fused together in the Neck module using up-sampling and down-sampling operations. This produces feature graphs with multi-scale information. During these operations, each spatial compression and channel expansion of the feature graph results in the loss of some detection information. Dense convolutional maximises the retention of hidden connections between each channel. Then, a smaller channel is placed in the centre of the GSConv model. This reduces the amount of additive computation while ensuring that effective detection information is delivered. However, due to the deeper layers of the GSConv network model, the deeper network increases the resistance of the data flow. It is therefore not suitable for use in all stages of YOLOv5s.

In Neck, the processed feature graphs are maximised in the channel dimension and minimised in the width and height dimensions. This is a better fit to the structure of the GSConv model as it does not require any transformation to be performed. There is an improvement in performance without a reduction in the efficiency of network inference due to the deepening of the layers. Therefore, a better choice is to use GSConv only at the Neck stage. As shown in Figure 3, the GSConv network is used in the CSP2_x module in the YOLOv5s network, and the GSbottleneck module is added based on GSConv. Based on this, the Neck module is reconstructed, keeping the original structure of the Backbone and Head modules unchanged. In Neck, the multi-scale fusion feature graphs processed by GSConv are less redundant and do not need to be compressed. In addition, the attention module is more effective.

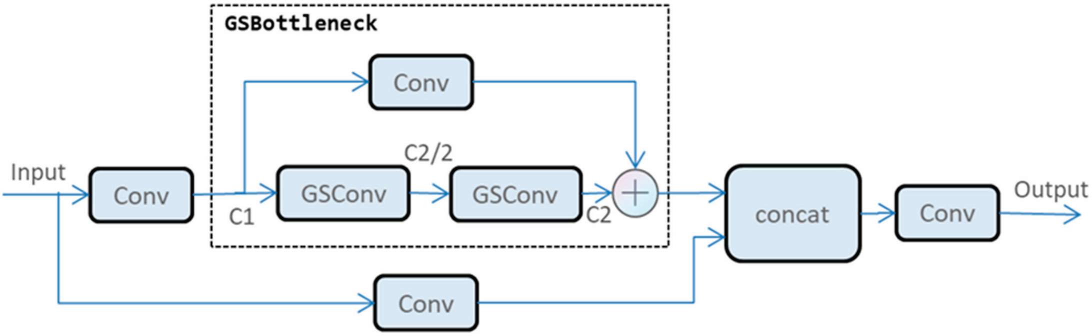


Figure 3. The CSP2_x module structure in GS-YOLOv5s network.

3.3. Dynamic Object Detection Based on Target Detection and Depth Image

The target detection algorithm used in this work is GS-YOLOv5s. This algorithm can only initially determine the potential dynamic objects by semantic labels and represent the dynamic objects with a rectangular detection box. However, if the coverage area of the detection box is large while the proportion of the area occupied by the dynamic objects in that detection box is small, it will result in the rejection of too many static points in the background. This will lead to insufficient static feature points for position estimation and affect the positioning accuracy of the system. Therefore, we adopt the method of combining target detection and depth information to further accurately determine the area of dynamic elements in the image. This is similar to obtaining a mask by semantic segmentation and maximises the retention of static feature points.

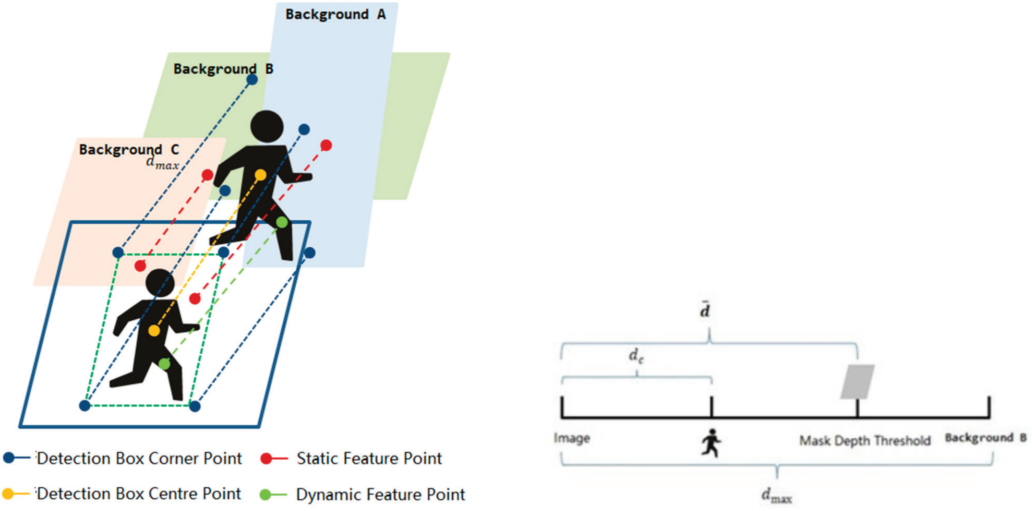
Most dynamic objects have some depth difference from the background. In addition, dynamic objects tend to occupy the centre of the detection frame. The four corners of the detection frame basically correspond to the static background points. Figure 4 demonstrates the positional relationships of dynamic objects in space and the meaning of every element.

$$d_{\max} = \max(d_{tl},\ \ d_{tr},\ \ d_{bl}, d_{br}), \tag{1}$$

where d_{tl}, d_{tr}, d_{bl}, and d_{br} represent the depth values corresponding to the four corner points of this detection frame. The depth of a pixel is assumed to be represented by d. The maximum background depth of a dynamic object is $d_{\max}$. The reason for not selecting the minimum depth is to avoid feature points near or in front of the dynamic element causing the dynamic mask detection to fail.

$$\overline{d} = d_c + \epsilon, d_{\max} - d_c > \epsilon, \tag{2}$$

where d_c represents the depth value corresponding to the centre of the detection frame. The values of d_c and $d_{\max}$ is used to determine the threshold $\overline{d}$ that distinguishes the foreground dynamic feature points from the background static feature points. $\overline{d}$ represents the mask depth threshold. The value of ϵ is a pre-determined distance based on empirical values of the frequently occurring positions of dynamic objects in the scene.



(**a**)

(**b**)

Figure 4. This shows the principle of semantic segmentation. (**a**) Background represents the static objects in the background: A is the static background closer to the corresponding dynamic object in the detection box, and B is the static background further away from the corresponding dynamic element in the detection frame. Different classes of feature points are represented by different colours. (**b**) Description of distance of various elements in Figure (**a**) in the camera coordinate system along the z-axis.

The process of obtaining the mask of dynamic objects is as follows: Firstly, we set the depth value of a depth image, but not the target detection frame, to 0. We then set a mask region with the size of the detection frame area where the depth value is $\overline{d}$. And the threshold value ϵ is to ensure that the depth position of the mask region is between the dynamic object and the background. Then, the pixel depth value in the detection box is compared to $\overline{d}$. If it is less than $\overline{d}$, the feature is considered to be in a dynamic region. Otherwise, it is considered a static region, and the depth value is set to 0. The result is a mask that serves the same purpose as semantic segmentation for a more accurate rejection of dynamic feature points.

As shown in Figure 5, combining the target detection results with the depth information from the RGB-D camera can obtain results comparable to that of semantic segmentation. Due to the complexity of the semantic segmentation neural network, it will be more computationally intensive, leading to an increase in running time. This may affect the real-time performance of the system. However, our method solves the above problems while ensuring the accuracy of the system.

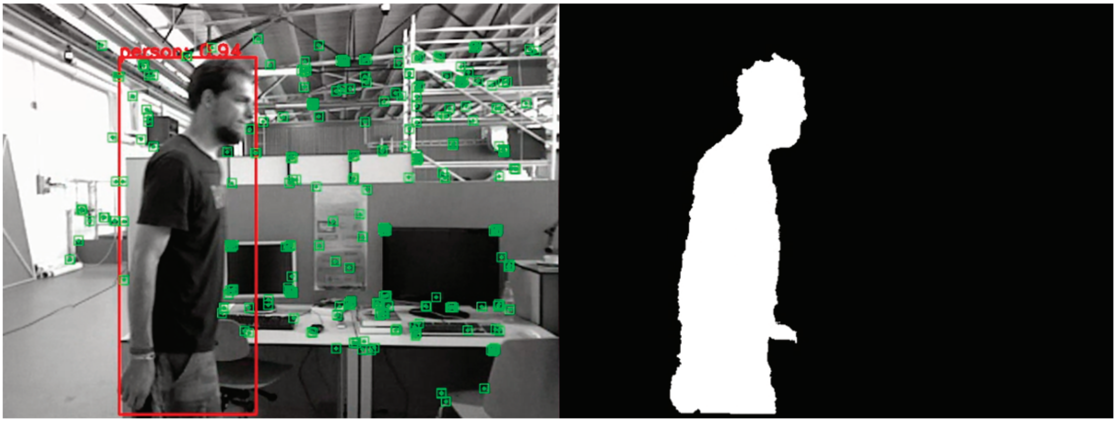


Figure 5. The result of semantic segmentation based on our method. The red represents the target detection box; the green represents static feature points in the keyframe. The depth value of white area is 1, while that of the black area is 0.

The method is used to obtain a semantic segmentation mask with the contour of the target object and to avoid culling dynamic feature points by over-culling static feature points in the detection box. Although there are open source semantic segmentation algorithms with this capability, they often do not run in real time. Our approach, by combining a target detection algorithm with depth information, makes full use of the sensor information and also improves the running efficiency of the algorithm.

In a word, the depth information is used to determine the depth in space of all objects in the detection box. The depth of the dynamic objects, d_c, and the depth of the static background, $d_{\max}$, are determined. The position of the mask is determined between the depth values d_c and $d_{\max}$. It can obtain the mask of the object, which separates dynamic objects in the foreground from static backgrounds by setting the value of ϵ in the detection box.

3.4. Dynamic Feature Point Rejection Strategy Based on Optical Flow

Target detection using the GS-YOLOv5s algorithm is combined with RGB-D depth images to obtain candidate masks of dynamic objects, which are sorted by semantic labels. The real static, potential, or real dynamic objects are then further sorted by dynamic probability based on the optical flow and semantic information, and then the feature points within the dynamic object mask are removed. In real environments, there are different kinds of objects; some objects have the ability to move autonomously, such as people, animals, etc., and they are usually in a state of motion, which is called dynamic objects, while objects such as tables, sofas, etc., usually do not move and are therefore called static objects. The potential dynamic objects are mainly objects that have a greater probability of being moved, such as books and chairs. This method will reduce the false rejection rate and improve the accuracy of the system in dynamic environments.

The optical flow can compare the motion of pixel points in adjacent frames. When processing the dynamic objects in a keyframe, the pixel points on the same dynamic object have the same motion trajectory. Therefore, the motion information of the whole dynamic object can be obtained by judging the motion of one pixel point in the dynamic object. In our method, the LK (Lucas–Kanade) optical flow method [37] is used. The LK optical flow method is a method based on the local lightness change of the image. The LK optical flow belongs to the sparse optical flow, which can ensure the performance effect while the computational amount is small.

In real scenes, the motion state of objects often changes. To address these problems, we design the dynamic probability and update the probability with subsequent data, which can more accurately distinguish between dynamic and static regions. In order to make the recognition of dynamic features robust, the dynamic probability is based on the semantic information, the relative velocity constraints are based on the optical flow, and there are three ways to update the feature point state in the last frame. The dynamic probability of the feature point update formula is as follows:

$$P_j(X_i) = \alpha_1 S_j(X_i) + \alpha_2 M_j(X_i) + \alpha_3 K_{j-1}(X_i) \tag{3}$$

where X_i represents a feature point on the *j*-th frame. $P_j(X_i)$ represents the probability that the feature point X_i in the current frame is a dynamic feature point. $S_j(X_i)$ represents the feature points classified by the semantic information which are located in the static, potential dynamic, and dynamic object masks. $M_j(X_i)$ represents the result of the verification using the optical flow method. The result that satisfies the threshold takes the value of 1; otherwise, it is 0. $K_{j-1}(X_i)$ represents the result of the verification of the feature point that matches the feature point X_i in the last frame and takes the value of 1 if it has been judged to be a dynamic feature point in the last frame; Otherwise, it takes the value of 0. The value is also set to 0 if the matching feature point is not searched. The values of α are weighting factors and satisfy the constraint that $\alpha_1 + \alpha_2 + \alpha_3 = 1$ where the values of $\alpha_1,\ \alpha_2, \alpha_3$ can be adjusted according to the specific environment.

The feature points within the mask are identified by the dynamic feature points waiting to be determined. It is assumed that the set P contains all feature points in the mask. Set D is the set of dynamic feature points, and it set the value of $S_j(X_i)$ to 1; set H is the set of feature points within the potential dynamic object mask, and it sets the value of $S_j(X_i)$ to 0.5; and set S is the set of static feature points, and it sets the value of $S_j(X_i)$ to 1. Furthermore, set P can be expressed as $P = D \cup H \cup S$.

The results of $M_j(X_i)$ are based on the optical flow method. Each feature point has coordinate and velocity information, which are calculated using the feature extraction algorithm and the optical flow method. In a real scene, the motion of dynamic objects will generate optical flow, and the background will also generate optical flow with the motion of the camera. Therefore, the average motion velocity of all static feature points in the set S needs to be calculated first for the accurate rejection of dynamic feature points. The motion velocity of static feature points is calculated as follows:

$$\begin{bmatrix} U \\ V \end{bmatrix} = \frac{1}{N} \sum\nolimits_{k=1}^{n} \begin{bmatrix} u_k \\ v_k \end{bmatrix}, \tag{4}$$

where U and V are the average motion velocity of the static feature point along the X- and Y-axis.

The motion velocity of the static feature points is used to identify all feature points in the potential dynamic object masks within set P. The equation for determining the results of $M_j(X_i)$ can be expressed as

$$\sqrt{(u_k - U)^2 + (v_k - V)^2} > t,\ k = 1, 2, \cdots n, \tag{5}$$

where t is a self-adaptive threshold.

If the motion velocity of the feature point is beyond this threshold, the value of $M_j(X_i)$ is considered to be 1, otherwise the value is 0.

Finally, we determine whether a feature point located in a mask is truly dynamic by the probability $P_j(X_i)$. The feature point which is determined to be a dynamic feature point will be rejected and will not be involved in the position estimation. The method used to reject dynamic feature points is shown in Algorithm 1.

Algorithm 1: Dynamic feature points rejection algorithm.

```
Input: Current frame's feature points $X_i$ in the semantic segmentation mask, $X_i \in$ P;
Output: The set $\mathrm{S}_{real}$ including all current frame's static feature point;
1 Obtain all feature points in the semantic segmentation mask belong to the set P;
2 Classify feature points in the set P according to the mask's semantic labels, $X_i^{d_m} \in$ D, $X_i^{h_m} \in$ H,
  and $X_i^{s_m} \in$ S, and assign semantic labels to every feature point, $S_j\left(X_i^{d_m}\right)$=1, $S_j\left(X_i^{h_m}\right)$=0.5, and
  $S_j\left(X_i^{s_m}\right)$=0;
3 for each feature point $p_i \in$ P do
4   Calculate the mean motion velocity $\begin{bmatrix} U \\ V \end{bmatrix} = \frac{1}{N}\sum_{k=1}^{n}\begin{bmatrix} u_k \\ v_k \end{bmatrix}$ of all the feature points, $X_i^{s_j}$,
      and calculate the velocity, $V(p_i)$= $\sqrt{(u_i - U)^2 + (v_i - V)^2}$;
5     if $V(p_i) > t$ then
6       $M_j(p_i)$=1;
7     else $M_j(p_i)$=0;
8     end if
9   Follow the method in step 5~8, judge the state of $p_i$ in last frame, and obtain the value of
      $K_{j-1}(p_i)$;
10  Calculate the dynamic probability of $p_i$, $P_j(p_i) = \alpha_1 S_j(p_i) + \alpha_2 M_j(p_i) + \alpha_3 K_{j-1}(p_i)$;
11    if $P_j(p_i) < \varepsilon$ then
12      Save the feature points $p_i$ to the set $\mathrm{S}_{real}$;
13    else reject the feature point $p_i$;
14    end if
15 end for
```

In step 2, $X_i^{d_m}$ represents the i-th feature point in the m-th dynamic detection box, $X_i^{h_m}$ represents the i-th feature point in the m-th potential dynamic detection box, and $X_i^{s_m}$ represents the i-th feature point in the m-th static detection box.

4. Experimental Results

This section presents experimental details to validate the proposed DLD-SLAM system. In order to evaluate and analyse the proposed DLD-SLAM system, the experiments are performed on a laptop with the following specifications: Intel Core i9-13800H processor, 16 GB RAM, NVIDIA GEFORCE GTX-4600, and 8 GB graphics memory which installed on Lenovo ThinkBook 16 computer. And the operation system is Ubuntu 20.04. The proposed DLD-SLAM algorithm is compared and studied in regards to three aspects: feature extraction and matching, target detection algorithm, and the positioning accuracy of the system. Through the experimental comparison, we demonstrate the superiority of the proposed method.

4.1. Feature Extraction and Matching

In order to verify the feature point extraction and matching effect of the GCNv2-tiny network, we selected the v_bird sequences in the HPatches dataset for the experiments comparing the matching effect of the GCNv2-tiny feature points and ORB feature points. The HPatches dataset is mostly used in the evaluation of image matching and description tasks of transformations of scale, view angle, luminance, etc. And each sequence has the exact corresponding feature points and the truth of the transform parameters. It can be used for the validation of the effectiveness of feature point extraction and matching methods.

The v_bird sequence within the HPatches dataset has some images from different viewpoints of the same scene. Figure 6 illustrates the impact of GCNv2-tiny and ORB feature extraction and matching across various viewing angles. When the viewing angle is not changed much, ORB has a good matching effect, but when the viewing angle is changed a lot, ORB has almost no matching effect. The GCNv2-tiny model demonstrates a robust capacity for matching. Regardless of variations in the viewing angle, the feature points can be matched. The results are presented in Table 1.

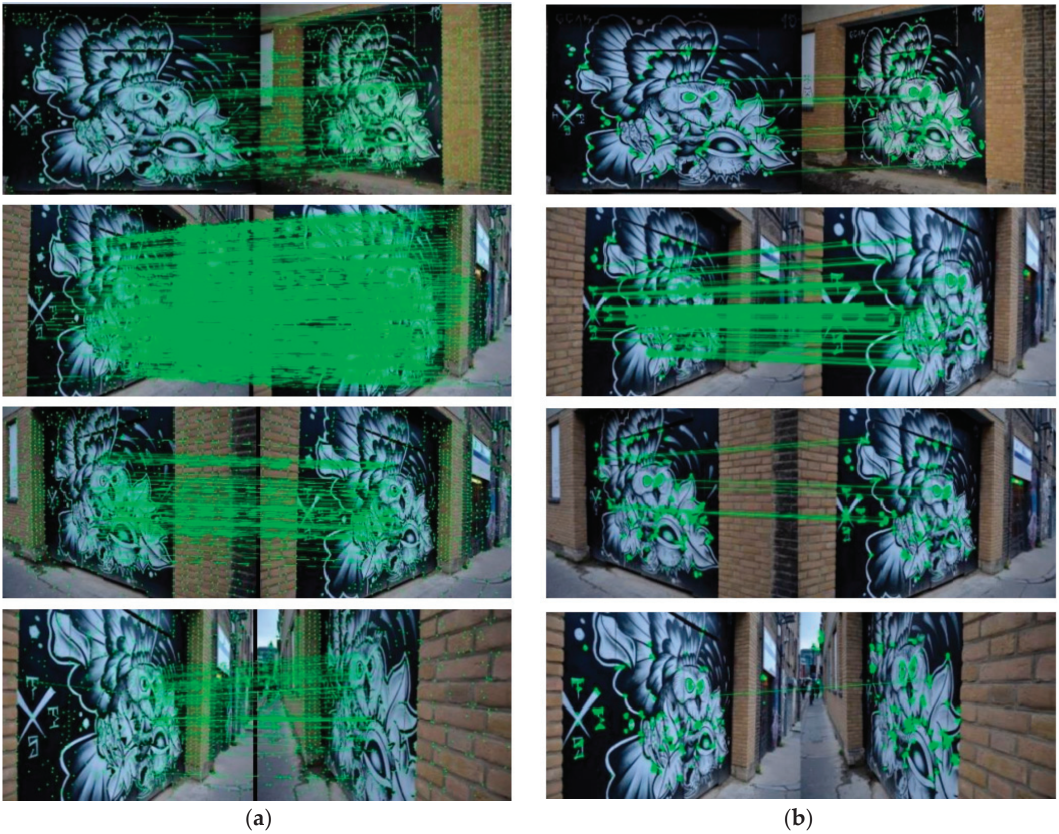


Figure 6. This figure shows the performance of feature point extraction and matching. From top to bottom, the scenes shows the different perspectives (1~4). (**a**) The results of the GCNv2-tiny method; (**b**) the results of the ORB method.

In Table 1, correct matches refer to the initial matches after the RANSAC algorithm has removed the false matches. The rate represents the proportion of correct matches to the number of initial matches. For small changes in perspective, the GCNv2-tiny feature point shows a significantly higher correct match rate over ORB. It can be concluded that the GCNv2-tiny feature points are more adaptable and robust in complex scenes with large perspective changes.

Table 1. The results of feature point extraction and matching by different viewing angles by two methods.

Method	Perspective	Initial Matching	Correct Matching	Rate (%)
ORB	1	329	46	13.98
	2	374	97	25.94
	3	261	17	6.51
	4	302	5	1.66
GCNv2-tiny	1	**341**	**124**	**36.36**
	2	**563**	**386**	**68.56**
	3	**367**	**89**	**24.25**
	4	**435**	**92**	**21.15**

The results of each method in this paper are bolded.

4.2. Target Detection Network Training and Performance

To verify the performance of the GS-YOLOv5s dynamic target recognition algorithm in this paper, the COCO dataset is used for training and validation. The training model for GS-YOLOv5s is performed first. The parameters of the experiment are set as follows: the network training batch size is 128, the initial learning rate is 0.01, the weight decay coefficient is 0.0005, and the total number of training rounds is 200. During the training process, the dataset is divided into three parts: the training set, the validation set, and the test set. The role of the training set is to train the model to obtain the parameters to make the model fit the data, the role of the validation set is to evaluate the performance of the model and select the best hyperparameters to avoid overfitting the model; and the role of the test set is to test the model's ability to generalise and to evaluate its performance on an unknown dataset. Figure 7 illustrates the progression of loss values throughout the training process. The accuracy of target detection and classification increases as the loss value decreases, approaching the true value.

As seen from Figure 7, as the number of training rounds increases, the loss decreases and tends to stabilise, and the model gradually converges. The target detection effect of the GS-YOLOv5s model in this paper is comparable than that of YOLOv5s. Combined with the loss curves of the validation set, it can be seen that the model does not show any overfitting phenomenon when applied to the validation set. The model is trained to learn effective features and does not rely too much on the training data.

From Figure 8, the fluctuations in precision and recall gradually decrease and become stable as the number of training rounds increases. And the precision of the YOLOv5s model fluctuates more during the training process. This indicates that its training effect is not as good as the GS-YOLOv5s model in this paper. However, it can be seen from its precision curve that its prediction effect is slightly higher than that of the model in this paper. The corresponding mAP performance indices are presented in Table 2 and were used to evaluate the training results.

Table 2. The comparison of the two models in performance parameters.

	YOLOv5s	GS-YOLOv5s	Promotion Rate (%)
mAP_0.5	94.291	**93.473**	−0.87
mAP_0.5:0.95	75.689	**79.418**	4.93
FPS	112	**135**	20.54
Params (M)	15.2	**12.7**	16.45
FLOPs (ms)	15.6	**13.2**	15.38

The results of method in this paper are bolded.

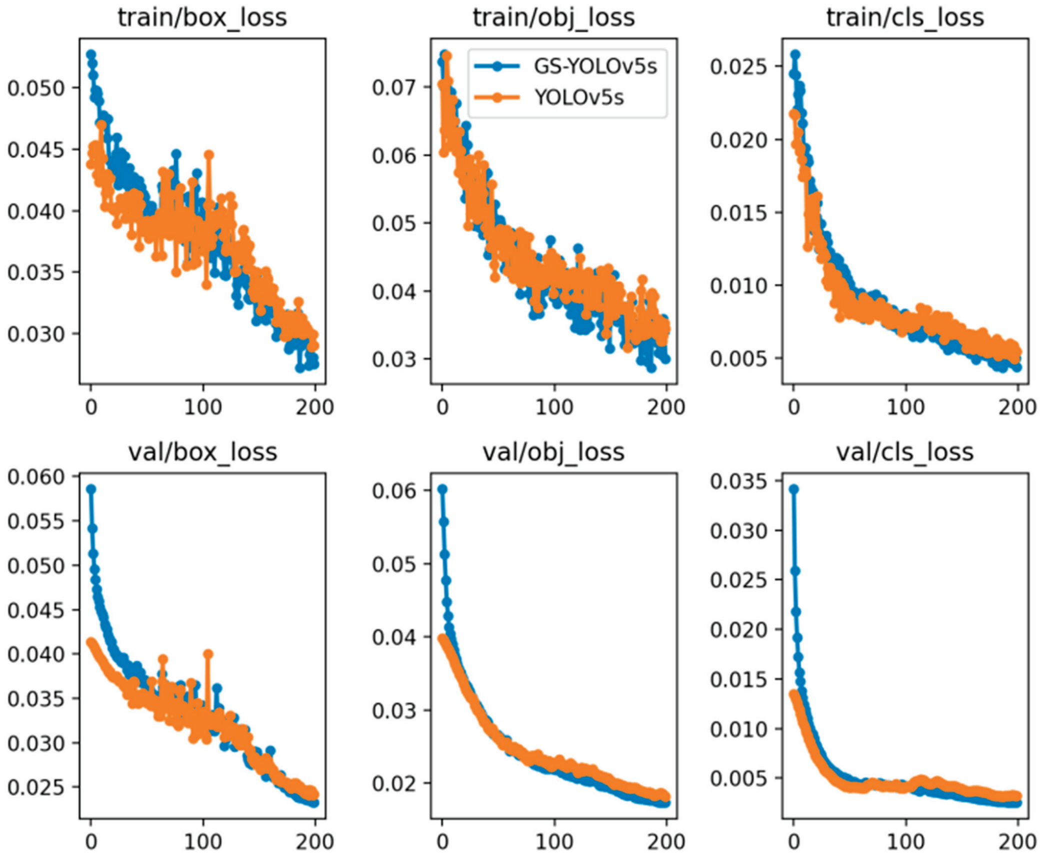


Figure 7. Plot of training loss variation for GS-YOLOv5s model. The variable box_loss is the bounding box regression loss; obj_loss is the confidence loss; cls_loss is the classification probability loss. The variable train represents the loss change in the training set, while val represents the loss change in the validation set.

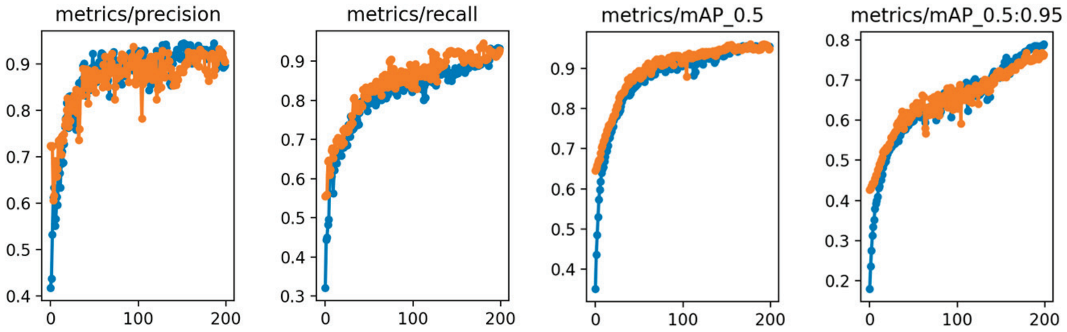


Figure 8. Plot of changes in performance metrics for GS-YOLOv5s model. The changes in precision, recall, and mAP of GS-YOLOv5s model during the training process. The variable mAP_0.5 refers to a mAP with IoU threshold larger than 0.5; mAP_0.5:0.95 denotes the average mAP at different IoU thresholds (from 0.5 to 0.95, with a step size of 0.05).

Table 2 presents a comparison between the two models in terms of performance parameters. The variable mAP is used to measure the prediction accuracy, and a higher the value represents a better performance. FPS is used to measure the running efficiency; a larger value represents a faster rate. Params is used to measure the model size; the smaller the value, the lighter the model. FLOPs is used to measure the speed of the algorithm; a smaller value represents a faster speed. It can be seen that GS-YOLOv5s is slightly inferior to the original algorithm in accuracy, but the running efficiency is significantly improved over the original algorithm. The proposed methodology demonstrates a 20.54% enhancement in speed, while accompanied by a 0.87% decrease in accuracy. Therefore, the algorithm proposed in this paper improves the running speed by reducing the accuracy slightly to enhance the real-time performance of the system.

The experimental results of the comparison are depicted in Figure 9. The majority of objects within the environment can be effectively detected, although the level of accuracy in detection is not as high as that achieved by YOLOv5s. The classification results are also slightly worse than those of the original algorithm. However, the target detection algorithm in this paper is used to further identify the dynamic objects and does not require strict semantic information for classification. The method employed in this study demonstrates a high level of accuracy in target detection, thereby satisfying the system's requirements.

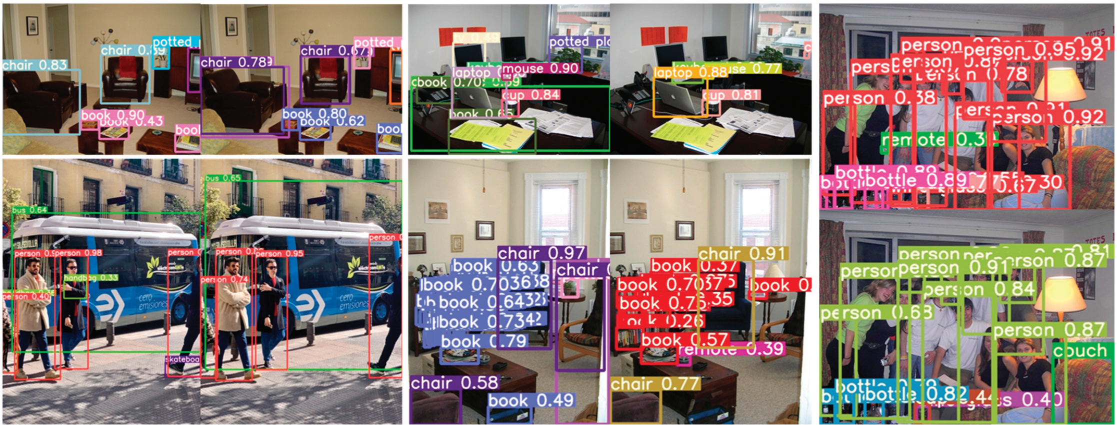


Figure 9. The comparison of the detection results of the two algorithms on the test samples. The left and right side show the comparison of target detection results between YOLOv5s and GS-YOLOv5s. In the last figure, the top is YOLOv5s's results, and the bottom is GS-YOLOv5s's.

From Figure 10, the mask of dynamic target objects is obtained. In Figure 10a,b, we mark the defects of the detection results by the yellow frame. In Figure 10a, this result is due to the fact that the dynamic object is too close to the neighbouring background, and the depth values of both are close enough to cause the distinction to fail. In Figure 10b, this result is due to the fact that the dynamic object is close to the camera. In the results of all frames, the defects are very rare, and they tend to take up a tiny area of the mask, which will not affect the performance of the system.

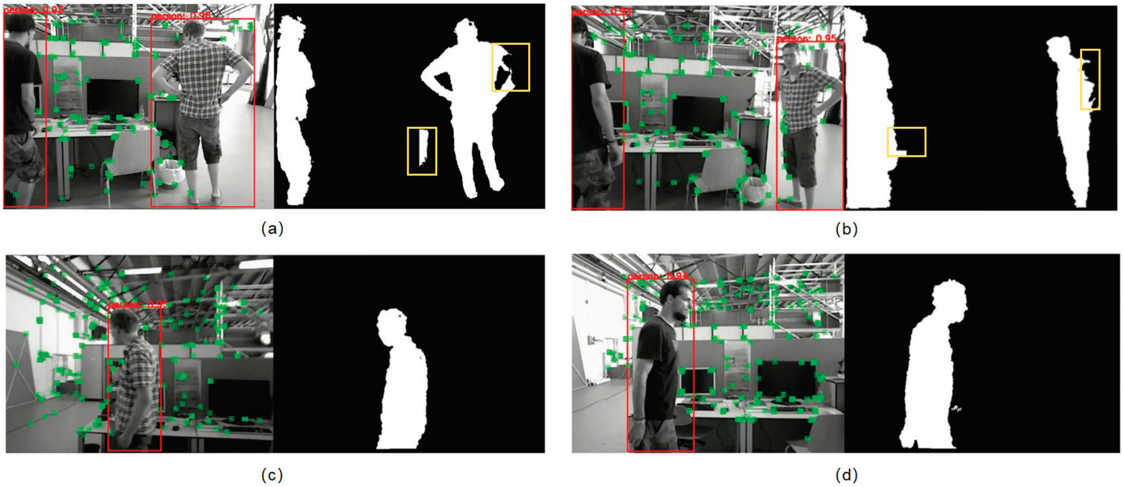


Figure 10. The mask of dynamic target object obtained by combining GS-YOLOv5s with depth images. The red box is the detection box; The green points are the feature points by the GVNv2-tiny method. The yellow box marks the defect of results in the mask. The Figure (**a**–**d**) are representative of a few frames from all results. The yellow boxes of (**a**,**b**) in the images represent flaws, while (**c**,**d**) are the results with better performance.

4.3. Trajectory Accuracy Verification Experiment in the Dynamic Environment

4.3.1. Trajectory Accuracy

The main metrics used to evaluate trajectories in Visual SLAM are absolute trajectory error (ATE) and Relative Pose Error (RPE). The ATE metric is used to measure the difference between the camera pose estimated by the algorithm and the true camera pose and is used to calculate the global consistency of the trajectory. The ATE metric is used to measure drift in the visual odometer system. And the smaller the ATE evaluation metric, the higher the accuracy. It is assumed that the visual odometry estimate of the position is denoted as $P_1, P_2, \cdots P_n \in SE(3)$. $SE(3)$ is the three-dimensional Euclidean transformation consisting of rotations and shifts. The true bit positions of the camera are denoted as $Q_1, Q_2, \cdots Q_n \in SE(3)$. These positions represent the camera's position relative to the world coordinate system at different times or in different frames. The definition of the ATE formulation is given below.

The absolute trajectory error at frame *i* is defined as

$$F_i = Q_i^{-1} S P_i, \tag{6}$$

The root mean square error (RMSE) formula for ATE is defined as

$$\text{RMSE}(F_{1:n}) = \sqrt{\frac{1}{m}\sum_{i=1}^{m} \parallel \text{trans}(F_i) \parallel^2}, \tag{7}$$

where the trans(.) denotes the translational component of the relative trajectory error.

This work focuses on evaluating the trajectory accuracy using ATE.

The TUM_RGBD dataset, an open source dataset provided by the Technical University of Munich in Germany, is used to validate the robustness of the algorithm and the positioning accuracy in dynamic environments. The dataset comprises a total of 39 image sequences depicting various indoor environments. The contents of the dataset encompass both RGB images and depth images. The sitting and walking sequences are dynamic environments. Specifically, the sitting sequence includes two people positioned in front of a table engaging in small-amplitude movements. On the other hand, the walking sequence involves two people performing larger-amplitude movements as they move around a table. Furthermore, these two dynamic subsequences can be associated with four distinct categories of camera motion. There are four different camera movements that can be observed: (1) halfsphere: the camera moves along the hemisphere with a diameter of 1 m; (2) xyz: the camera moves along the x, y, and z axes; (3) rpy: the camera rotates along the roll, pitch, and yaw axes; and (4) static: the camera is stationary.

The fr3-walking-xyz, fr3-walking-rpy, fr3-walking-halfsphere, and fr3-walking-static dataset sequences from the highly dynamic environment in the TUM dataset are chosen as the test data. And we compare DLD-SLAM with ORB-SLAM3 to demonstrate the effectiveness of dealing with dynamic interference. Furthermore, the EVO evaluation tool is employed to measure the absolute trajectory error, which quantifies the difference between the estimates provided by the SLAM system and the actual values obtained from the dataset. Figure 10 displays the ATE of the ORB-SLAM3 and DLD-SLAM methods. The red line indicates the distance between the true value and the estimated value, commonly referred to as the absolute trajectory error (ATE). The results of localisation in the TUM dataset are depicted in Figures 11 and 12.

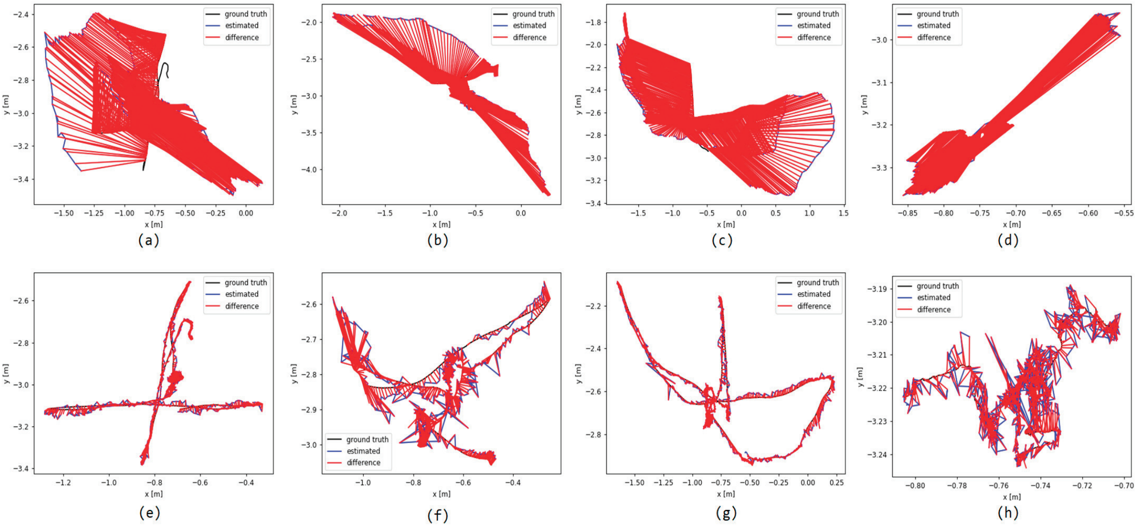


Figure 11. The ATE of the fr3_walking_xyz, fr3_walking_rpy, fr3_walking_halfsphere, and fr3_walking_static sequences from left to right. (**a**–**d**) are the ATE plots of the ORB-SLAM3 algorithm. (**e**–**h**) are the ATE plots of DLD-SLAM.

It is evident that DLD-SLAM has a substantial impact on enhancing the precision of localisation in visual SLAM systems operating within dynamic indoor environments.

The experimental findings, as presented in Table 3, indicate that DLD-SLAM demonstrates significantly enhanced accuracy in comparison to ORB-SLAM3; the ATE is greatly reduced. The RMSE is reduced by 97.29%, 94.71%, 96.89%, and 98.61% for the fr3-walking-xyz, fr3-walking-rpy, fr3-walking-halfsphere, and fr3-walking-static sequences. And the robustness of the camera's various irregular movements performs better when dealing with different sequences.

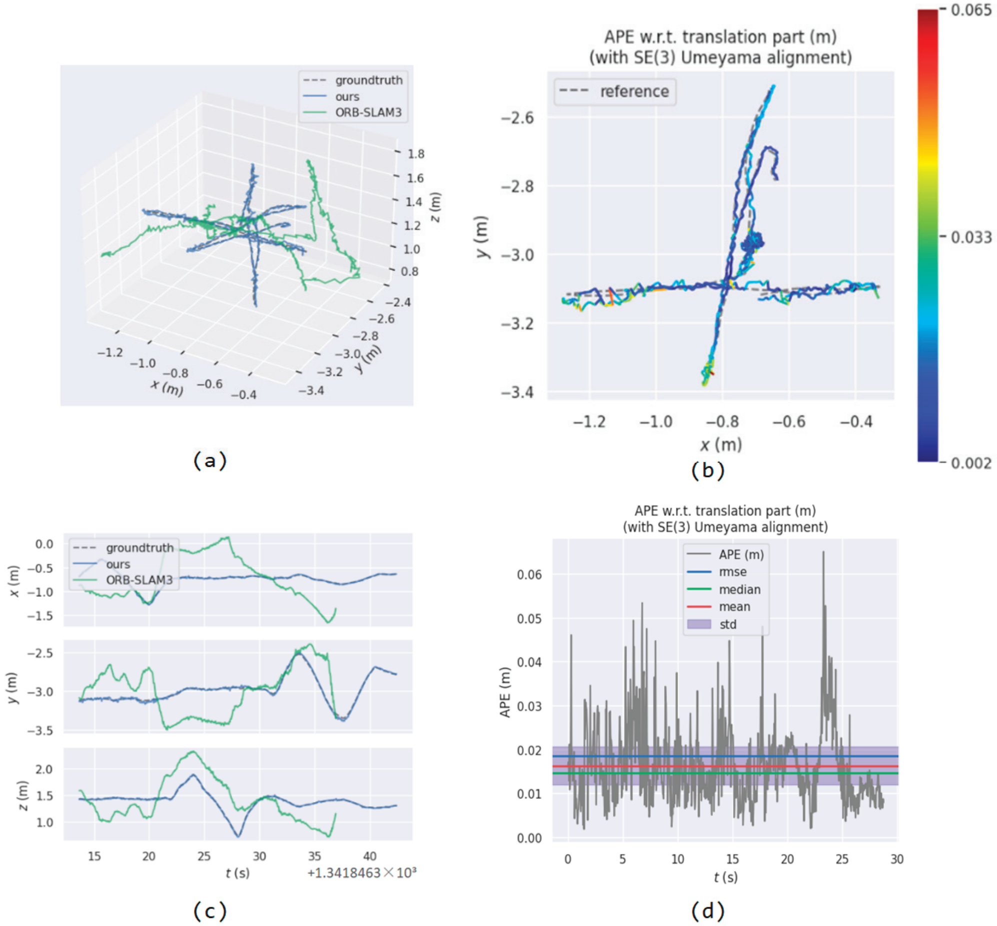


Figure 12. Taking the fr3_walking_xyz sequence as an example, (**a**,**c**) show the estimated versus true values of ORB-SLAM3 and DLD-SLAM on the trajectory as well as the error along the x, y, and z axis directions, and (**b**,**d**) show the ATE of DLD-SLAM with both the change in position and the change in time.

Table 3. The ATE of ORB-SLAM3 and DLD-SLAM on the TUM dataset (m). RMSE is the root mean square error, Mean is the mean, Median is the median, and S.D. is the standard deviation.

Sequence	ORB-SLAM3				DLD-SLAM				Promotion Rate of RMSE (%)
	RMSE	Mean	Median	S.D.	RMSE	Mean	Median	S.D.	
fr3-w-xyz	0.6847	0.6097	0.6306	0.3116	**0.0185**	0.0163	0.0147	0.0088	**97.29**
fr3-w-rpy	0.8003	0.6846	0.6584	0.4145	**0.0424**	0.0307	0.0229	0.0293	**94.71**
fr3-w-halfsphere	0.7057	0.6481	0.6041	0.2792	**0.0219**	0.0186	0.0156	0.0118	**96.89**
fr3-w-halfsphere	0.4028	0.368	0.3017	0.1638	**0.0056**	0.0049	0.0043	0.0028	**98.61**

The results of method in this paper are bolded.

Table 4 presents the absolute trajectory error RMSE of ORB-SLAM3, DS-SLAM, Detect-SLAM, DynaSLAM, and DLD-SLAM. DS-SLAM and Detect-SLAM are similar to DLD-SLAM in that they all use different networks for the semantic segmentation of the dynamic objects and then determine the real dynamic targets for dynamic feature point rejection.

Their positioning accuracy is not much different from this paper's method, and the ATE of DLD-SLAM can be reduced by about 30% at most. In terms of positioning accuracy, DynaSLAM's exceeds DLD-SLAM's. DynaSLAM achieves higher accuracy by leveraging the MASK-RCNN network for dynamic target segmentation and incorporating geometric constraints. Nevertheless, the real-time performance of the MASK-RCNN network is limited by its extended duration.

Table 4. The absolute trajectory RMSE of each traditional algorithm (unit: m).

Sequence	ORB-SLAM3	DS-SLAM	Detect-SLAM	DynaSLAM	DLD-SLAM
fr3-w-xyz	0.6847	0.0257	0.0254	0.0156	**0.0185**
fr3-w-rpy	0.8003	0.4453	0.4559	0.0358	**0.0424**
fr3-w-halfsphere	0.7057	0.0346	0.2021	0.0179	**0.0219**
fr3-w-static	0.4028	0.0072	0.0069	0.0011	**0.0056**

The results of method in this paper are bolded.

The method presented in this paper demonstrates enhanced capabilities in addressing localisation challenges in the presence of dynamic interference. The results indicate a notable enhancement in both the accuracy and robustness of the localisation. Moreover, it improves the localisation accuracy of real-time algorithms compared to classical dynamic visual SLAM. Although the localisation accuracy is slightly lower than that of DynaSLAM, DLD-SLAM makes up for the shortcomings of DynaSLAM in operational efficiency.

4.3.2. The Efficiency of the Algorithm

Figure 13 depicts a box plot of the duration taken by this algorithm to process each frame of the fr3_walking_xyz sequence. After removing abnormal values, the time taken to process each frame is between 62 and 68 ms, with a mean value of 65.8188 ms.

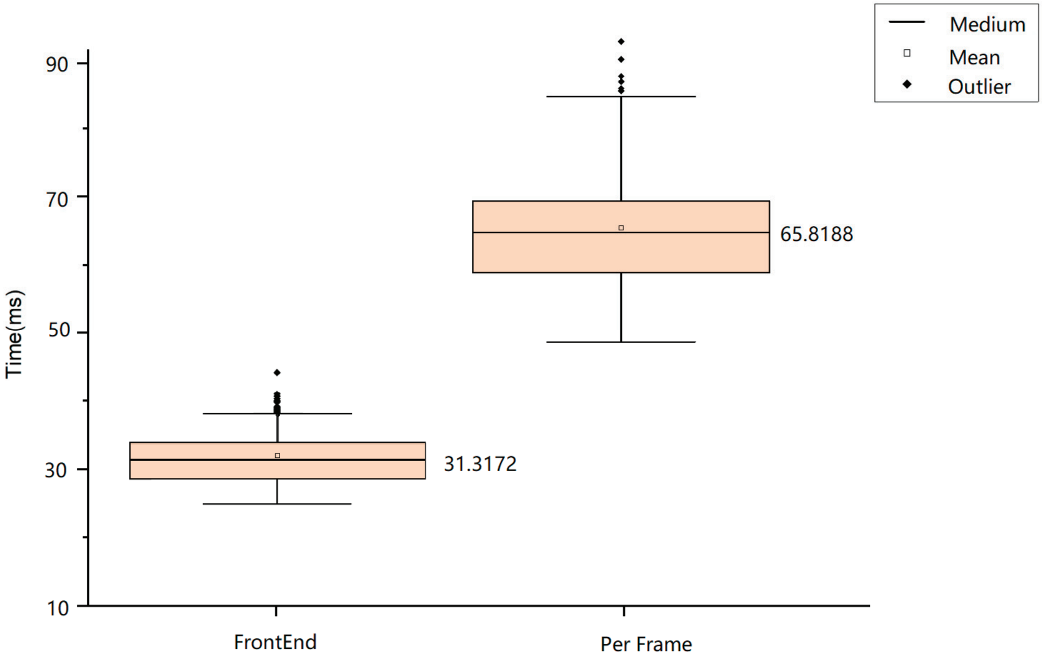


Figure 13. The average time consumption to process the front end per frame and process each frame of the fr3_walking_xyz sequence.

Table 5 provides a comparative analysis of the running efficiency of DLD-SLAM and various traditional algorithms in the processing of the fr3-walking-xyz sequence. The time required by each algorithm to process each frame is recorded. One of the considered methods, ORB-SLAM3, does not address the localisation challenge in dynamic environments and includes processes for detecting and rejecting dynamic feature points; consequently, the running time is reduced. DS-SLAM and Detect-SLAM are both examples of dynamic feature point rejection methods, which are employed in the research paper being discussed. However, the method in this paper replaces traditional ORB feature point extraction and matching with a more efficient and stable deep learning method in the front-end session. The integration of a lightweight network and RGB-D depth information is used in dynamic feature point detection and rejection, resulting in enhanced operational efficiency and real-time performance. DynaSLAM cannot run in real-time due to the use of the MASK-RCNN network for segmentation. Therefore, DLD-SLAM demonstrates great superiority in running efficiency in the dynamic visual SLAM algorithm.

Table 5. The comparison of average time of fr3_walking_xyz sequence processed by various traditional algorithms.

Method	Time Consumption (Unit: ms)	
	Front End	Per Frame
ORB-SLAM3	-	46.81
DynaSLAM	310.87	376.36
DS-SLAM	42.61	78.46
Detect-SLAM	57.74	96.14
DLD-SLAM	**31.32**	**65.82**

The results of method in this paper are bolded.

5. Discussion

The performance of localisation. The results presented in Figure 6 and Table 1 demonstrate that the DLD-SLAM system successfully extracts a sufficient number of feature points during the feature extraction and matching process. Furthermore, these feature points are matched with greater resilience compared to the conventional ORB features. Figures 7 and 8 demonstrate that our improved target detection method, GS-YOLOv5s, can be effectively trained and predicted. The target detection results depicted in Figure 9 demonstrate that they satisfy the algorithm's requirements and effectively identify the area where dynamic feature points with semantic labels are situated, when combined with the depth image. Figures 11 and 12, and Table 3 embody the accuracy comparison between DLD-SLAM and ORB-SLAM3 in processing the dynamic sequences of the TUM dataset. It can be seen that the method of this paper can effectively solve the localisation problem of the system in a dynamic environment. By integrating target detection and depth image, the approach successfully acquires object mask and semantic information. Additionally, the adoption of a dynamic feature point rejection strategy further enhances the method's effectiveness. This method has also proved to be feasible. Table 4 presents a comparative analysis of the positioning accuracy achieved by DLD-SLAM and traditional dynamic SLAM methods; DS-SLAM and Detect-SLAM are the same type of dynamic visual SLAM that can be operated in real-time as this paper, and the accuracy of DLD-SLAM is more superior. This paper presents several contributions that enhance the accuracy of feature point extraction and matching methods. Additionally, it introduces a dynamic feature point rejection method and strategy. The accuracy performance of DynaSLAM surpasses that of other systems; however, its network model lacks the capability to operate in real time. Table 4 illustrates the method's superior performance in terms of localisation.

The time consumption. Table 2 presents the observed decrease in network parameters of GS-YOLOv5s in comparison to YOLOv5s, along with the corresponding enhancement in running speed. This demonstrates the efficacy of our approach in relation to its compact model size and rapid processing capabilities. Figure 13 demonstrates the running time of DLD-SLAM, including the time consumed per front end and the time consumed per frame. Table 5 embodies the running efficiency comparison between DLD-SLAM and other classical algorithms. The running times per frame of DS-SLAM, Detect-SLAM, and DLD-SLAM are similar in removing the front end time. They differ mainly in the front end time consumed per frame. This reflects our contribution to facilitating the processing of the YOLOv5s algorithm and obtaining semantic segmentation masks by combining them with depth images to improve the running efficiency of the algorithm. DynaSLAM requires more time and does not reach an adequate real-time running performance. In comparison, it is proved that the method in this paper effectively improves the real-time operation of the algorithm.

6. Conclusions

In this paper, a deep learning-based RGB-D visual SLAM algorithm is proposed, which can be applied to dynamic environments and has improved running efficiency. The GCNv2-tiny deep learning method is employed in the tracking thread of ORB-SLAM3, replacing the conventional pyramid-based ORB feature point extraction and matching technique. This method can not only extract enough and uniformly distributed feature points but also has better performance in terms of running efficiency as well as robustness. And the semantic segmentation thread for dynamic targets is added to the original three threads. In this thread, the YOLOv5s target detection network is enhanced through the incorporation of GSConv convolution, which improves the CSP2_x module, resulting in a reduction in the parameters of the deep learning network. In order to meet the system detection accuracy at the same time and improve efficiency, the target detection algorithm needs to be more lightweight. The dynamic targets derived from the target detection process are subsequently integrated with the RGB-D depth data in order to acquire the semantic mask. This approach has the ability to minimise the excessive elimination of static feature points within the detection frame so as to avoid insufficient feature points for position estimation. After determining the mask of the potential dynamic target, this paper adopts the LK optical flow method to judge the relative motion of the feature points in the detection frame by comparing it with the velocity threshold. And the dynamic probability is determined by combining the last frame state of the feature point and the semantic labels. The identification of real dynamic objects is based on the potential dynamic targets. Within the mask region, the dynamic feature points are excluded to ensure that only static feature points are considered for position estimation and optimisation.

The verification of the method's feasibility and performance is conducted on various sequences from the TUM dataset. In this paper, the algorithm is also evaluated against the ORB-SLAM3, DS-SLAM, Detect-SLAM, and DynaSLAM algorithms in terms of both positioning accuracy and running efficiency. The research results demonstrate a substantial decrease in the absolute trajectory error within dynamic environments. The accuracy of the system increases by approximately 95% when compared to ORB-SLAM3. In comparison to real-time dynamic visual SLAM with its high accuracy, there has been a notable increase of 31.54% in terms of running efficiency. The comprehensive performance of our method demonstrates its superiority. The efficacy of DLD-SLAM in enhancing positioning accuracy, operational efficiency, and robustness in dynamic environments has been substantiated.

In the future, improvements will be made in the following aspects: Sensors such as IMU and LIDAR will be implemented. By broadening the scope of application, it is possible to enhance the accuracy and robustness of the system. The enhancement of running efficiency in deep learning networks can be accomplished by implementing lightweighting techniques such as model pruning. Moreover, it is reasonable to improve system performance by using faster and more accurate target detection methods, such as YOLOv7 and so on. Additionally, taking advantage of semantic information can facilitate the construction of a comprehensive semantic map, which can improve the representation of information in a more complete scene.

Author Contributions: Conceptualisation, Q.W.; methodology, H.Y.; software, H.Y.; validation, H.Y.; formal analysis, H.Y.; investigation, H.Y. and L.L.; resources, H.Y.; data curation, H.Y.; writing—original draft preparation, H.Y.; writing—review and editing, C.Y. and Y.S.; visualisation, H.Y.; supervision, C.Y., Y.F. and Q.W.; project administration, Q.W. and C.Y.; funding acquisition, Q.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 42074039). The authors would like to thank the referees or their constructive comments.

Data Availability Statement: Publicly available datasets were analysed in this study. These data can be found here: TUM dataset: https://cvg.cit.tum.de/data/datasets/rgbd-dataset (accessed on 1 August 2023); COCO dataset: https://cocodataset.org/#download (accessed on 1 August 2023); HPatches dataset: https://github.com/hpatches/hpatches-dataset (accessed on 1 August 2023); the EVO evaluation tool: https://github.com/MichaelGrupp/evo (accessed on 1 August 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Abaspur Kazerouni, I.; Fitzgerald, L.; Dooly, G.; Toal, D. A Survey of State-of-the-Art on Visual SLAM. *Expert Syst. Appl.* **2022**, *205*, 117734. [CrossRef]
2. Mur-Artal, R.; Tardós, J.D. ORB-SLAM2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [CrossRef]
3. Campos, C.; Elvira, R.; Rodríguez, J.J.G.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM. *IEEE Trans. Robot.* **2021**, *37*, 1874–1890. [CrossRef]
4. Qin, T.; Li, P.; Shen, S. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [CrossRef]
5. Forster, C.; Pizzoli, M.; Scaramuzza, D. SVO: Fast semi-direct monocular visual odometry. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014.
6. Chengqi, D.; Kaitao, Q.; Rong, X. Comparative Study of Deep Learning Based Features in SLAM. In Proceedings of the 2019 4th Asia-Pacific Conference on Intelligent Robot Systems (ACIRS), Nagoya, Japan, 13–15 July 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 250–254.
7. Mohamed, A.; Tharwat, M.; Magdy, M.; Abubakr, T.; Nasr, O.; Youssef, M. DeepFeat: Robust Large-Scale Multi-Features Outdoor Localization in LTE Networks Using Deep Learning. *IEEE Access* **2022**, *10*, 3400–3414. [CrossRef]
8. Yi, K.M.; Trulls, E.; Lepetit, V.; Fua, P. LIFT: Learned Invariant Feature Transform. In Proceedings of the Computer Vision—ECCV 2016, Amsterdam, The Netherlands, 11–14 October 2016; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 467–483.
9. Ballester, I.; Fontán, A.; Civera, J.; Strobl, K.H.; Triebel, R. DOT: Dynamic Object Tracking for Visual SLAM. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 11705–11711.
10. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. *arXiv* **2022**, arXiv:2206.02424. [CrossRef]
11. Xie, Y.; Tang, Y.; Tang, G.; Hoff, W. Learning To Find Good Correspondences Of Multiple Objects. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 2779–2786.
12. Dusmanu, M.; Rocco, I.; Pajdla, T.; Pollefeys, M.; Sivic, J.; Torii, A.; Sattler, T. D2-Net: A Trainable CNN for Joint Description and Detection of Local Features. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 8084–8093.
13. Revaud, J.; Weinzaepfel, P.; Souza, C.D.; Humenberger, M. R2D2: Repeatable and Reliable Detector and Descriptor. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Curran Associates Inc.: Red Hook, NY, USA, 2019; pp. 12414–12424.
14. DeTone, D.; Malisiewicz, T.; Rabinovich, A. SuperPoint: Self-Supervised Interest Point Detection and Description. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018. [CrossRef]
15. Tang, J.; Ericson, L.; Folkesson, J.; Jensfelt, P. GCNv2: Efficient Correspondence Prediction for Real-Time SLAM. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3505–3512. [CrossRef]
16. Clark, R.; Wang, S.; Wen, H.; Markham, A.; Trigoni, N. VINet: Visual-Inertial Odometry as a Sequence-to-Sequence Learning Problem. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31. [CrossRef]
17. Zhang, R.; Zhang, X. Geometric Constraint-Based and Improved YOLOv5 Semantic SLAM for Dynamic Scenes. *ISPRS Int. J. Geo-Inf.* **2023**, *12*, 211. [CrossRef]
18. Zhang, X.; Zhang, R.; Wang, X. Visual SLAM Mapping Based on YOLOv5 in Dynamic Scenes. *Appl. Sci.* **2022**, *12*, 11548. [CrossRef]

19. Kim, D.-H.; Kim, J.-H. Effective Background Model-Based RGB-D Dense Visual Odometry in a Dynamic Environment. *IEEE Trans. Robot.* **2016**, *32*, 1565–1573. [CrossRef]
20. Kerl, C.; Sturm, J.; Cremers, D. Dense visual SLAM for RGB-D cameras. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 2100–2106. [CrossRef]
21. Guohao, F.; Lele, B.; Cheng, Z. Geometric Constraint-Based Visual SLAM Under Dynamic Indoor Environment. *Comput. Eng. Appl.* **2021**, *57*, 203–212.
22. Zhang, C.; Zhang, R.; Jin, S.; Yi, X. PFD-SLAM: A New RGB-D SLAM for Dynamic Indoor Environments Based on Non-Prior Semantic Segmentation. *Remote Sens.* **2022**, *14*, 2445. [CrossRef]
23. Yu, C.; Liu, Z.; Liu, X.-J.; Xie, F.; Yang, Y.; Wei, Q.; Fei, Q. DS-SLAM: A Semantic Visual SLAM towards Dynamic Environments. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1168–1174.
24. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
25. Zhong, F.; Wang, S.; Zhang, Z.; Chen, C.; Wang, Y. Detect-SLAM: Making Object Detection and SLAM Mutually Beneficial. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 1001–1010. [CrossRef]
26. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the Computer Vision—ECCV 2016, Amsterdam, The Netherlands, 11–14 October 2016; Volume 9905, pp. 21–37.
27. Liu, Y.; Miura, J. RDS-SLAM: Real-Time Dynamic SLAM Using Semantic Segmentation Methods. *IEEE Access* **2021**, *9*, 23772–23785. [CrossRef]
28. Li, A.; Wang, J.; Xu, M. DP-SLAM: A visual SLAM with moving probability towards dynamic environments. *Inf. Sci.* **2020**, *556*, 128–142. [CrossRef]
29. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2980–2988. [CrossRef]
30. Bescos, B.; Fácil, J.M.; Civera, J.; Neira, J. DynaSLAM: Tracking, Mapping, and Inpainting in Dynamic Scenes. *IEEE Robot. Autom. Lett.* **2018**, *3*, 4076–4083. [CrossRef]
31. Wu, W.; Guo, L.; Gao, H.; You, Z.; Liu, Y.; Chen, Z. YOLO-SLAM: A semantic SLAM system towards dynamic environment with geometric constraint. *Neural Comput. Appl.* **2022**, *34*, 6011–6026. [CrossRef]
32. Wei, S.; Wang, S.; Li, H.; Liu, G.; Yang, T.; Liu, C. A Semantic Information-Based Optimized vSLAM in Indoor Dynamic Environments. *Appl. Sci.* **2023**, *13*, 8790. [CrossRef]
33. Wang, X.; Zhang, X. MCBM-SLAM: An Improved Mask-Region-Convolutional Neural Network-Based Simultaneous Localization and Mapping System for Dynamic Environments. *Electronics* **2023**, *12*, 3596. [CrossRef]
34. Xiao, J.; Owens, A.; Torralba, A. SUN3D: A Database of Big Spaces Reconstructed Using SfM and Object Labels. In Proceedings of the 2013 IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 1625–1632.
35. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
36. Wang, C.-Y.; Liao, H.-Y.M.; Wu, Y.-H.; Chen, P.-Y.; Hsieh, J.-W.; Yeh, I.-H. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 1571–1580. [CrossRef]
37. Lucas, B.D.; Kanade, T. An Iterative Image Registration Technique with an Application to Stereo Vision. In Proceedings of the 7th International Joint Conference on Artificial Intelligence—Volume 2, Vancouver, BC, Canada, 24–28 August 1981; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1981; pp. 674–679.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

Article

Characterization of BDS Multipath Effect Based on AT-Conv-LSTM Network

Jie Sun, Zuping Tang, Chuang Zhou and Jiaolong Wei *

School of Electronic Information and Communication, Huazhong University of Science and Technology, Wuhan 430074, China; jessiesun@hust.edu.cn (J.S.); tang_zuping@hust.edu.cn (Z.T.); zhouchuang@hust.edu.cn (C.Z.)
* Correspondence: jlwei@hust.edu.cn

Abstract: Multipath effects are the most challenging error sources for the Global Navigation Satellite System receiver, affecting observation quality and positioning accuracy. Due to the non-linear and time-varying nature, multipath error is difficult to process. Previous studies used a homogeneous indicator to characterize multipath effects and only revealed the temporal or spatial correlations of the multipath, resulting in limited correction performance. In this study, we consider the code multipath to be influenced not only by the elevation and azimuth angle of certain stations to satellites but also to be related to satellite characteristics such as nadir angle. Hence, azimuth angle, elevation angle, nadir angle and carrier-to-noise power density ratio are taken as multiple indicators to characterize the multipath significantly. Then, we propose an Attention-based Convolutional Long Short-Term Memory (AT-Conv-LSTM) that fully exploits the spatiotemporal correlations of multipath derived from multiple indicators. The main processing procedures using AT-Conv-LSTM are given. Finally, the AT-Conv-LSTM is applied to a station for 16 consecutive days to verify the multipath mitigation effectiveness. Compared with sidereal filtering, multipath hemispherical map (MHM) and trend-surface analysis-based MHM, the experimental results show that using AT-Conv-LSTM can decrease the root mean square error and mean absolute error values of the multipath error more than 60% and 13%, respectively. The proposed method can correct the code multipath to centimeter level, which is one order of magnitude lower than the uncorrected code multipath. Therefore, the proposed AT-Conv-LSTM network could be used as a powerful alternative tool to realize multipath reduction and will be of wide practical value in the fields of standard and high-precision positioning services.

Keywords: BDS; multipath; AT-Conv-LSTM; spatiotemporal domain

Citation: Sun, J.; Tang, Z.; Zhou, C.; Wei, J. Characterization of BDS Multipath Effect Based on AT-Conv-LSTM Network. *Remote Sens.* **2024**, *16*, 73. https://doi.org/10.3390/rs16010073

Academic Editors: Zhetao Zhang, Guorui Xiao, Zhixi Nie, Vagner Ferreira and Giuseppe Casula

Received: 2 November 2023
Revised: 18 December 2023
Accepted: 22 December 2023
Published: 24 December 2023

1. Introduction

As an essential technology, the Global Navigation Satellite System (GNSS) has been utilized in various fields, including aviation [1], geodesy [2] and earthquake detection [3] and climate monitoring [4]. However, in complex environments like urban canyons, GNSS stations face challenges due to the reflection, diffraction and obstruction of signals by local obstacles around the antenna [5]. Currently multipath does not have a commonly accepted method for its correction that introduces meter-scale code observation errors and centimeter-scale carrier phase observation errors [6]. Hence, it is crucial to develop a method for mitigating multipaths to enhance the accuracy of standard and high-precision positioning services.

For certain environments and receivers, there are two distinct multipath mitigation classifications: hardware enhancement and data handling. The former methods principally include antenna-based designs [7,8] and receiver-based architectures [9] to mitigate multipath errors. However, the enhancement has limited effectiveness and are difficult to implement. The data handling methods eliminated or mitigated multipath errors via code and phase observations combination [10,11], parameterization [12] or model correction.

The parameterization category chooses the carrier-to-noise power density ratio (C/N0) and signal-to-noise ratio (SNR) [13,14], parameterized into the stochastic model to characterize multipath effects.

One kind of empirical model correction is based on time-domain repeatability of multipath, such as sidereal filtering (SF) [15]. Genrich and Bock proposed the SF method and calculated the orbital repeat time (ORT) of GPS satellites to be 23 h, 56 min and 4 s to establish multipath correction model. Choi et al. regarded the ORT should be calculated for individual satellite [16], which revealed that the sidereal repeat period deviates from the widely accepted sidereal recurrence by roughly 9 s. The position-domain SF method has been introduced to attain multipath mitigation in precise point positioning (PPP) [17]. Ragheb et al. carried out position-domain and observation-domain multipath mitigation methods for GPS and applied them to the precise point position model [18]. Hung and Rau improved the multipath corrected efficiency via bandpass filters in the position domain [19]. In addition, Atkins and Ziebart evaluated and compared the effectiveness of observation domain SF with position domain SF for GPS PPP. Different from the position domain, these variations of orbital repetition time are slightly different for individual GPS satellites in the observation domain. And observation-domain multipath mitigation is more advantageous in the handling of high-frequency components [20]. However, the sidereal filtering requires the pre-computation of the orbital repetition time, thereby deteriorating the real-time performance and positioning accuracy of multipath mitigation.

Another approach used in the literatures is based on spatial-domain repeatability. Considering that the position of GNSS antenna and the environment around the GNSS stations remain unchanged, the multipath is solely in accordance with the particular elevation and azimuth angle of station-satellite couple. The emerging methods consist of multipath spherical harmonic model [21], multipath stacked (MPS) model [22] and multipath hemispherical diagram (MHM) [23]. The multipath correction value for the MPS and MHM approaches is the average of all satellite residuals in each grid. These approaches realize $1^\circ \times 1^\circ$ high-resolution equal lattice grids on the sky map, which are easier to produce than for the spherical harmonic model. Multipath hemispherical diagram based on trend-surface (T-MHM) is proposed to fit the multipath spatial distribution with trend-surface modeling within the grid [24]. Lu et al. discussed the optimal modeling days, applied T-MHM to BDS-3 PPP and evaluated the multipath correction effect simultaneously, and validated the mitigation effect of T-MHM on different grid scales [25,26]. Zhang et al. evaluated the multipath mitigation effect of MHM in BDS2/BDS3 real-time kinematic (RTK) [27]. Although this method has a low complexity, it disregards the multipath's spatial distribution within the grid. Thus, it is more effective in correcting low frequency multipath but has limited effectiveness in correcting the high-frequency multipath. It is clear that averaging residuals could somewhat filter out high-frequency signals in the grids. This drawback can be resolved by reducing the grid but introduces the new risk of rendering the model less robust.

The majority of research and studies on BDS-3 multipath mitigation have mainly been conducted on the different observation model, whereas few have explored BDS-3 multipath mitigation strategy on a undifferenced and uncombined PPP model. When existing methods are used to mitigate the BDS with hybrid constellation, the multipath model will be more complex than GPS systems. In current studies, SNR or C/N0 are used to evaluate the observations and identify the multipath, elevation and azimuth angle, which are other indicators used to the characterize multipath in the MHM method. However, the indicator to reflect the characteristics of multipath effects in current studies is comparatively homogeneous, the multiple joint indicators are critical to guaranteeing the best performance. In this study, we use the azimuth angle, elevation angle, nadir angle and C/N0 as multiple indicators to characterize the multipath significance in undifferenced and uncombined PPP modes.

The modeling and mitigation of the multipath pose significant challenges due to its complex nonlinear and time-varying nature. In recent years, deep learning has emerged as a powerful technique for addressing non-linear problems and has been successfully employed in various domains, such as ionosphere forecasting [28,29], troposphere tomography [30], satellite orbit broadcast [31], satellite clock prediction [32], self-driving [33] and integrated navigation [34]. Deep learning algorithms such as neural networks are data-driven models that use large and extensive datasets to obtain correlations without relying on complex physically based models [35]. Moreover, multiple indicators make it more difficult to establish the multipath mathematical model. The attention mechanism is able to focus on the important information from the massive datasets and ignore mostly dispensable information [36,37]. The attention mechanism can help the network in assigning weights of the TEC time series to forecast ionospheric TEC [38]. The attention-based Conv-LSTM network is proposed, CNN and LSTM modules are applied to obtain the spatial feature and the temporal feature, respectively, while the attention mechanism could emphasize importance level in the dataset [39]. Recently, few researchers have utilized artificial intelligence in the multipath mitigation field [35]. But existing methods only reveal the temporal or spatial correlations of multipath, which fail to fully grasp the elaborate characteristics of individual time series and has limited enhancement in mitigation performance. Therefore, we propose a multipath mitigation using the Attention-based Convolutional Long Short-Term Memory (AT-Conv-LSTM) to maximize the spatiotemporal repeatability of multipath with multiple indicators.

In this study, the potential of utilizing AT-Conv-LSTM for mitigating multipath effects is elaborated. Firstly, we derive the multipath extracting method on the undifferenced and uncombined PPP model and analyze multiple indicators, including elevation, azimuth, nadir angle and C/N0 to characterize the multipath spatiotemporal correlation. Secondly, we propose the multipath mitigation method based on AT-Conv-LSTM network, and the main pre-processing steps are outlined. Then, the comparative experiment is used to analyze the multipath discrepancy of SF, MHM, T-MHM and AT-Conv-LSTM. Finally, the study concludes with a summary and suggestions for future research.

2. Multipath Analysis Method

2.1. Multipath Extraction

The code and carrier observations of the GNSS signal received at the station are:

$$p_{r,i}^{s,Q} = \mathbf{u}_r^{s,Q}\cdot\mathbf{R}_r^s + dt_r^s - dt^{s,Q} + M_{\mathrm{w}}\cdot Z_{\mathrm{w}} + \gamma_i^Q \cdot I_1^{s,Q} + b_{r,i}^s - b_i^{s,Q} + m_{r,i}^{s,Q} + \varepsilon_{r,i}^{s,Q}, \tag{1}$$

$$l_{r,i}^{s,Q} = \mathbf{u}_r^{s,Q}\cdot\mathbf{R}_r^s + dt_r^s - dt^{s,Q} + M_{\mathrm{w}}\cdot Z_{\mathrm{w}} - \gamma_i^Q \cdot I_1^{s,Q} + \lambda_i^s \cdot N_i^{s,Q} + B_{r,i}^s - B_i^{s,Q} + M_{r,i}^{s,Q} + \xi_{r,i}^{s,Q}, \tag{2}$$

where $p_{r,i}^{s,Q}$ and $l_{r,i}^{s,Q}$ indicate the values of "observed minus computed (OMC)" for code and carrier phase observables, individually; s represents the PRN number, Q represent the satellite system, r expresses the receiver ID, i expresses the frequency band number; $\mathbf{u}_r^{s,Q}$ denotes the line-of-sight (LOS) unit vector; $\mathbf{R}_r^s$ is the vector of receiver position increments relative to the a priori position; dt_r^s states the receiver clock offsets, $dt^{s,Q}$ states the satellite clock offsets; Z_{w} means the zenith wet delay; M_{w} means the wet mapping function; $I_1^{s,Q}$ implies the ionospheric delay on the first frequency band, $\gamma_i^Q = \left(\frac{f_1^{s,Q}}{f_{iW}^{s,Q}}\right)^2$ is the multiplier factor introduced to convert to frequency f_i; $b_{r,i}^s$ indicates the receiver uncalibrated code delay (UCDs) and $b_i^{s,Q}$ indicates the satellite UCDs corresponding to frequency f_i; λ_i^s denotes the carrier wavelength on ith frequency band; $N_i^{s,Q}$ denotes the integer phase ambiguity; $B_{r,i}^s$ and $B_i^{s,Q}$ are the receiver and satellite uncalibrated phase delays (UPDs), which is different on each frequency band; the code and phase multipath error can be expressed as $m_{r,i}^{s,Q}$ and $M_{r,i}^{s,Q}$; and $\varepsilon_{r,i}^{s,Q}$ and $\xi_{r,i}^{s,Q}$ represent the code and carrier measurement noise. The receiver and satellite antenna phase center offsets (PCOs) and variations (PCVs), phase windup, tidal loadings, LOS hydrostatic delay, relativistic effects and Sagnac effects should be corrected via empirical models.

For convenience, the coefficient for the ionosphere-free (IF) combination are defined as:

$$\begin{aligned} \alpha_{mn}^{Q} &= \frac{\left(f_m^{s,Q}\right)^2}{\left(f_m^{s,Q}\right)^2 - \left(f_n^{s,Q}\right)^2} \\ \beta_{mn}^{Q} &= -\frac{\left(f_n^{s,Q}\right)^2}{\left(f_m^{s,Q}\right)^2 - \left(f_n^{s,Q}\right)^2} \end{aligned} \tag{3}$$

where $f^{s,Q}$ stands for the frequency band $(m,\ n = 1,2; m \neq n)$ and α_{nm}^{Q} and β_{nm}^{Q} are the IF combination coefficients.

When linearly coupled with the satellite UCD, the satellite clock offset cannot be separated independently unless excess baseline constraints are incorporated. Currently, the IGS precise satellite clock products are estimated by utilizing the IF combination observables. Consequently, the satellite clock offsets containing the IF combination of satellite UCDs are:

$$\begin{aligned} \mathrm{dt}^{s}_{\mathrm{IF}_{12}^{s,Q}} &= dt^{s,Q} + \left(\alpha_{12}^{Q} \cdot b_1^{s,T} + \beta_{12}^{Q} \cdot b_2^{s,Q}\right) \\ &= dt^{s,Q} + b_{\mathrm{IF}_{12}}^{s,Q}, \end{aligned} \tag{4}$$

with

$$\begin{cases} DCB_{P_mP_n}^{s,Q} = b_m^{s,Q} - b_n^{s,Q}, \mathrm{DC}B_{r,P_mP_n}^{s,Q} = b_{r,m}^{s,Q} - b_{r,n}^{s,Q} \\ b_{IF_{mn}}^{s,Q} = \alpha_{mn}^{Q} \cdot b_m^{s,Q} + \beta_{mn}^{Q} \cdot b_n^{s,Q}, b_{r,IF_{mn}}^{s,Q} = \alpha_{mn}^{Q} \cdot b_{r,m}^{s,Q} + \beta_{mn}^{Q} \cdot b_{r,n}^{s,Q} \end{cases}, \tag{5}$$

where $DCB_{P_mP_n}^{s,Q}$ and $\mathrm{DC}B_{r,P_mP_n}^{s,Q}$ express satellite and receiver DCB between pseudoranges $p_{r,m}^{s,Q}$ and $p_{r,n}^{s,Q}$ distinctly.

In the dual-frequency undifferenced and uncombined PPP model, it is assumed that j satellites are simultaneously tracked by the receiver r. Equations (1) and (2) can be rewritten as:

$$\begin{bmatrix} p_{r,1}^{1,Q} \\ l_{r,1}^{1,Q} \\ \vdots \\ p_{r,2}^{j,Q} \\ l_{r,2}^{j,Q} \end{bmatrix} = \begin{bmatrix} -\mathbf{u}_r^{s,Q} & 1 & \boldsymbol{M}_{\mathrm{w}} & \mathbf{K} & \boldsymbol{R}_1 & \boldsymbol{R}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathrm{d}\bar{t}_r^{Q} \\ Z_{\mathrm{w}} \\ \bar{\mathbf{I}}_1^{Q} \\ \overline{\mathbf{N}}_1^{Q} \\ \overline{\mathbf{N}}_2^{Q} \end{bmatrix} + \begin{bmatrix} m_{r,1}^{Q} + \varepsilon_{r,1}^{Q} \\ M_{r,1}^{Q} + \xi_{r,1}^{Q} \end{bmatrix}, \tag{6}$$

with

$$\begin{cases} \mathrm{d}\bar{t}_r^{Q} = \mathrm{d}t_r + d_{r,\mathrm{IF}_{12}}^{Q} \\ \bar{\mathbf{I}}_1^{s,Q} = I_{r,1}^{s,Q} + \beta_{12}^{Q} \cdot \left(\mathrm{DCB}_{r,P_1P_2}^{Q} - \mathrm{DCB}_{P_1P_2}^{s,Q}\right) \\ \overline{\mathbf{N}}_1^{s,Q} = \lambda_1^{Q} \cdot \left(N_{r,1}^{s,Q} + b_{r,1}^{s,Q} - b_1^{s,Q}\right) + b_{\mathrm{IF}_{12}}^{s,Q} - b_{r,\mathrm{IF}_{12}}^{s,Q} + \beta_{12}^{Q} \cdot \left(\mathrm{DCB}_{r,P_1P_2}^{Q} - \mathrm{DCB}_{P_1P_2}^{s,Q}\right) \\ \\ \overline{\mathbf{N}}_2^{s,Q} = \lambda_2^{Q} \cdot \left(N_{r,2}^{s,Q} + b_{r,2}^{s,Q} - b_2^{s,Q}\right) + b_{\mathrm{IF}_{12}}^{s,Q} - b_{r,\mathrm{IF}_{12}}^{s,Q} + \gamma_2^{Q} \cdot \beta_{12}^{Q} \cdot \left(\mathrm{DCB}_{r,P_1P_2}^{Q} - \mathrm{DCB}_{P_1P_2}^{s,Q}\right) \\ \end{cases}, \tag{7}$$

where 1 expresses a unit column vector of receiver clock offsets $\mathrm{d}\bar{t}_r^{Q}$, which has $2 \times$ j rows; **K** represents the unit column vector of ionospheric parameter $\bar{\mathbf{I}}_1^{Q}$, which the ingredient assigned to $p_{r,1}^{1,Q}$ is 1 and the factor assigned to $l_{r,1}^{1,Q}$ is -1; $\boldsymbol{R}_1$ indicates the matrix of ambiguity parameters $\overline{\mathbf{N}}_1^{s,Q}$, the factor assigned to $p_{r,1}^{s,Q}$ is 0 and the factor assigned to $l_{r,1}^{s,Q}$ is 1; $\boldsymbol{R}_2$ indicates the matrix of the ambiguity parameters $\overline{\mathbf{N}}_2^{s,Q}$, the factor assigned to $p_{r,2}^{s,Q}$ is 0 and the factor assigned to $l_{r,2}^{s,Q}$ is 1.

In this study, we only discuss the code multipath, and the comprehensive data handling strategies are stated in Table 1. The parameters $\overline{\mathbf{X}} = \left[\mathrm{d}\bar{t}_r^Q, \overline{Z}_\mathrm{w}, \overline{\mathbf{I}}_1^{s,Q}, \overline{\mathbf{N}}_1^{s,Q}, \overline{\mathbf{N}}_2^{s,Q}\right]$ that are estimated with a Kalman filter are substituted into (1) to extract the code multipath as follows:

$$\mathrm{E}\left[m_{r,i}^{s,Q}\right] = P_{r,i}^{s,Q} - \mathbf{u}_r^{s,Q} \cdot \mathbf{R}_r^s - \mathrm{d}\bar{t}_r^Q - M_\mathrm{w} \cdot \overline{Z}_\mathrm{w} - \gamma_i^Q \cdot I_1^{s,Q}, \tag{8}$$

where E[*] is defined as extraction operator and $\mathbf{R}_r^s$ is acquired from previous receiver position. It can be seen that after eliminating other modeled errors, the remaining code residuals only contain multipath errors and random noise.

Table 1. Data processing strategies.

Items	Strategies
Observations	BDS: B1/B3
Sampling rate	30 s
Elevation cutoff	7°
Parameter estimator	Kalman filter
Satellite orbits and clocks	WHU MGEX precise orbit (5 min interval) and clock (30 s interval) products
Carrier phase windup	Corrected using the external model
Tidal load	Corrected using the IERS convention model
Relativity effects, Earth rotation	Corrected using the external model
Satellite and receiver antenna Phase center	Corrected with igs14.atx
Slant ionospheric delays	Estimated as random-walk noise parameters ($0.01\mathrm{m}^2/\mathrm{s}$)
Tropospheric delays	The mapping function utilized for line of sight direction is global mapping function, zenith hydrostatic delays are corrected using the Saastamoinen model, zenith wet delays are estimated as random-walk noises (10^{-7} m^2/s)
Receiver clocks	Estimated as white noises
Phase ambiguities	Estimated as float constants
Station coordinates	Estimated as day constants
Stochastic model	Elevation-dependent weighting (prior variance as 0.003 and 0.3 m for code and phase observations)

2.2. Multipath Analysis Method Based on Different Indicators

Multipath means that the satellite signal arrives at the antenna through multiple paths after being blocked, reflected and refracted by obstacles in the environment surrounding the receiver. The amplitude and phase of the multipath signal depend on the receiver's position as well as the environment. Therefore, existing studies generally argue that the code multipath can be modeled as a function of the corresponding satellite azimuth and elevation, unless there are changes in the nearby environment. However, even if the environment near the antenna remains unchanged in application, multipath errors exhibit characteristics related to the variations of satellite position. It is indispensable to explore the correlation between nadir angle with code multipath.

Ruan conducted the detailed modeling of satellite-induced multipath and proposes that the satellite-induced multipath should be established as functions that are relative to the nadir angle instead of the elevation angle [40]. In this study, we take the nadir angle into consideration. The satellite-induced variation of the multipath error can be well characterized by the nadir angle. This model will be proved next.

As shown in Figure 1, a triangle is formed by satellite S, station A and geocenter O, the earth is approximated as a sphere, and according to the Law of Sines:

$$\frac{sin(e_A + 90^\circ)}{R_e + H_s} = \frac{sin(e_z)}{R_e + H_a} \tag{9}$$

where indicates e_A and H_a are the elevation angle and altitude of station A, respectively; e_z represents the nadir angle of station A; H_a represents the altitude of station A, R_e indicates the earth's radius and H_s represents the altitude of satellite S.

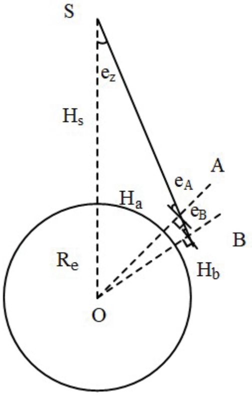


Figure 1. Geometrical relationship of satellite, earth and station.

For another station B on the LOS vector AS, the similar mathematic relationship can be formed as follows:

$$\frac{sin(e_B + 90^\circ)}{R_e + H_s} = \frac{sin(e_z)}{R_e + H_b} \tag{10}$$

where indicates e_B and H_b are the elevation and altitude of station B, respectively.

Substituting (10) into (9), the equation can be expressed as:

$$\frac{sin(e_A + 90^\circ)}{R_e + H_a} = \frac{sin(e_B + 90^\circ)}{R_e + H_b} \tag{11}$$

It is clear that stations at different altitudes observe the same satellite at different elevation angles. We introduce the nadir angle as an independent variable to jointly model the satellite-induced multipath with receivers at different altitudes.

As mentioned before, the main parameters for each GNSS satellite include C/N0 to reflect the characteristics of multipath effects [14]. Different from SNR, C/N0 is irrelevant to the receiver's front-end bandwidth and represents the carrier power-to-noise power ratio that normalized to the unit bandwidth. The multipath signal is a composite signal formed by the reflection, refraction and diffraction components induced by obstacles in the surroundings. The multipath effect can modify the signal's amplitude and phase, causing the distortion of the original signal. Generally, the composite multipath signals can be expressed as:

$$S_m = A_d cos\varphi + A_i \cos(\varphi + \Delta\varphi), \tag{12}$$

where A_d and A_i indicate the amplitudes of direct and indirect signals, individually; φ represents the direct signal phase; and $\Delta\varphi$ represents the phase shift delayed by the indirect signal.

The C/N0 can reflect the composite signal's amplitude formed by superposing several multipath components in accordance with each phase. According to (12), the relationship is expressed as:

$$\mathrm{C/N0}^2 = A_d^2 + A_i^2 + 2A_d A_i cos\Delta\varphi, \tag{13}$$

It is obvious that C/N0 represents the quality of received signal, and the multipath signal has a notable influence on C/N0. Therefore, it is reasonable to characterize the multipath signals with C/N0.

In this study, the azimuth angle, elevation angle, nadir angle and C/N0 are taken as multiple joint indicators to characterize the multipath.

3. Multipath Characterization with the AT-Conv-LSTM Network

3.1. AT-Conv-LSTM Network

In this study, we propose a multipath error modeling method based on AT-Conv-LSTM network. The algorithm flowchart is shown in Figure 2: (1) Raw BDS-3 observations collected from GNSS stations are stored in the database, and then undergo a post-processing procedure to proceed multipath extraction operations to be used as training data. (2) A sliding window size is set, and the AT-Conv-LSTM network is updated by incorporating the earlier model and incoming multipath errors, and the multipath errors are estimated in the current epoch. (3) The code multipath is corrected at the corresponding epoch based on the AT-Conv-LSTM network to mitigate the multipath.

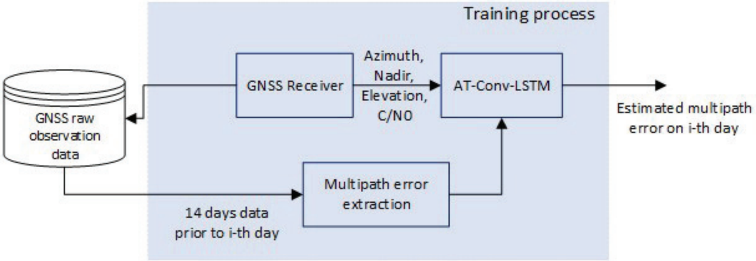

Training process
GNSS raw observation data
GNSS Receiver
Azimuth, Nadir, Elevation, C/NO
AT-Conv-LSTM
Estimated multipath error on i-th day
Multipath error extraction
14 days data prior to i-th day

Figure 2. Multipath mitigation flowchart.

Since multipath errors are commonly considered to be random and nonlinear, it is imperative for a robust multipath model to possess the capability to accurately represent these intricate characteristics. Multipath errors are usually characterized via spatiotemporal correlation and periodicity. More specifically, the point of interest (POI) region of the multipath error is not only related to the multipath errors of its neighboring observation epoch but also its dependence on previous time. Moreover, the multipath errors also exhibit periodic repetitive patterns. In this paper, the AT-Conv-LSTM network is proposed to estimate the multipath errors. The proposed model comprises two Conv-LSTM modules integrated with attention mechanism, which can effectively utilize the spatiotemporal correlation and mitigate the multipath errors of BDS-3, as depicted in Figure 3.

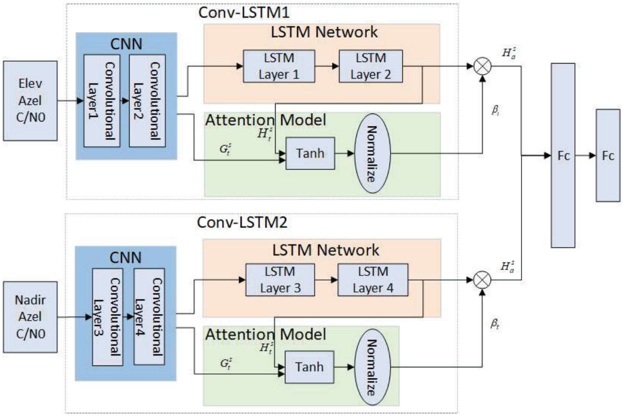


Figure 3. AT-Conv-LSTM network.

As mentioned earlier, the multipath is associated with the station environment, satellite position and receiver position. The multiple indicators that azimuth angle, elevation angle, nadir angle and C/N0 are taken to characterize the multipath significance. The multipath error can be denoted at *i*th satellite vary with the multiple indicators as:

$$m_{r,i}^{s,Q} = f(az_i, elev_i, nadir_i, f_i, CNR_i), \tag{14}$$

where $az_i, elev_i$ and f_i are the azimuth angle, elevation angle and frequency; $nadir_i$ represents the nadir angle; and CNR_i represents the C/N0 of *i*th satellite.

Hence the inputs of the first Conv-LSTM module are elevation angles and azimuth angles represent the variation related to receiver's surroundings and position, and the inputs of the second Conv-LSTM module are nadir angles and azimuth angles represent the variation related to satellite's position.

The core component of the proposed network is the Conv-LSTM module, encompassing two convolutional layers and two LSTM layers. The C/N0, nadir angle and azimuth angle of the *m*th satellite on the epoch *n* can be represented as $X_n^m = [az_n^m, nadir_n^m, CNR_n^m]$. Subsequently, we aggregate the historical angles from its neighboring locations (total n epochs) as follows:

$$X_t^s = \begin{bmatrix} X_1^s \\ X_2^s \\ \vdots \\ X_n^s \end{bmatrix} = \begin{bmatrix} X_1^1 & X_2^1 & \cdots & X_n^1 \\ X_1^2 & X_2^2 & \cdots & X_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ X_1^m & X_2^m & \cdots & X_n^m \end{bmatrix}, \tag{15}$$

where s $= 1, 2, \ldots, m$ symbolizes the count of satellites and $t = 1, 2, \ldots, n$ symbolizes the count of epochs.

The matrix X_t^s at each epoch *t* is subjected to a one-dimensional convolution process in order to capture the spatial feature. A sliding filter is used to capture the local perceptual domain using a one-dimensional convolution kernel filter. The following illustration shows how the convolution kernel filter works:

$$Y_t^s = \sigma(w_s * X_t^s + b_s), \tag{16}$$

where w_s represents the filter weights, b_s represents the bias, symbol * defined as the convolution operation, σ represents the activation function and Y_t^s expresses the convolutional layer output. The network uses the tanh activation function. The aforementioned process promotes the extraction of the spatial feature from the neighboring observation regions.

To enhance optimize the efficiency of the deep neural network, the most common approach involves augmenting the model through the addition of layers. By incorporating multiple LSTM layers into the network, this study enhances the network's capacity to adapt the multipath errors. After being extracted through two convolutional layers, the spatial features are inputted into the stacked LSTM network. Through the stacking of LSTM layers, each subsequent layer in the stack receives the hidden state served as the input of preceding layer. The LSTM's mathematical equation with different cell states is given below:

$$\begin{array}{c} i_t = \sigma\left(w_i\left[h_{t-1}^s, X_t^s\right] + b_i\right) \\ f_t = \sigma\left(w_f\left[h_{t-1}^s, X_t^s\right] + b_f\right) \\ \widetilde{C} = \tanh\left(w_c\left[h_{t-1}^s, X_t^s\right] + b_c\right) \\ C_t = f_t \circ C_{t-1} + i_t \circ \widetilde{C} \\ o_t = \sigma\left(w_o\left[h_{t-1}^s, X_t^s\right] + b_o\right) \\ H_t^s = o_t \circ \tanh(C_t) \end{array}, \tag{17}$$

where i_t represents the input of the LSTM layer on epochs t, i_t, f_t, o_t, which indicate the input gate, the forget gate and output gate at epoch *t*, respectively; $\circ$ expresses the Hadamard product; and w and b represent the weights and biases of the network, respectively. Finally, we obtain the spatiotemporal feature H_t^s for time step *t*.

3.2. Attention Mechanism Considering Multiple Indicators

The introduction of the attention mechanism aims to investigate the intrinsic features of the sequence and enhance the effectiveness of information handling. It enables models to assign different weights to different positions within the input sequence, enabling them to concentrate on the most significant components while processing each sequence element.

After two convolution layers, the spatial feature has been extracted from the input matrix X_t^s, including the C/N0, nadir and azimuth angles. G_t^s is denoted the convolutional layer 2 output. As is widely recognized, the multipath also demonstrates temporal correlations in adjacent epochs. LSTM is usually employed to uncover hidden temporal features in a time series. Therefore, after the processing of spatial information through the two convolutional layers, the output is subsequently linked to the LSTM network. Thus, we obtain the spatiotemporal feature H_t^s.

The observation data are not continuous in the time domain due to the limited visibility time of MEO and GEO satellites for a specific observation station. In order to address this issue, we introduce an attention mechanism to calculate the important score of each Conv-LSTM output. Through the attention mechanism, we can obtain the estimated multipath that automatically assigns a different importance score to each visible epoch.

The Conv-LSTM output at epoch t is calculated by combining the outputs of the CNN and LSTM module with weighting coefficients, as demonstrated below:

$$\begin{array}{c} s_t = v_s^T tanh(w_h G_t^s + w_l H_t^s) \\ \beta_k = \frac{exp(s_k)}{\sum_{k=1}^{n+1} exp(s_k)} \\ H_t^a = \sum_{k=1}^{n+1} \beta_k H_{t-(k-1)}^s, \end{array}, \tag{18}$$

where s_t, β_k, and H_t^a are, respectively, the importance score of each input part, the attention value and the output at epoch t. w and v are the weights. Ultimately, the spatiotemporal feature H_t^a for time step t is obtained.

The same structure as the other Conv-LSTM will not be explained here. Subsequently, all these features are combined into a feature vector, which is then fed into two regression layers for estimating purposes. The objective function of regression involves a loss function that calculates the mean squared error of the estimated multipath errors.

3.3. Model Training and Evaluation

Within the proposed model, the optimization of model parameters is achieved by employing the Adam optimization algorithm, which enables adaptive adjustment with the learning rate.

In order to provide a quantitative evaluation of the estimated accuracy of, the root mean square error (RMSE) and mean absolute error (MAE) are selected to calculate the difference between the real multipath error extracted in Section 2.1 and the estimated value. The following is the mathematical equation of RMSE and MAE:

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(m_t - \hat{m}_t)^2}, \tag{19}$$

$$MAE = \frac{1}{n}\sum_{t=1}^{n}|m_t - \hat{m}_t|, \tag{20}$$

where n represents sampling points number, m_t represents the extracted multipath in previous section and $\hat{m}_t$ is estimated value of multipath for ith sampling point. Moreover, as the RMSE and MAE values approach zero, the network's estimated results become closer to the actual multipath error, indicating better performance.

4. Results

4.1. Data Description

The BDS global PNT services consists of the experiment system (BDS-1), the regional system (BDS-2) and the global system (BDS-3) and reached full operational status on July 31, 2020 [41]. The BDS offers high-precision, real-time services to users worldwide. They are made up of a constellation of Medium Earth Orbit (MEO) (C19–C30, C32–C37, C41–46), Inclined Geosynchronous Orbit (IGSO) (C31, C38–C40) and Geosynchronous Earth Orbit (GEO) (C59, C60) satellites, which is different from GPS, GLONASS and Galileo. In addition to the antiquated B1l and B3l broadcasts that were carried over from the BDS-2 satellites, the BDS-3 satellites broadcast a variety of new open service signals. The frequencies, wavelengths and chip rates of BDS-3 signals are stated in Table 2. In our study, we analyzed the BDS dual-frequency observations (B1/B3) from the International GNSS Service (IGS) MGEX station for 16 days from DOY 160-176, 2021. The multi-frequency GNSS station, namely JFNG, is capable of tracking BDS, as shown in Table 3. The multi-frequency GNSS station, namely JFNG, is capable of tracking BDS as shown in Figure 4. And the detail station information is stated in Table 3.

Table 2. BDS-3 signal characteristics.

BDS-3 System	B1I	B1C	B2a	B2b	B3I
Frequency (MHz)	1561.098	1575.420	1176.450	1207.140	1268.520
Chip Rate (Mcps)	2.046	1.023	10.23	10.23	10.23
Wavelength (cm)	19.20	19.03	25.48	24.83	23.63

Table 3. Station information.

Items	Strategies
Station Name	JFNG
Localization	China
Latitude	30.51557°
Longitude	114.49102°
Receiver	TRIMBLE ALLOY—6.20
Antenna Type	TRM59800.00
Constellations	GPS + GLO + GAL + BDS + QZSS + IRNSS + SBAS

Figure 4. JFNG station.

4.2. Code Multipath Analysis

The BDS-3 MEO satellite (C22) and IGSO satellite (C38) are selected as examples for the analysis next, the multipath is extracted using the multipath analysis method in equation (8). Figure 5 shows the BDS-3 MEO satellite (C22) and IGSO satellite (C38) code multipath on B1I and B3I frequency. When the satellite just enters the visible range, the elevation is small, while the lower elevation angle leads to a lager code multipath and bias of dual frequency. This phenomenon may be due to the fact that signals from satellites with low elevation angles have a higher probability of occurrence. Compared with MEO satellites, the multipath of IGSO satellites with dual frequency show more obvious differences, and the maximum differences can reach the meter level. The bias of each code multipath and the difference between the multipath on dual frequency can seriously affect the accuracy of positioning. The low and high-frequency parts can be found in the multipath, which definitely reduces the positioning accuracy. Figure 6 shows number of visible BDS-3 satellites on DOY 176 and the corresponding dilution of precise (DOP). The number of tracked satellites varied between 6 and 11. The vertical-DOP (VDOP) and the horizontal-DOP (HDOP) values vary from 0.5 to 3, showing that BDS-3 is capable of delivering accurate positioning services independently.

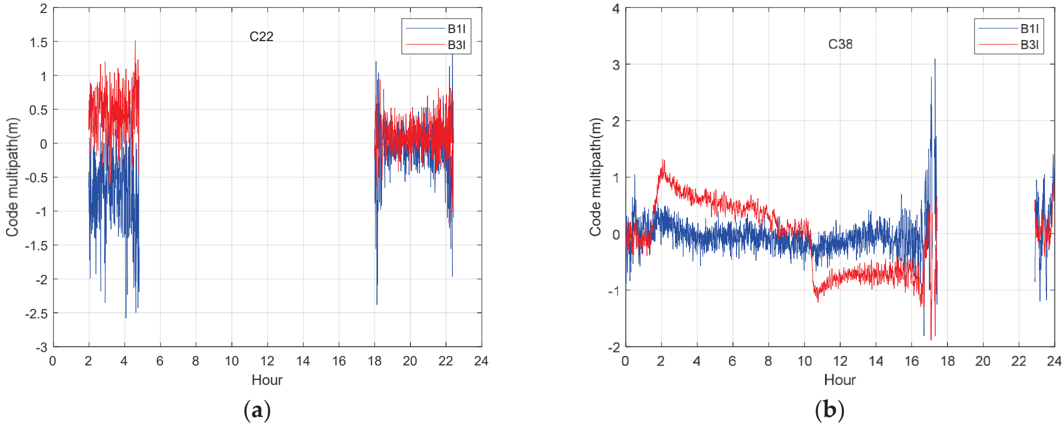


Figure 5. Code multipath at B1I and B3I of station JFNG on DOY 176,2021. (**a**) Code multipath at two frequencies of MEO satellite; (**b**) code multipath at two frequencies of IGSO satellite.

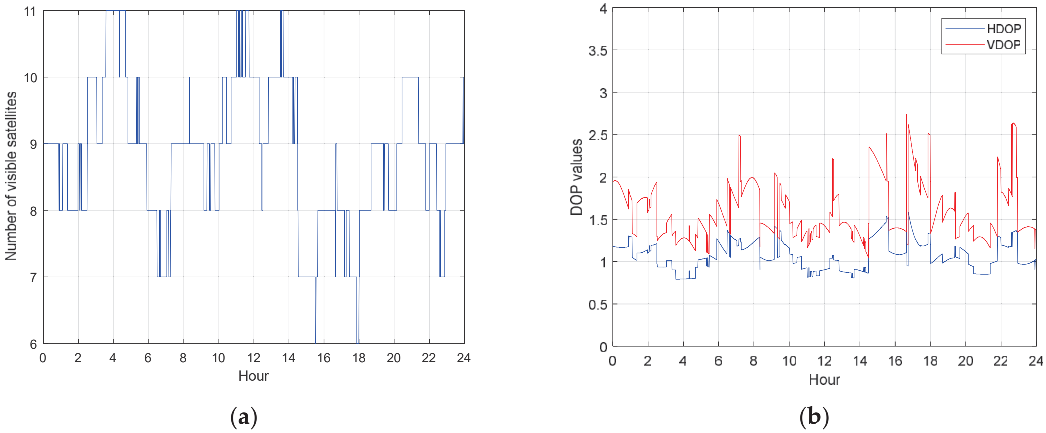
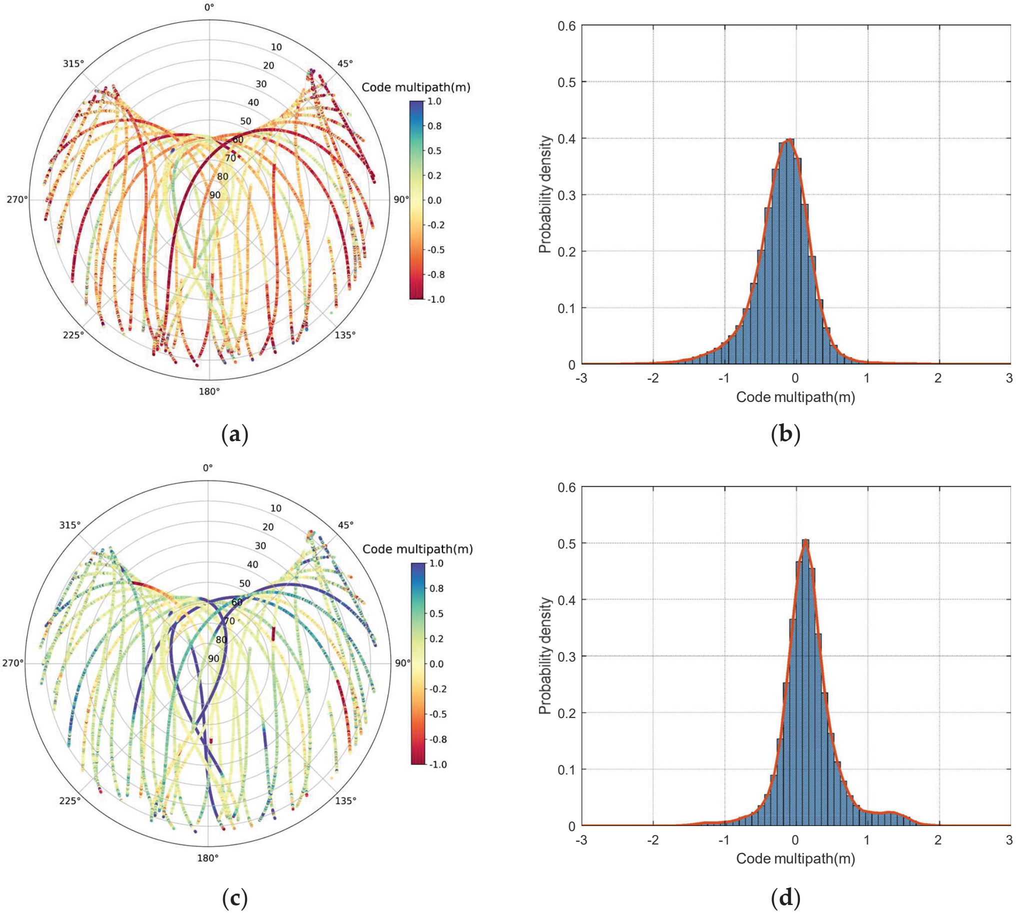


Figure 6. BDS-3 visible satellites on DOY 176. (a) number of BDS-3 visible satellites (b) HDOP and VDOP of BDS-3 satellites.

Figure 7 shows the probability density of the code multipath. Most code multipath range from −2 to 2 m, with almost 95% ranging from −1 to 1 m. The code multipath is obviously larger in the low elevation angle region. When the altitude angle is lower than 30°, the code multipath is generally larger than 1 m and the code multipath of signals in the high elevation regions are generally smaller than 1 m; therefore, the code multipath is highly correlated with the elevation angles. The most elevation-dependent effects are eliminated above 35° elevation [42]. This phenomenon occurs because most of the external effects can be eliminated in the signal accuracy analysis when the satellite altitude is higher than 35°. It is noted that there is a constant bias in multipath, actually this is caused by the unmodeled error (e.g., hardware delay) [14]. Fortunately, the constant bias present in the multipath will not influence multipath analysis, and it is reasonable to ignore the bias.



(**a**)

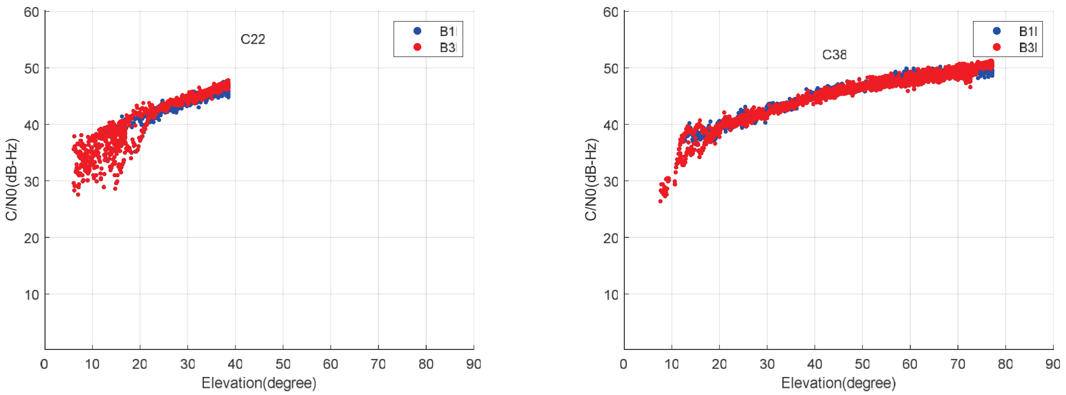
(**b**)

(**c**)

(**d**)

Figure 7. BDS-3 multipath sky map and histogram of code multipath. (**a**) B1I multipath sky map; (**b**) histogram of B1I code multipath; (**c**) B3I multipath sky map; (**d**) histogram of B3I code multipath. The red line in right panels is the envelope of the code multipath distribution.

Figure 8 shows the BDS-3 MEO satellite (C22) and IGSO satellite (C38) C/N0 on the B1I and B3I frequencies with respect to elevation angle. At lower elevations, the C/N0 is small, which corresponds to a large code multipath in Figure 5. The C/N0 value decreases significantly and the code multipath becomes large and divergent, particularly when the elevation angle is below 10°. This justifies the rationality of characterizing the variation of code multipath with C/N0.

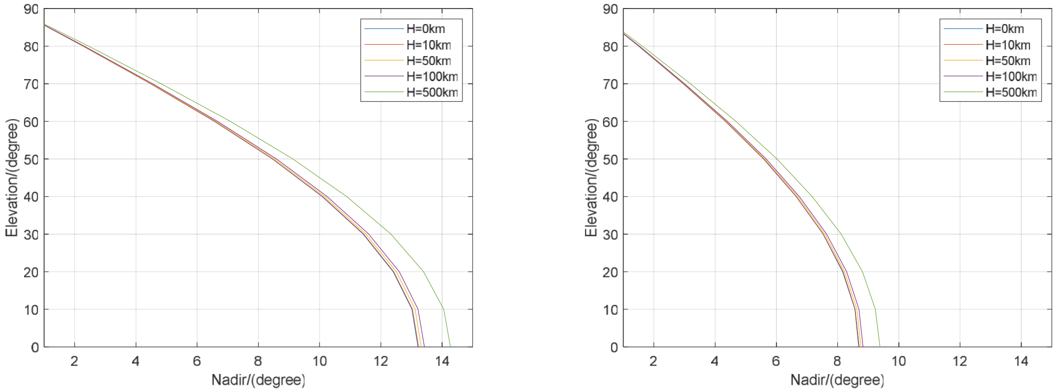


(**a**) (**b**)

Figure 8. C/N0 of B1I and B3I of MEO satellites (**a**) and IGSO (**b**) with respect to elevation angle.

4.3. Correlation Analysis of Nadir Angles and Code Multipath

Figure 9 demonstrates that the variation of the elevation angle of MEO and GEO/IGSO satellites to the receivers at different elevations with the nadir angle. Along with the growth in elevation angle, the range of the observable nadir angle increases. The elevation angle changes with the station altitude for the same nadir angle, and as the nadir angle increases, the difference in elevation angle at different latitudes becomes more noticeable. For example, if the nadir is 13.21° (H = 0 km, elevation = 0°), the elevation angles are 3.21°, 7.15°, 10.08° and 21.99° when the station is located at 10 km, 50 km, 100 km and 500 km, respectively. In conclusion, extra errors will be introduced into multipath error correction using elevation angle as the feature, according to previous studies.

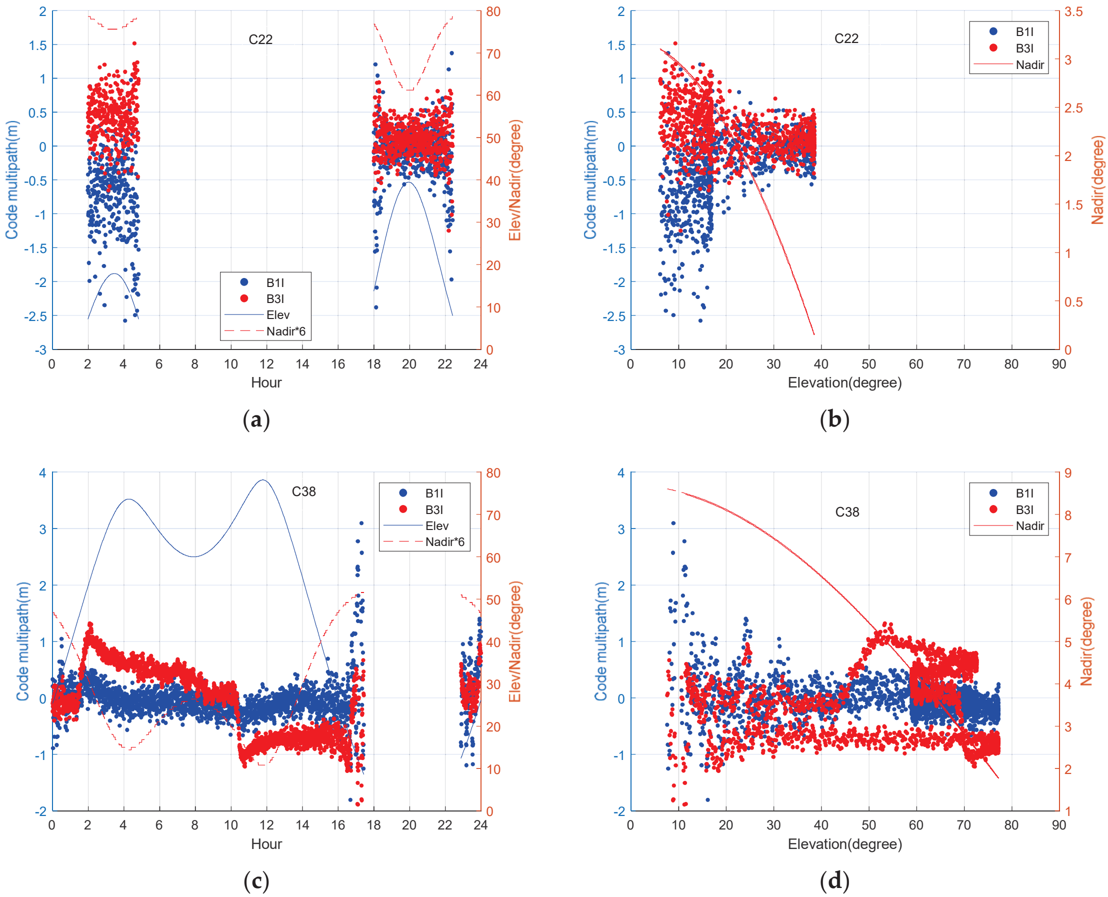


(**a**) (**b**)

Figure 9. Elevations of MEO satellites (**a**) and GEO/IGSO (**b**) with respect to receivers at different altitude as a function of nadir angle.

From Figure 10, when the elevation angle exceeds 30°, the code multipaths of C22 are between −0.5 and 0.5 m and those of C38 are between −1 and 1 m, and when the elevation angle is smaller than 30°, the code multipaths are obviously increased. It can be concluded that the smaller the elevation and nadir angles are, the larger the code multipath and the more discrete distribution are. These results demonstrate the correlation of code multipath with the elevation and nadir angle. Moreover, the relationship between the elevation angle and nadir angle exhibits a nonlinearity that can clearly be seen from the code multipath of

C38, which verifies that the nadir angle rather than the elevation angle should be used as the independent variable for multipath error modeling.



(a)

(b)

(c)

(d)

Figure 10. (**a**–**d**) Variation of BDS satellites multipath and nadir angles with respect to epoch or elevation angle. The amplified values of nadir angles are used to make the relationship clear.

4.4. Comparison of AT-Conv-LSTM with Other Methods

To assess the effectiveness of the multipath mitigation method based on the AT-Conv-LSTM network, we used the data from DOY 160–175 of JFNG station 2021 for training and those from DOY 176 for testing. The left panel in Figure 11 illustrates the uncorrected code multipaths of MEO (C22) and IGSO (C38) before the multipath correction and the multipath errors predicted by AT-Conv-LSTM, respectively. The right panel in Figure 11 illustrates the bias of the uncorrected code multipath minus the predicted multipath error. The MAE of the uncorrected multipath is 0.3322, and the MAE of multipath corrected by the AT-Conv-LSTM decreases to 0.0681. As can be seen, our method can effectively mitigate the code multipath, and can essentially correct the fluctuating low-frequency components. The corrected multipath errors are analogous to the white noise series, signifying that the AT-Conv-LSTM network is able to mitigate the multipath effectively.

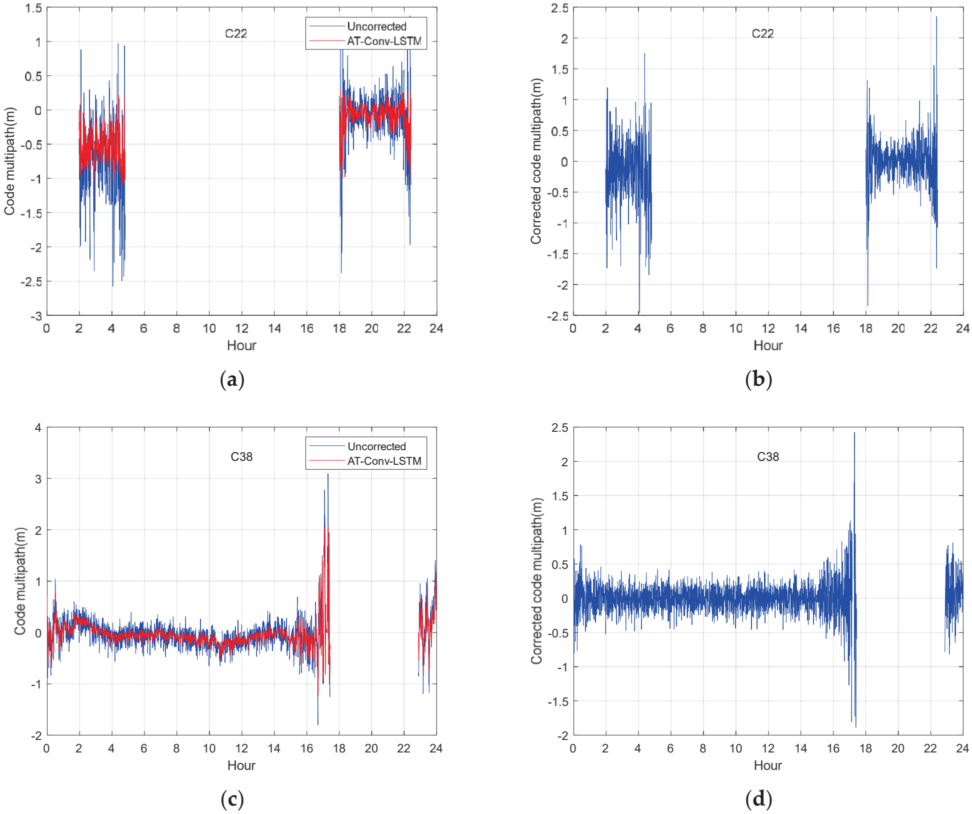


Figure 11. MEO satellite C22 multipath before (**a**) and after correction (**b**); IGSO satellite C38 multipath before (**c**) and after correction (**d**).

To confirm the effectiveness of multipath mitigation using the AT-Conv-LSTM network, we conducted comparative experiments with SF, MHM and T-MHM for comparison. The surrounding environments of the stations remain unchanged. The ORTs of BDS satellites are calculated on an individual basis in advance of the SF method. The multipath models were first established through the code multipath of the corresponding days using low-pass-filters, then were removed from the code multipath of next day. Figure 12 illustrates that the multipath errors on DOY 166 and DOY 173 of C12 satellite has a strong temporal correlation, the correlation coefficient is 0.82 and the multipath error of C38 satellite represents a strong temporal correlation in three days DOY 171, DOY 172 and DOY 173, the correlation coefficient is 0.79. Considering the differences between the IGSO and MEO orbits, we correct the multipath errors of the IGSO and MEO satellites individually. In this study, our analyses primarily focused on the ORTs of IGSO satellites (such as C38) and MEO satellites (such as C22). We extract the multipath errors of the IGSO and MEO satellites by using the data of the previous day and the previous seven days, respectively. The time advances of different orbital satellites are computed through code multipath correlation, in order to conduct multipath mitigation experiments in observation domain. As claimed by the analyses above, the ORT of GEO and IGSO satellites is around 86,165 s, while the ORT of MEO satellites is around 84,697 s and six days.

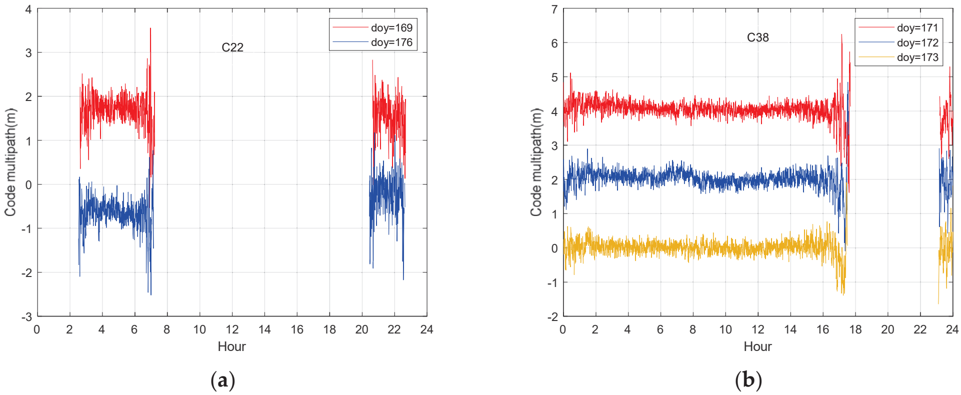


Figure 12. Comparison of the multipath errors for C22 (**a**) and C38 (**b**). The vertical 2 m offset was applied to the code multipath sequences for illustration purposes.

The MHM method for multipath error mitigation involves dividing the multipath into sky grids of certain sizes determined by the azimuth and elevation angles. Next, the average multipath from all satellites in the certain grid is used to construct a multipath error correction value table. We used the DOY 160–175 data to construct the MHM grids, as shown in Figure 13. The multipath curve obtained from MHM exhibits similarities to rectangular waves and is characterized by a deficiency in high-frequency information. Afterwards, T-MHM as an improved method is proposed to describe the multipath spatial distribution per grid specifically. The multipath code is divided into a grid in the sky, with dimensions determined by the azimuth and elevation angles. This grid is used to conduct trend-surface analysis on the multipath within each specific grid. The resulting trend-surface fitting coefficients are then stored. We filtered and denoised the multipath data from different satellites in the next experiments and removed the outliers based on the 3-sigma principle, then divided the grid points into sizes 1° × 1° consistent with the MHM. Finally, a linear function is adopted to fit the trend surface, and the coefficients are stored to construct the correction table.

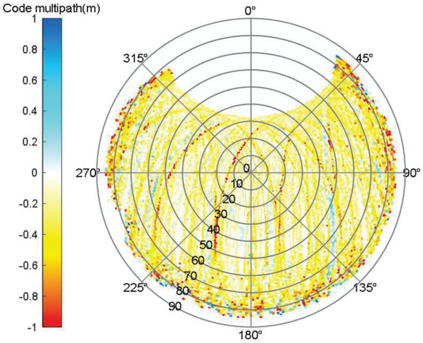
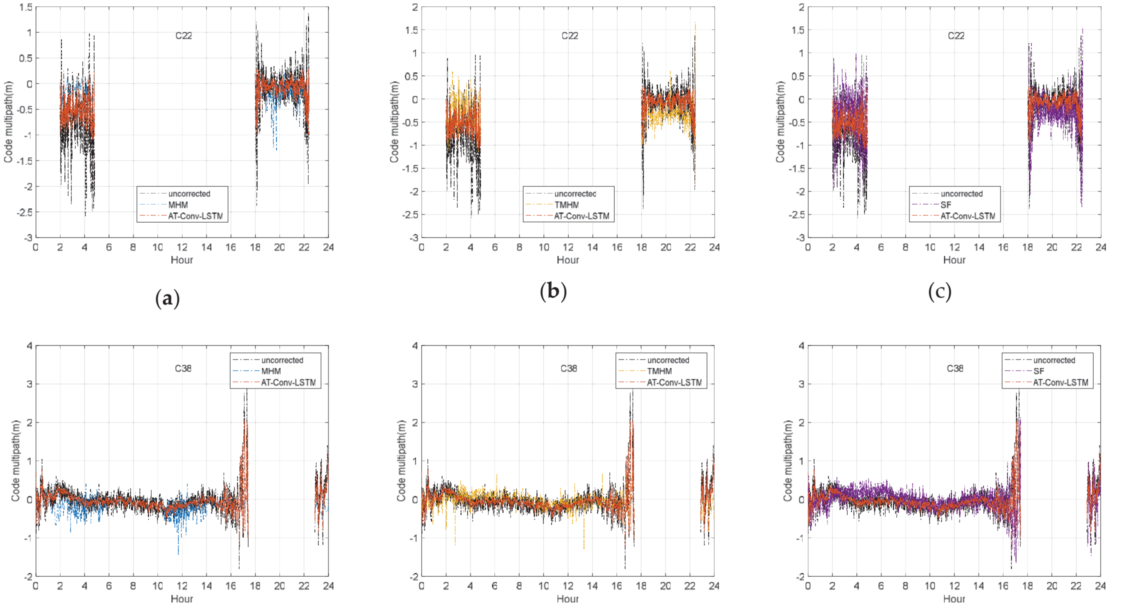


Figure 13. Sky map of the MHM grid.

The DOY 176 satellites data with two types of orbits such as MEO C22 (upper) and IGSO C38 (bottom) are used for comparison in Figure 14. Compared the estimated multipath from our proposed method with the uncorrected code multipath and SF, MHM and T-MHM model in Figure 14. The upper panel illustrates that other methods fit well for low-frequency fluctuations, but poorly for more obvious high-frequency fluctuations, especially in C22 satellites. However, our proposed method is most consistent with the variation trend of the uncorrected code multipath series.



(**a**) (**b**) (c)

(**d**) (**e**) (**f**)

Figure 14. The time series of the C22 MEO satellite uncorrected code multipath and the AT-Conv-LSTM-estimated code multipath compared with MHM (**a**), T-MHM (**b**) and SF (**c**). The time series of the C38 IGSO satellite uncorrected code multipath and AT-Conv-LSTM-estimated code multipath compared with MHM (**d**), T-MHM (**e**) and SF (**f**).

The experiment used 16 days of data for MHM modeling. The substandard quality of observation data on certain days and the limited fitting ability of the linear function within the grid points can affect the MHM and TMHM modeling. And the inaccurate estimation of the ORTs of different satellites may affect the performance of the SF. All these reasons could affect the effect of the other three methods for the multipath correction. As shown in the comparison results in Table 4, the other methods have some improvements in the MAE value, but the improvement in the RMSE is not obvious. Our method reduces the MAE value of code multipath from 0.3322 to 0.0681 for MEO satellite C23 and from 0.0614 to 0.0241 for IGSO satellite C39, and the MAE correction of code multipath is improved by about 70%. The code multipath can be corrected to a centimeter-level bias near 0. The corrected code multipath exhibits a reduction in magnitude by one order compared to the original uncorrected code multipath. And there is a corresponding improvement in RMSE, which is reduced from 0.5596 to 0.4747 in MEO satellite C23 and from 0.2766 to 0.2386 in IGSO satellite C39, and the RMSE of the code multipath is improved by about 13%. Due to space limitations, we will only use the B1I frequency point as an example in the above performance analysis and calculate the correction enhancement of the code multipath MAE using AT-Conv-LSTM, respectively. Taking C22, C23, C36, C38 and C39 as examples, the correction of the code multipath MAE can reach more than 60%.

Table 4. MAE and RMSE values of code multipath with and without multipath model correction in Figure 14.

PRN	Uncorrected		MHM-Corrected		T-MHM-Corrected		SF-Corrected		AT-Conv-LSTM-Corrected	
	MAE (m)	RMSE (m)	MAE (m)	RMSE (m)	MAE (m)	RMSE (m)	MAE (m)	RMSE (m)	MAE (m)	RMSE (m)
C22	0.3322	0.5596	0.1581	0.6154	0.0143	0.5731	0.0050	0.6608	0.0681	0.4447
C23	0.3322	0.5596	0.1797	0.6175	0.0143	0.5731	0.0050	0.6608	0.0681	0.4747
C36	0.3620	0.3809	0.1827	0.4225	0.0529	0.2952	0.0587	0.5641	0.0812	0.3615
C38	0.0340	0.3364	0.1106	0.3932	0.0020	0.3588	0.0340	0.3364	0.0480	0.4118
C39	0.0614	0.2766	0.0505	0.3501	0.0191	0.2686	0.0447	0.3514	0.0241	0.2386

5. Discussion

The current multipath analysis methods are limited to a single indicator representation. However, this cannot fully describe the characteristics of the multipath. First, we analyzed the strong correlation between code multipath and nadir angles, elevation angles, azimuth angles and C/N0 in "Section 4.2. Code Multipath Analysis". In "Section 4.3 Correlation Analysis of Nadir Angles and Code Multipaths", we explained the nonlinear relationship between nadir angles and elevation angles. Our analysis suggests that nadir angles, elevation angles, azimuth angles and C/N0 should be jointly considered as characteristic indicators of code multipaths. There is a constant bias in multipaths, caused by the unmodeled error (e.g., hardware delay), which can be ignored due to its stability. And the multipath is caused by lots of factors, such as the reflection, diffraction and obstruction of signals by local obstacles. In the future, we might work on a more accurate multipath analysis method to adapt different types of multipath.

The AT-Conv-LSTM network spatial features of elevation angles, nadir angles, azimuth angles and C/N0 through convolutional layers and extracts temporal features from the output sequence of the convolutional layers through LSTM. An attention mechanism is introduced to automatically allocate greater weights to the visible periods of satellites. These modules enable AT-Conv-LSTM to better extract spatiotemporal features. In our simulation, the sliding window size cannot be set too small. Since the variation of elevation, azimuth and zenith angles between adjacent epoch elements is inconspicuous, a small sliding window size will result in the inability to adequately extract the temporal features. The sliding window size is configured as 120 points for a 1 h observation data with 30 s sampling intervals in our study. In addition, a dropout layer is added to prevent the network from overfitting, and the dropout rate is set to 0.6. From Figure 14 and Table 4, it can also be seen that the multipath correction results of AT-Conv-LSTM network can correct the code residuals to a deviation of around zero, which is one degree lower than original uncorrected code residuals and also shows improvement compared to the other three methods. Furthermore, we aim to characterize the phase multipath error with the elevation angle, azimuth angle, nadir angle and C/N0 value as multiple indicators. Specifically, a more suitable deep learning model can be further investigated to mitigate the phase multipath error effectively.

6. Conclusions

The multipath, which is widely recognized as the most challenging remaining error, imposes constraints on the accuracy of GNSS positioning. Despite efforts to develop precise error models, the multipath remains a significant source of error that hinders high-precision positioning. This study focuses on extracting the multipath error from the raw BDS-3 code observations using the undifferenced and uncombined PPP model. Firstly, we extracted the multipath error from the raw BDS-3 code observations based on the undifferenced and uncombined PPP model. Since the amplitude and phase of multipath signal rely on the position of satellite and receiver in addition to the environment, the correlation among the multipath error and elevation, nadir and azimuth angle is analyzed. Correspondingly, we analyzed the non-linear relationship between elevation and nadir angles. Therefore, azimuth angle, elevation angle, nadir angle and C/N0 are taken as multiple indicators to characterize the multipath significance. Furthermore, an AT-Conv-LSTM network is proposed to exploit the temporal correction from the multiple indicators' changing pattern over time and exploit the spatial correction from the multiple indicators' changing pattern over angles. Thus, our method can maximize the temporal and spatial repeatability of the multipath for real-time multipath mitigation. And the proposed method takes into account the spatial distribution of multipath without requiring ORTs, which can also correct both low and high-frequency components of multipath errors. Finally, our method significantly decreases multipath MAE and RMSE in comparison to SF, MHM and TMHM. Moreover, it has the capability to correct code multipath with a deviation at the centimeter level. Therefore, the proposed AT-Conv-LSTM network could mitigate the multipath efficiently and will be of broad practical value in the fields of standard positioning service and high-precision positioning.

Author Contributions: Conceptualization, J.S. and J.W.; methodology, J.S. and Z.T.; software, J.S.; validation, J.S., J.W., Z.T. and C.Z.; formal analysis, Z.T.; investigation, J.S. and Z.T.; writing—original draft preparation, J.S. and Z.T.; writing—review and editing, J.S., C.Z. and J.W.; visualization, C.Z.; supervision, J.W. and Z.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Nos. 62171191 and 62271223).

Data Availability Statement: The GNSS raw observation data was available at https://cddis.nasa.gov/archive/gnss/ (accessed on 2 August 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dow, J.M.; Neilan, R.E.; Rizos, C. The International GNSS Service in a changing landscape of Global Navigation Satellite Systems. *J. Geod.* **2009**, *83*, 191–198. [CrossRef]
2. Zhang, Z.; Li, Y.; He, X.; Hsu, L. Resilient GNSS real-time kinematic precise positioning with inequality and equality constraints. *GPS Solut.* **2023**, *27*, 116. [CrossRef]
3. Zang, J.; Xu, C.; Li, X. Scaling earthquake magnitude in real time with high-rate GNSS peak ground displacement from variometric approach. *GPS Solut.* **2020**, *24*, 101. [CrossRef]
4. Su, M.; Zheng, F.; Shang, J.; Qiao, L.; Qiu, Z.; Zhang, H. Influence of flooding on GPS carrier-to-noise ratio and water content variation analysis: A case study in Zhengzhou, China. *GPS Solut.* **2023**, *27*, 21. [CrossRef]
5. Zhang, Z.; Li, Y.; He, X.; Chen, W.; Li, B. A composite stochastic model considering the terrain topography for real-time GNSS monitoring in canyon environments. *J. Geod.* **2022**, *96*, 79. [CrossRef]
6. Seepersad, G.; Bisnath, S. Reduction of PPP convergence period through pseudorange multipath and noise mitigation. *GPS Solut.* **2015**, *19*, 369–379. [CrossRef]
7. Lau, L.; Cross, P. Development and testing of a new ray-tracing approach to GNSS carrier-phase multipath modelling. *J Geod.* **2007**, *81*, 713–732. [CrossRef]
8. Suzuki, T.; Matsuo, K.; Amano, Y. Rotating GNSS Antennas: Simultaneous LOS and NLOS Multipath Mitigation. *GPS Solut.* **2020**, *24*, 86. [CrossRef]
9. Hsu, L.; Jan, S.; Groves, P.; Kubo, N. Multipath mitigation and NLOS detection using vector tracking in urban environments. *GPS Solut.* **2015**, *19*, 249–262. [CrossRef]
10. Zhang, Z.; Yuan, H.; Li, B.; He, X.; Gao, S. Feasibility of easy-to-implement methods to analyze systematic errors of multipath, differential code bias, and inter-system bias for low-cost receivers. *GPS Solut.* **2021**, *25*, 116. [CrossRef]
11. Luo, X.; Mayer, M.; Heck, B.; Awange, J. A realistic and easy-to-implement weighting model for GPS phase observations. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 6110–6118. [CrossRef]
12. Zhang, Z. Code and phase multipath mitigation by using the observation-domain parameterization and its application in five-frequency GNSS ambiguity resolution. *GPS Solut.* **2021**, *25*, 144. [CrossRef]
13. Yuan, H.; Zhang, Z.; He, X.; Dong, Y.; Zeng, J.; Li, B. Multipath mitigation in GNSS precise point positioning using multipath hierarchy for changing environments. *GPS Solut.* **2023**, *27*, 193. [CrossRef]

14. Zhang, Z.; Li, B.; Gao, Y.; Shen, Y. Real-time carrier phase multipath detection based on dual-frequency C/N0 data. *GPS Solut.* **2019**, *23*, 7. [CrossRef]
15. Genrich, J.F.; Bock, Y. Rapid resolution of crustal motion at short ranges with the Global Positioning System. *J. Geophys. Res. Solid Earth.* **1992**, *97*, 3261–3269. [CrossRef]
16. Choi, K.; Bilich, A.; Larson, K.M.; Axelrad, P. Modified sidereal filtering: Implications for high-rate GPS positioning. *Geophys Res. Lett.* **2004**, *31*, 178–198. [CrossRef]
17. Takasu, T. High-rate Precise Point Positioning: Detection of crustal deformation by using 1-Hz GPS data. In Proceedings of the GPS/GNSS Symposium 2006, Tokyo, Japan, 15–17 November 2006; pp. 1–8.
18. Ragheb, A.E.; Clarke, P.J.; Edwards, S.J. GPS sidereal filtering: Coordinate- and carrier-phase-level strategies. *J Geod.* **2007**, *81*, 325–335. [CrossRef]
19. Hung, H.K.; Rau, R.J. Surface waves of the 2011 Tohoku earthquake: Observations of Taiwan's dense high-rate GPS network. *J. Geophys. Res. Solid Earth* **2013**, *118*, 332–345. [CrossRef]
20. Atkins, C.; Ziebart, M. Effectiveness of observation-domain sidereal filtering for GPS precise point positioning. *GPS Solut.* **2016**, *20*, 111–122. [CrossRef]
21. Cohen, C.E.; Parkinson, B.W. Mitigating multipath error in GPS based attitude determination. In Proceedings of the Advances in the Astronautical Sciences, AAS Guidance and Control Conference, San Diego, CA, USA, 2–6 February 1991; pp. 74–78.
22. Fuhrmann, T.; Luo, X.; Knöpfer, A.; Mayer, M. Generating statistically robust multipath stacking maps using congruent cells. *GPS Solut.* **2015**, *19*, 83–92. [CrossRef]
23. Dong, D.; Wang, M.; Chen, W.; Zeng, Z.; Song, L.; Zhang, Q.; Cai, M.; Cheng, Y.; Lv, J. Mitigation of multipath effect in GNSS short baseline positioning by the multipath hemispherical map. *J. Geod.* **2016**, *90*, 255–262. [CrossRef]
24. Wang, Z.; Chen, W.; Dong, D.; Wang, M.; Cai, M.; Yu, C.; Zheng, Z.; Liu, M. Multipath mitigation based on trend surface analysis applied to dual-antenna receiver with common clock. *GPS Solut.* **2019**, *23*, 104. [CrossRef]
25. Lu, R.; Chen, W.; Dong, D.; Wang, Z.; Zhang, C.; Peng, Y.; Yu, C. Multipath mitigation in GNSS precise point positioning based on trend-surface analysis and multipath hemispherical map. *GPS Solut.* **2021**, *25*, 119. [CrossRef]
26. Lu, R.; Chen, W.; Zhang, C.; Li, L.; Peng, Y.; Zhang, Z. Characteristics of the BDS-3 multipath effect and mitigation methods using precise point positioning. *GPS Solut.* **2022**, *26*, 41. [CrossRef]
27. Zhang, Z.; Dong, Y.; Wen, Y.; Luo, Y. Modeling, refinement and evaluation of multipath mitigation based on the hemispherical map in BDS2/BDS3 relative precise positioning. *Measurement* **2023**, *213*, 112722. [CrossRef]
28. Kaselimi, M.; Voulodimos, A.; Doulamis, N.; Doulamis, A.; Delikaraoglou, D. Deep recurrent neural networks for ionospheric variations estimation using gnss measurements. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–15. [CrossRef]
29. Li, Q.; Su, X.; Xie, X.; Tao, C.; Cui, J.; Chen, H.; Liu, Z. Accuracy Analysis of Error Compensation in the Ionospheric Model of BDS Broadcasting Based on ABC-BP Neural Network. In Proceedings of the China Satellite Navigation Conference (CSNC 2022), Lecture Notes in Electrical Engineering, Beijing, China, 25–27 May 2022; pp. 54–63. [CrossRef]
30. Liu, C.; Yao, Y.; Xu, C. Conventional and neural network-based water vapor density model for GNSS troposphere tomography. *GPS Solut.* **2022**, *26*, 4. [CrossRef]
31. Chen, H.; Niu, F.; Su, X.; Geng, T.; Liu, Z.; Li, Q. Initial Results of Modeling and Improvement of BDS-2/GPS Broadcast Ephemeris Satellite Orbit Based on BP and PSO-BP Neural Networks. *Remote Sens.* **2021**, *13*, 4801. [CrossRef]
32. Zhao, L.; Li, N.; Li, H.; Wang, R.; Li, M. BDS satellite clock prediction considering periodic variations. *Remote Sens.* **2021**, *13*, 4058. [CrossRef]
33. Kim, H.; Bae, T. Deep learning-based GNSS network-based real-time kinematic improvement for autonomous ground vehicle navigation. *J. Sensors* **2019**, *2019*, 3737265. [CrossRef]
34. Taghizadeh, S.; Safabakhsh, R. An integrated INS/GNSS system with an attention-based hierarchical LSTM during GNSS outage. *GPS Solut.* **2023**, *27*, 71. [CrossRef]
35. Min, D.; Kim, M.; Lee, J.; Circiu, M.S.; Meurer, M.; Lee, J. DNN-Based Approach to Mitigate Multipath Errors of Differential GNSS Reference Stations. *IEEE T. Intell. Transp.* **2022**, *23*, 25047–25053. [CrossRef]
36. Erhan, D.; Bengio, Y.; Courville, A.; Manzagol, P.A.; Vincent, P.; Bengio, S. Why does unsupervised pre-training help deep learning? *J. Mach. Learn. Res.* **2010**, *11*, 625–660.
37. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Polosukhin, I. Attention is All You Need. In *Advances in Neural Information Processing Systems, Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, California, USA, 4–9 December 2017*; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 6000–6010.
38. Tang, J.; Li, Y.; Ding, M.; Liu, H.; Yang, D.; Wu, X. An Ionospheric TEC Forecasting Model Based on a CNN-LSTM-Attention Mechanism Neural Network. *Remote Sens.* **2022**, *14*, 2433. [CrossRef]
39. Zheng, H.; Lin, F.; Feng, X.; Chen, Y. A Hybrid Deep Learning Model With Attention-Based Conv-LSTM Networks for Short-Term Traffic Flow Prediction. *IEEE T. Intell. Transp.* **2021**, *22*, 6910–6920. [CrossRef]
40. Ruan, R.; Jia, X.; Feng, L. Analysis on BDS Satellite Internal Multipath and Its Impact on Wide-lane FCB Estimation. *Acta Geod. Cartogr. Sin.* **2017**, *46*, 961–970. [CrossRef]

41. CSNO. The Application Service Architecture of BeiDou Navigation Satellite System (Version 1.0). Available online: http://en.beidou.gov.cn/SYSTEMS/Officialdocument/201812/P020181227424526837905.pdf (accessed on 15 June 2021).
42. Zhang, Z.; Li, B.; Shen, Y.; Yang, L. A noise analysis method for GNSS signals of a standalone receiver. *Acta Geod. Geophys.* **2017**, *52*, 301–316. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

A Low-Cost and Robust Multi-Sensor Data Fusion Scheme for Heterogeneous Multi-Robot Cooperative Positioning in Indoor Environments

Zhi Cai [1,2], Jiahang Liu [1,*], Weijian Chi [1] and Bo Zhang [3]

[1] College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China; zhi.cai@nuaa.edu.cn (Z.C.); wj_chi@nuaa.edu.cn (W.C.)
[2] School of Aeronautic Engineering, Nanjing Vocational University of Industry Technology, Nanjing 210023, China
[3] School of Network and Communication, Nanjing Vocation College of Information Technology, Nanjing 210023, China; zhangbo@njcit.cn
* Correspondence: jhliu@nuaa.edu

Abstract: The latest development of multi-robot collaborative systems has put forward higher requirements for multi-sensor fusion localization. Current position methods mainly focus on the fusion of the carrier's own sensor information, and how to fully utilize the information of multiple robots to achieve high-precision positioning is a major challenge. However, due to the comprehensive impact of factors such as poor performance, variety, complex calculations, and accumulation of environmental errors used by commercial robots, the difficulty of high-precision collaborative positioning is further exacerbated. To address this challenge, we propose a low-cost and robust multi-sensor data fusion scheme for heterogeneous multi-robot collaborative navigation in indoor environments, which integrates data from inertial measurement units (IMUs), laser rangefinders, cameras, and so on, into heterogeneous multi-robot navigation. Based on Discrete Kalman Filter (DKF) and Extended Kalman Filter (EKF) principles, a three-step joint filtering model is used to improve the state estimation and the visual data are processed using the YOLO deep learning target detection algorithm before updating the integrated filter. The proposed integration is tested at multiple levels in an open indoor environment following various formation paths. The results show that the three-dimensional root mean square error (RMSE) of indoor cooperative localization is 11.3 mm, the maximum error is less than 21.4 mm, and the motion error in occluded environments is suppressed. The proposed fusion scheme is able to satisfy the localization accuracy requirements for efficient and coordinated motion of autonomous mobile robots.

Keywords: multi-robot system; collaborative positioning; sensor integration; EKF; visual inspection

Citation: Cai, Z.; Liu, J.; Chi, W.; Zhang, B. A Low-Cost and Robust Multi-Sensor Data Fusion Scheme for Heterogeneous Multi-Robot Cooperative Positioning in Indoor Environments. *Remote Sens.* **2023**, *15*, 5584. https://doi.org/10.3390/rs15235584

Academic Editor: Andrzej Stateczny

Received: 26 September 2023
Revised: 22 November 2023
Accepted: 29 November 2023
Published: 30 November 2023

1. Introduction

Multi-robot collaborative positioning technology has found extensive applications in defense and civil fields—including robot formation combat, unmanned driving, robot team field exploration, unmanned disaster rescue, and unmanned carrier cooperative handling—owing to the advancements in artificial intelligence, computer technology, and automation. The multi-robot system is more than just a combination of robots; it involves collaboration and cooperation. Each robot is equipped with different sensors to fuse information based on task requirements. The robots within the group maintain spatial relationships and achieve collaborative positioning through complex communication and information exchange processes. This enables support for formation patrol, emergency risk avoidance, sudden stop, and obstacle avoidance autonomously. Multi-robot cooperative positioning technology outperforms traditional single-robot positioning technology in terms of enabling faster global awareness and planning, thus enhancing the robustness and autonomous positioning capability of multi-robot systems through capability complementarity and redundancy increase. Additionally, the use of small robots with a single function facilitates fault detection and functional upgrades.

Positioning in robot navigation relies on sensors to perceive the surrounding environment, serving as the foundation and prerequisite for task execution. Common indoor robot sensors include inertial measurement units (IMU), odometers, vision sensors, laser range finders, and sonar detectors. Each sensor has its own advantages and limitations. Sensors that operate without relying on external signals offer fast sampling speed but suffer from significant cumulative error. Laser rangefinders exhibit high accuracy but can only determine object distances directly. Vision sensors excel in perceiving the environment and capturing rich environmental information. However, vision detection and localization algorithms necessitate high system computational performance, are influenced by ambient light and target characteristics [1], and do not provide global positioning information for the robot. The fusion of the advantageous performance of each sensor is a promising research area aiming to reduce costs, enhance system robustness and flexibility, and enable seamless collaborative localization in multi-robot systems through the fusion of multiple information sources.

Multi-sensor fusion is a well-established algorithm in the field of robotics and control [2–4]. It aims to integrate multiple sensors to enhance system performance and achieve precise positioning. Researchers worldwide have extensively investigated the robot localization problem by leveraging sensor information fusion. For example, Yan Lu and Joseph Lee et al. [5] fused laser range information and sensor data from monocular vision to propose a fusion scheme for acquiring position information in GPS-denied environments. Howard and Gaurav S. fused mutual observation data from two robots [6] and presented a method to localize members of a mobile robot team using the robots themselves as landmarks. Other researchers have employed various fusion algorithms, such as the extended Kalman filter method [7,8], which assumes a Gaussian distribution for systematic error and effectively addresses the prediction problem of nonlinear systems. Anastasios used an extended Kalman filter (EKF)-based algorithm for real-time vision-aided inertial navigation, which improved the fault tolerance performance of the measurement data [9]. There is also the use of great likelihood estimators to derive relative positions between robots [10,11], or real-time position estimation of robots using particle filtering [12–14]. I.M. Rekleitis estimated a larger group of robots that can mutually determine one another's position (in 2D or 3D) and uncertainty using a sample-based (particle filter) model of uncertainty in a specific scenario [15]. The application of co-localization results in multi-robot systems is also becoming a key issue of current interest. Boda Ning et al. were concerned with the collective behavior of robots beyond the nearest neighbor rule [16]. Brian Shucker proposed a control and communication scheme for multi-robot systems in long-distance situations with his team [17].

The limitations of the existing multi-source co-location algorithms include a lack of real-time comprehensive state estimation from the sensors of different types of robots and the consideration of calculation burden, cost, and rigorous mathematical models. Most of the above research belongs to the primary stage; some have provided integration schemes for single-robot application scenarios. Furthermore, some research has the potential to involve multi-robot positioning and dynamic performance estimation realized by positioning each robot independently, but the actual systems are limited by high computation and maintenance costs. In addition, the state estimation performance degradation is more serious, partly caused by multistage errors in practical applications and the limitations of the current algorithm architecture.

As far as we know, there is a lack of research providing insight into the fusion of the laser rangefinder, camera, and other sensor data in heterogeneous robotic systems with varying levels of complexity and sensor configurations. This work makes three primary contributions. First, the co-position process gathers observations from both the robot itself and other robots, providing increased flexibility and robustness in sensor information fusion. Second, by leveraging the advantageous performance of diverse sensors, a variety of observation and communication tasks can be dynamically assigned to different types of robots, thereby improving overall task fulfillment and the system's anti-interference capabilities. Third, the system can utilize small robots with simple structures and single functions, simplifying upgrades and maintenance compared to complex, independent robot systems.

The remainder of the paper is organized as follows. In Section 2, we present a method for a multi-sensor data fusion co-position scheme applicable to heterogeneous multi-robot systems and common navigation modes. Section 3 analyzes practical errors and suggests improvements. Section 4 showcases our experimental results, and Section 5 provides the conclusions.

2. Methods

2.1. Integration Architecture

Considering the flexibility, practicality, and scalability of the multi-robot system, the navigation mode chosen is a common Leader–Follower robot swarm. The main structure of the proposed integration is presented in Figure 1 and consists of three steps.

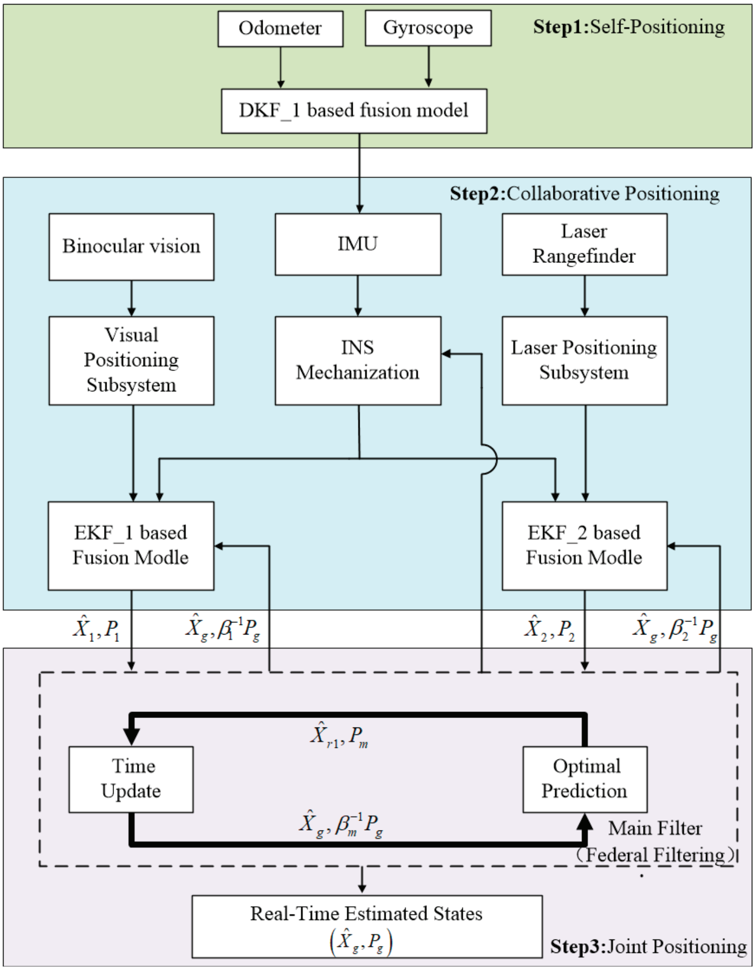


Figure 1. Proposed joint filtering integration scheme.

The integration involves three sub-filters and one main filter. The first discrete Kalman filter, DKF_1, fuses the odometer and gyroscope information within the robot to determine the inertial system's state equation. The extended Kalman filters, EKF_1 and EKF_2, estimate the target localization information using the vision sensor and laser rangefinder, respectively, with the inertial guidance system as a reference. The vision localization subsystem employs the improved YOLO deep-learning-based target detection algorithm and camera calibration algorithm to obtain target localization information. This information is then used as the observation input to estimate the target localization using the extended Kalman filter. Similarly, the laser ranging subsystem utilizes the angle between the target's horizontal projection and the optical axis of the camera, obtained from the vision sensor, to obtain positioning information. This positioning information is used as the observation input to estimate the positioning state of the laser rangefinder using the extended Kalman filter. The main filter calculates the information allocation factor for each sub-filter and combines the estimated results using the extended Kalman filter, yielding accurate target localization.

2.2. Coordinate Frames Definition and Transformation

2.2.1. Definition of Global and Local Coordinate Frames

The positioning solutions provided by different sensors are expressed in different navigation systems and then integrated into a unified navigation frame for data fusion. In this paper, we mainly use the definition and transformation of global and local coordinate systems to describe the positioning of the surrounding environment, targets, and obstacles, and the corresponding coordinate systems are shown in Figure 2.

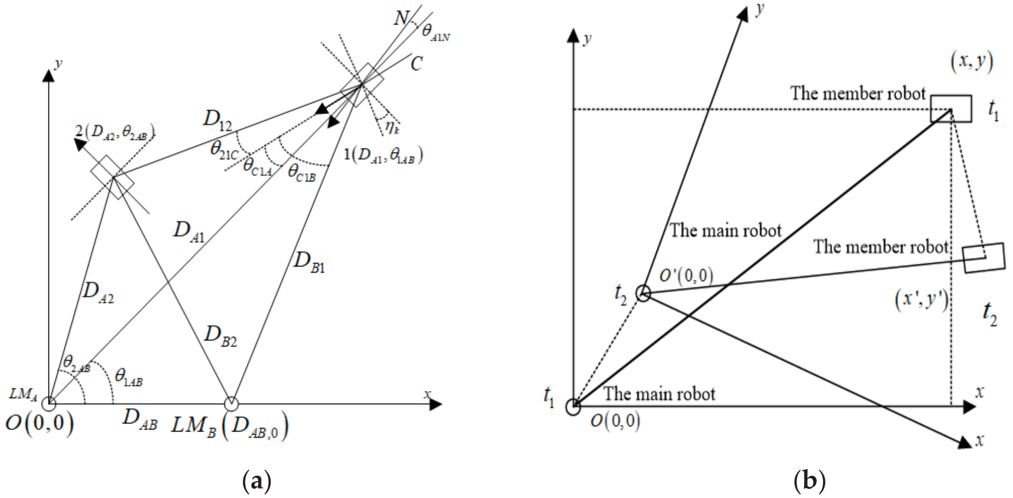


Figure 2. Coordinate system framework for multi-robot systems: (**a**) global coordinate frame; (**b**) local coordinate frame.

The elements in the global coordinate frames are described as follows, where 1 and 2 represent the main robot and member robots, respectively. LM_A and LM_B are artificial road signs. N indicates the main robot heading, and D_{AB} is the camera optical axis centerline direction of the main robot. H_k is the angle between the main robot heading N and the camera's optical axis centerline direction. The robot-equipped sensors provide the following data: D_{12}, D_{A1}, D_{B1}, θ_{21C}, θ_{C1A}, and θ_{C1B}. D_{12}, $D_{A1,}$ and D_{B1} can be obtained from the laser range scanner, while θ_{21C}, θ_{C1A}, and θ_{C1B} can be read by the vision sensors. D_{AB} and η_K represent known observation data. Based on the geometric relationships before each robot and the road signs, the positions of the member robots in the global coordinate system can be solved for and expressed in polar coordinates as

$$D_{A2} = \sqrt{D_{12}{}^2 + D_{B1}{}^2 - 2D_{12}D_{B1}\cos(\theta_{21c} + \theta_{C1A})} \tag{1}$$

$$(D_{A2}, \theta_{2AB}) = \left(D_{A2}, \arccos\left(\frac{D_{A2}{}^2 + D_{AB}{}^2 - (D_{12}{}^2 + D_{B1}{}^2 - 2D_{12}D_{B1}\cos(\theta_{21c} + \theta_{C1B}))}{2D_{A2}D_{AB}}\right)\right) \tag{2}$$

Similarly, the position of the master robot in the global coordinate system can be expressed in polar coordinate form as

$$(D_{A1}, \theta_{1AB}) = \left(D_{A1}, \arccos\left(\frac{D_{A1}{}^2 + D_{AB}{}^2 - D_{B1}{}^2}{2D_{A1}D_{AB}}\right)\right) \tag{3}$$

The heading angle of the main robot at this point can be calculated as follows:

$$\theta = \theta_{1AB} + \eta_k - \theta_{C1A} \tag{4}$$

The local coordinate frame is employed to describe the relative positions of individual robots in a multi-robot queue, including the updated relationships during their travel. In this coordinate system, the current position of the main robot serves as the origin, and the current heading represents the positive x-axis. At time t_1 in the figure, the position of the member robot in the $x\ o\ y$ coordinate system is (x, y). The heading and position of the main robot change as it moves. At time t_2, the coordinate system is updated to $xo'y$, and the position of the member robot is updated to (x', y'). The time interval ($t_2 - t_1$) is defined as the coordinate system update step, during which the $x\ o\ y$ coordinate system is utilized for both the main robot and the member robots.

2.2.2. Transformation of Visual Coordinate Frames

The purpose of using binocular vision [18,19] is to obtain the distance D between the target position and the camera by imaging the target in the vision sensor, and the angle θ_k between the horizontal 2D spatial projection of the target in 3D space and the centerline of the camera's optical axis. In the camera point model, the point $[x, y, z, l]^T$ in 3D space is converted into the 2D spatial projection coordinates $[u, v, l]^T$, which can be described by the following equation:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{z}\begin{bmatrix} f_{cx} & 0 & c_x & 0 \\ 0 & f_{cy} & c_y & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{5}$$

where f_{cx} and f_{cy} are the components of the camera focal length in the x and y direction respectively, and $(c_x, c_y)^{\mathrm{T}}$ are the coordinates of the main point of the image. Due to daylight refraction, the imaging lens of the camera produces a certain radial deformation, which eventually affects the imaging of the target in the camera. Considering the effect of the mirror deformation factor, the projection coordinates of the point in 3D space in 2D space can be rewritten as

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{z} * \begin{bmatrix} f_{cx} & 0 & c_x & 0 \\ 0 & f_{cy} & c_y & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}\left(1 + k_1r^2 + k_2r^4 + k_3r^6\right) \tag{6}$$

where k_1, k_2, and k_3 are the refraction coefficients of light, $r^2 = (x/z)^2 + (y/z)^2$. The parameters f_{cx}, f_{cy}, c_x, c_y, k_1, k_2, and k_3 can be obtained by the camera calibration algorithm.

After removing the radial deformation, the coordinates matrix of the target in the imaging plane are

$$\begin{bmatrix} u^c \\ v^c \end{bmatrix} = \frac{1}{(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)} * \begin{bmatrix} u \\ v \end{bmatrix} \tag{7}$$

The binocular vision imaging model is shown in Figure 3; vou and v'ou are the image imaging plane, z and z' are the central optical axis of the camera, $p(u_p, v_p)$ and $p'(u_p{}', v_p{}')$ are the target O in the two planes of the imaging point.

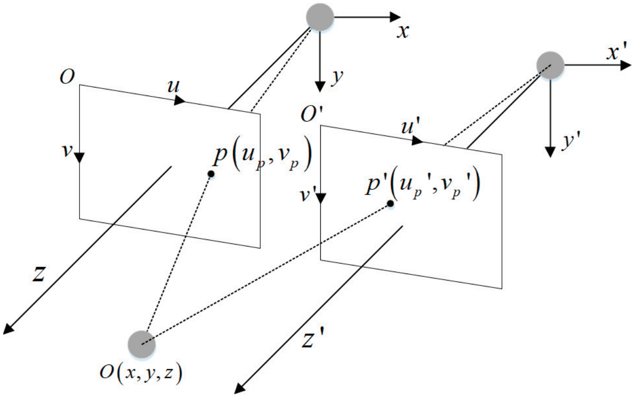


Figure 3. Binocular vision imaging model.

Then, the angle η between the target O and the two camera's center optical axes in the horizontal plane can be expressed as

$$\eta = \arctan\left(\frac{|(W/2) - u^c|}{(W/2)} \tan \alpha \right) \tag{8}$$

where W is the actual physical width of the imaged image and 2α is the horizontal tensor angle of the camera.

2.3. Model Design

Referring to the joint filtering scheme in Figure 1, the model is designed using Leader–Follower as the navigation mode. Two types of robots in the swarm are equipped with different sensors. The main robot is equipped with internal sensors (odometer and gyroscope) as well as external sensors (laser range scanner and binocular vision system), and the member robots are only equipped with odometers, gyroscopes, and sonar arrays to reduce the cost. During the formation march, each robot performs self-localization using its internal sensor data. Simultaneously, the main robot utilizes vision and laser devices to observe the positions of the current member robots. However, due to significant cumulative errors in the internal sensors and observation errors caused by time delay and drift in the external sensors, the acquired positioning data deviate considerably from the actual positions.

2.3.1. Odometer and Gyroscope Integration (Step 1—Self Positioning)

Deducing the robot's position and heading accuracy solely from the odometer data leads to relative cumulative errors. To enhance the accuracy, it is necessary to fuse these data with the gyroscope data. Given the linear system characteristics, this paper employs a discrete Kalman filter to fuse the data from both sources. Let $X = [v_L, v_R, w_g]^T$ represent the state variables. The state equation of the system can then be expressed as

$$X_{k+1} = A_{k+1} X_k + w_k \tag{9}$$

where A_{k+1} is the state transfer matrix from k to k + 1 moments and W_k is the Gaussian white noise with covariance matrix Q. Referring to Deng et al.'s use of the sensor model in the outdoor mobile robot localization method [20], the transfer matrix is

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1/\beta D_{RL} & 1/\beta D_{RL} & 0 \end{bmatrix} \tag{10}$$

where β is the odometer correction factor determined down from the actual test data, and the system observation equation is

$$Z_k = H_k X_k + v_k \tag{11}$$

where the observations can be read from the odometer and gyroscope and the measurement matrix H_k is the unit matrix I.

2.3.2. Distributed Data Fusion of Visual Positioning Subsystem and Laser Observation Subsystem (Step 2—Collaborative Positioning)

1. Inertial navigation system state derivation

Combined with the motion characteristics of the robot, we use the arc model to describe the motion of heading changes; the corresponding state derivation model is shown in Figure 4.

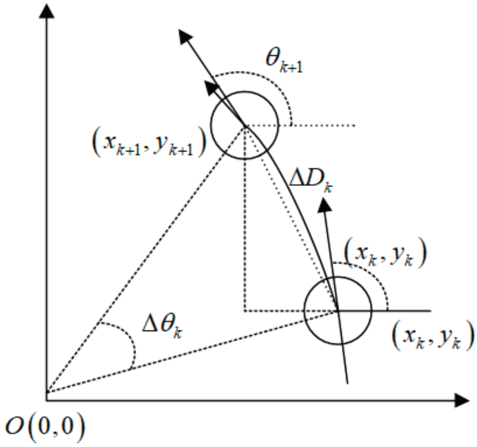


Figure 4. State derivation model for INS.

The motion model of the inertial guidance system can be expressed as

$$X_{k+1} = f(X_k, u_k) + \omega_k \tag{12}$$

$$f(X_k, u_k) = \begin{bmatrix} x_k + \frac{\Delta D_k}{\Delta\theta_k}(\sin(\theta_k + \Delta\theta_k) - \sin\theta_k) \\ y_k - \frac{\Delta D_k}{\Delta\theta_k}(\cos(\theta_k + \Delta\theta_k) - \cos\theta_k) \\ \theta_k + \Delta\theta_k \end{bmatrix}, |\Delta\theta_k| > 0 \tag{13}$$

where $X = [\text{x}, \text{y}, \theta]^T$, $u_k = [\Delta D_k, \delta\theta_{\text{k}},]^T$ is the input to the odometer model, D_k is the length of the arc, and θ_k is the change in steering angle; the values of the latter two are both obtained by fusing the odometer and gyroscope data. w_k is the Gaussian white noise of the system with covariance matrix Q. Then, the gained Jacobi matrix A of the state equation can be derived from the motion model as

$$A = \frac{\partial f}{\partial x} = \begin{bmatrix} 1 & 0 & \frac{\Delta D_k}{\Delta \theta_k}(\cos(\theta_k + \Delta\theta_k) - \cos\theta_k) \\ 0 & 1 & \frac{\Delta D_k}{\Delta \theta_k}(\sin(\theta_k + \Delta\theta_k) - \sin\theta_k) \\ 0 & 0 & 1 \end{bmatrix} \tag{14}$$

Assuming that its noise is Gaussian white noise, the covariance matrix P_{k+1} of the model can be derived as

$$P_{k+1} = \frac{\partial f}{\partial x} P_k \left(\frac{\partial f}{\partial x}\right)^T + E\left[w_k {w_k}^T\right] \tag{15}$$

2. Visual target detection and localization

The visual localization process involves using a deep-learning-based convolutional neural network algorithm for target detection. This algorithm detects the visual targets and extracts their center coordinates. The subsequent steps include camera calibration to establish the mapping relationship between the camera coordinate system and the world coordinate system, and correcting lens deformation errors. Finally, the algorithm calculates the distance from the center point to the camera's optical axis in the horizontal direction and the angle of deviation. This process enables accurate target localization. The complete visual localization algorithm process is depicted in Figure 5.

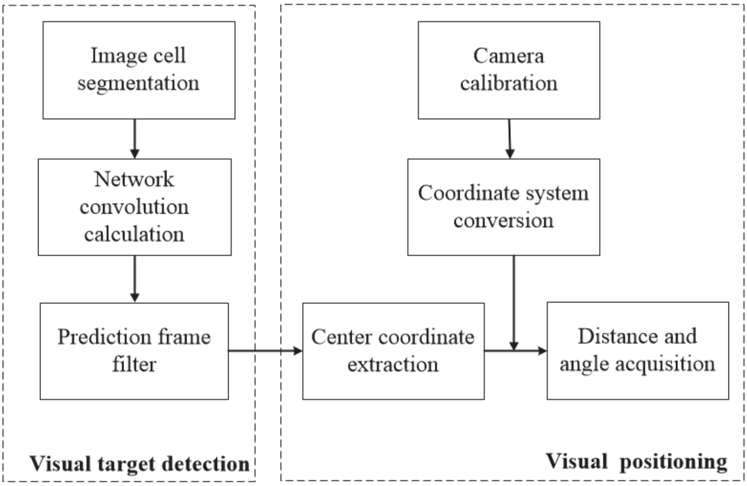


Image cell segmentation
Network convolution calculation
Prediction frame filter
Visual target detection
Camera calibration
Coordinate system conversion
Center coordinate extraction
Distance and angle acquisition
Visual positioning

Figure 5. Process of visual localization algorithm.

For robot target detection, we utilize an improved YOLO algorithm based on convolutional neural networks. This algorithm offers significant advantages in terms of precision, measurement speed, and accuracy [21], making it suitable for multi-robot indoor mobile scenarios. Unlike traditional two-stage target detection algorithms that rely on candidate regions, the YOLO algorithm treats target detection as a single regression process. It employs a single convolutional neural network to predict the detection frame and its corresponding position confidence level. The main architecture of the algorithm, illustrated in Figure 6, comprises an input layer, a backbone network, and a neck and head network. The specific working process is as follows:

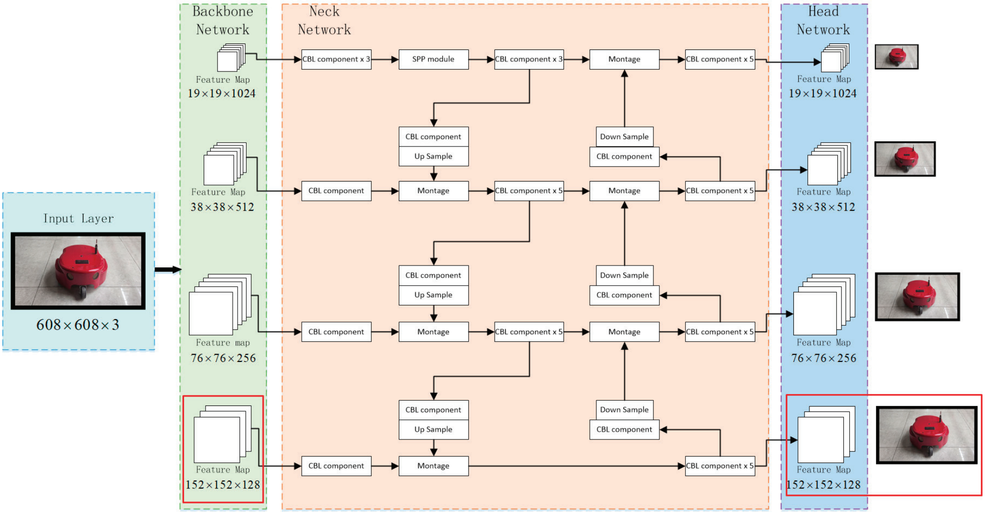


Figure 6. YOLO algorithm network architecture.

The input layer completes the pre-processing of the input avatar, scales the input image to a fixed size, and completes the normalization operation. The network input size of the YOLO algorithm can be adjusted to meet a multiple of 32. The larger the input image size is, the more details are retained and the better the detection effect is, but at the same time, the image detection speed will be reduced. Combined with the image characteristics of the robot target, the network input size is set to 608 × 608 in this paper.

The backbone network, comprising five Cross Stage Partial (CSP) modules, is utilized to extract deep semantic information from the image. These modules propagate features to the CBM (Convolution operation + BN batch normalization + Mish activation function) convolution module while also sending them to the residual block. The results of these two convolutions are then combined using the concat splicing unit [22]. This design reduces computational effort, enables richer gradient combinations, and avoids the problem of gradient disappearance in excessively deep networks. Each CSPX module contains $5 + 2 \times X$ convolutional layers, resulting in a total of 72 convolutional layers in the backbone network. When the input image size is 608 × 608, the operations performed by each CSP module, as well as the head CBM module, and the corresponding output results are depicted in Figure 4. The feature map changes from 608 to 304, 152, 76, 38, and 19, forming a three-layer feature map of sizes 76 × 76, 38 × 38, and 19 × 19, which constitute a feature pyramid.

The neck and head networks are utilized to extract fused features more effectively and accomplish multi-size target prediction, including the extraction of center coordinates. To enhance feature extraction and representation, the YOLO algorithm incorporates the PAN (Path Aggregation Network) structure after the Feature Pyramid Network (FPN) for feature fusion across different backbone layers. The neck network ultimately produces three standard-sized feature maps (76 × 76, 38 × 38, and 19 × 19) for multi-scale prediction by the head network. While this algorithm generally achieves good detection accuracy, there is a low probability of false detection in low-light environments or when encountering objects with similar shapes. To address this, we introduce an additional branch with a downsampling rate of 4 (resulting in a 152 × 152 feature map) to the original YOLO algorithm, which already includes downsampling rates of 32, 16, and 8 (corresponding to 19 × 19, 38 × 38, and 76 × 76 feature maps, respectively). This modification enables finer prediction results, enhances detection performance for smaller objects, improves the algorithm's spatial differentiation capability, and effectively reduces the probability of false detection [23].

Figure 7 showcases representative detection results achieved by the optimized YOLO target detection algorithm using the constructed dataset. The optimized algorithm demonstrates higher confidence in detecting target robots, road signs, and distinctive shape markers, meeting the accuracy requirements for visual localization.

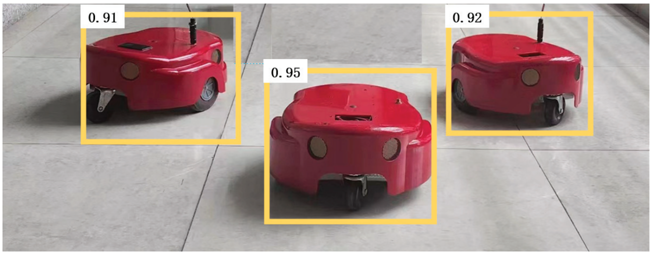


Figure 7. Target detection.

If the distance between the two cameras of the binocular vision sensor is known to be D, and θ_1, θ_2 are the angles between the binocular camera line and the camera-to-target center line, respectively, then the distance from the midpoint of the binocular camera to the target center can be solved as follows:

$$D_{OO'} = \frac{D_{C1C2}\sqrt{2\sin^2\theta_1 + 2\sin^2\theta_2 - \sin^2(\pi - \theta_1 - \theta_2)}}{2\sin(\pi - \theta_1 - \theta_2)} \tag{16}$$

If the main robot observes itself at moment k with coordinates (x_k, y_k), then its distance observation equation can be derived as

$$Z_j^d = H_j^d X_j + v_j^g = \sqrt{\left(x_k - x_j\right)^2 + \left(y_k - y_j\right)^2} + v_j^g \tag{17}$$

where $v_j{}^g$ is Gaussian white noise, and its Jacobi measurement matrix can be deduced as

$$H_j^d = \frac{\partial h}{\partial x} = \left[\frac{(x_j - x_k)}{\sqrt{(x_k - x_j)^2 + (y_k - y_j)^2}} \quad \frac{(y_j - y_k)}{\sqrt{(x_k - x_j)^2 + (y_k + y_j)^2}} \quad 0\right] \tag{18}$$

3. Laser observation and localization

If two colored cylinders are placed in the direction of the member robots' heading, the distance difference between the two cylinders is obtained by the visual target detection method, and the robot heading can be calculated at this moment. The laser rangefinder positioning heading model is shown in Figure 8.

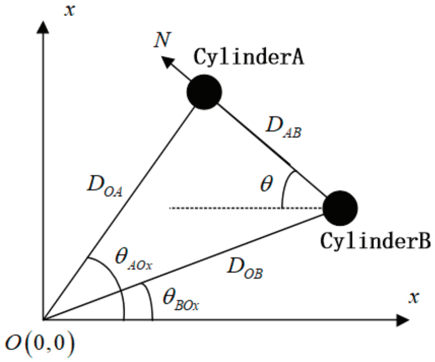


Figure 8. Laser rangefinder positioning heading model.

The forward direction of the robot can be determined by using different colored cylinders. D_{AB} is a known as a fixed distance; then, the robot heading can be deduced as

$$\theta = \arccos\left(\frac{{D_{AB}}^2 + {D_{OB}}^2 - {D_{OA}}^2}{2D_{AB}D_{OB}}\right) - \theta_{BOx} \tag{19}$$

The heading observation equation of the laser rangefinder can be expressed as

$$Z_j^b = H_j^b X_j + v_j^b = \arccos\left(\frac{{D_{AB}}^2 + {D_{OB}}^2 - {D_{OA}}^2}{2D_{AB}D_{OB}}\right) - \theta + v_j^b \tag{20}$$

where ${v_j}^b$ is Gaussian white noise.

The real-time distance of the member robot relative to the main robot can be obtained according to the laser rangefinder model and Equation (8). Let the coordinates of the main robot itself at the time of observation be (x_k, y_k); then, the distance equation of the laser rangefinder is

$$Z_j^d = H_j^d X_j + v_j^g = \sqrt{\left(x_k - x_f\right)^2 + \left(y_k - y_j\right)^2} + v_j^g \tag{21}$$

From the above equation, its measurement matrix can be further deduced as

$$H_j = \frac{\partial h}{\partial x} = \begin{bmatrix} H_j^d \\ H_j^b \end{bmatrix} = \begin{bmatrix} \frac{(x_j - x_k)}{\sqrt{(x_k - x_j)^2 + (y_k - y_j)^2}} & \frac{(y_j - y_k)}{\sqrt{(x_k - x_j)^2 + (y_k + y_j)^2}} & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{22}$$

The total observation equation of the laser sub-filter can be obtained as

$$Z_j = H_j X_j + v_j \tag{23}$$

2.3.3. Joint Filtering Algorithm (Step 3—Joint Filtering)

Referring to the joint filtering model depicted in Figure 1, the process begins with the discrete Kalman filter performing data fusion of the odometer and gyroscope at the front end to estimate the real-time state of the inertial guidance system. Then, the visual sensor extracts the angle information of the target, which is then used to establish laser distance data, facilitating interaction and cooperation among the subsystems. Finally, the joint filtering model employs federated filtering to accomplish information fusion within the distributed sensor network. Specifically, the fusion involves using the extended Kalman filter to merge the state of the inertial guidance system, serving as the reference state, with the visual positioning and laser positioning subsystems. The estimation results from the two extended Kalman sub-filters are subsequently passed to the main filter, enabling optimal fusion of information and estimation of the system's global state value $X = [x, y, \theta]^{\mathrm{T}}$. The joint filtering algorithm comprises the following steps:

1. Completing the solution of information allocation factors β_1, β_2. The information allocation factor β_i satisfies the information conservation principle, which means

$$\sum_{i=1}^{n} \beta_i + \beta_m = 1 \tag{24}$$

β_m represents the information distribution coefficient of the main filter, while the value of $\beta_1, \ldots, \beta_n$ directly impact the performance of joint filtering. Typically, larger coefficient values are assigned to sensors with higher measurement accuracy. However, since the external sensor subsystem employed in this study may experience sudden measurement errors like drift, using a fixed coefficient value is not appropriate. Hence, we employ eigenvalue decomposition based on the true variance P_i to calculate the real-time value of β_i. P_i can be decomposed by eigenvalues $P_i = L\Lambda_i L^T$, where $\Lambda_i = diag\{\lambda_{i1}\lambda_{i2}\ldots\lambda_{iN}\}$; the value of β_i can be deduced as

$$\beta_i = \frac{(tr\Lambda_i)^{-1}}{\sum\limits_{i=1}^{N}(tr\Lambda_i)^{-1} + (tr\Lambda_m)^{-1}} \tag{25}$$

2. Initialize the global state estimate value $\hat{X}_{i0}$ and the covariance matrix P_{i0} and assign the information to each sub-filter and the main filter in proportion to the information factor β_i:

$$P_i^{-1}(k) = P_g^{-1}(k)\beta_i \tag{26}$$

$$\hat{X}_i(k) = \hat{X}_g(k) \tag{27}$$

3. The time correction is applied simultaneously to each sub-filter and the main filter, and the common noise is allocated to each sub-filter based on the information factor β_i:

$$Q_i^{-1}(k) = Q_g^{-1}(k)\beta_i \tag{28}$$

then, the filtering time is corrected to

$$\hat{X}_i(k) = \hat{X}_g(\mathrm{k}) \tag{29}$$

$$P_i(k+1|k) = A(k+1,k)P_i(k)A^T(k+1,k) + \Gamma(k)Q_i(k)\Gamma^T(k) \tag{30}$$

where Γ is the system noise matrix and A is the state transfer matrix.

4. Each sub-filter uses a local observation Zi for the observed data correction.

$$P_i^{-1}(k+1|k+1) = P_i^{-1}(k+1|k) + H_i(k+1)R_i^{-1}(k+1)H_i^T(k+1) \tag{31}$$

$$P_i^{-1}(k+1)\hat{X}_i(k+1) = P_i^{-1}(k+1|k)\hat{X}_i(k+1|k) + H_i(k+1)R_i^{-1}(k+1)Z_i^T(k+1) \tag{32}$$

where H_i is measurement matrix.

5. Main filter completes information fusion [24].

$$P_g^{-1}(k+1) = \sum_{i=1}^{n} P_i^{-1}(k+1) \tag{33}$$

$$\hat{X}_\mathrm{g}(k+1) = P_\mathrm{g}(k+1)\sum_{i=1}^{n} P_i^{-1}(k+1)\hat{X}_\mathrm{i}(k+1) \tag{34}$$

where n is the amount of sub-filters.

3. Analysis and Correction of Practical Errors

The theoretical analysis presented above discusses the feasibility of cooperative localization and outlines its implementation process. However, in practical applications of multi-robot formation navigation, errors are inevitable. Therefore, it is crucial to analyze the causes of different types of errors and propose appropriate correction solutions.

3.1. Visual Target Detection Delay

The visual target detection algorithm studied in this paper consists of the following two main processes:

Firstly, the main robot observes manual road signs for self-positioning in the global coordinate system. The main robot uses target detection and extracts road signs from the environment to obtain the relative distance information between the camera and the road signs, which is fused with information from its INS to obtain accurate global positioning information. Secondly, the master robot observes the member robots and positions them for observation in a relative coordinate system. The main robot uses binocular vision sensors and laser rangefinders to locate the member robots and sends the position information to the member robots through the mesh network. The member robots use this information to fuse with their own inertial guidance systems to obtain accurate relative position information.

The visual target detection information is fused with the information from the inertial guidance system. However, this fusion process introduces errors due to the delay in target detection. The error can be managed through two approaches: first, by restoring the motion of the observed robot to its position prior to the time delay, and second, by reducing the target detection time to fall within the acceptable error range. Table 1 illustrates the time delay observed in each stage of the target detection process.

Table 1. Time consumption of each stage of visual target detection.

Sub-Processes	Time (ms)	Instruction
Camera imaging	≤ 5	Imaging of the target object, pre-processing, scaling, and normalization of the input image.
Target detection	≤ 25, ≥ 20	Feature-pyramid-based target prediction using a convolutional-neural-network-trained classifier.
Laser Positioning	≤ 5	This process can be performed in parallel with the above process.

The time required for the target detection process can be reduced to less than 15 ms by significantly reducing the image area to be inspected in indoor scenes. In the fusion strategy, no action is taken during the first imaging, and the data from the second imaging are used as observation data for the inertial guidance system. Assuming the robot moves at a speed of 600 mm/s, the observation error range caused by the target detection algorithm can be controlled to be below the millisecond level, thus meeting the localization accuracy requirements once again.

3.2. Network Communication Latency

The master robot transmits the observation and positioning information of the member robots to them through the mesh network. The member robots then integrate this information with their own inertial guidance systems to obtain precise real-time relative position information. In practical applications, there is a network time delay during communication. However, since the robots form a dynamic mobile network, the communication time delay is minimal, and the differences in time delay mainly depend on the real-time network conditions. To manage these errors, motion recovery is performed on the robots.

Figure 9 illustrates the initial assumption of the robot's circular motion with a radius R around the origin O.

In the figure, (x, y) represents the actual position of the robot, while (x', y') denotes the estimated position observed when the member robot reaches the position (x_{k1}, y_{k1}). In the absence of network time delay Δt, the estimated position is corrected to $(x_{k1'}, y_{k1'})$ after filtering. However, due to the presence of time delay Δt, the actual position of the robot reaches (x_{k2}, y_{k2}) before the observations from the subsystem reach the member robots for fusion. If these data are fused with the inertial guidance system data, errors will occur. Assuming the sensors within the inertial guidance system have undergone internal fusion n times within Δt time, the sequence of state estimates is preserved $\{X_1, X_2, \ldots, X_m\}$, where $X = \{x, y, \theta\}$ and m > n. After receiving the observation from the network, the member robot calculates the network time delay Δt, retrieves the state estimate from the inertial guidance system's state estimation sequence at the corresponding time point for fusion filtering, and the filtered state estimation point should be $(x_{k1'}, y_{k1'})$. Then, a state sequence is established, representing the correlation chain of the robot's internal sensor adjustment angle. By fitting a trajectory that progresses from the point $(x_{k1'}, y_{k1'})$, the corrected position $(x_{\mathrm{k}2'}, y_{k2'})$ that the robot should adopt at that moment is obtained.

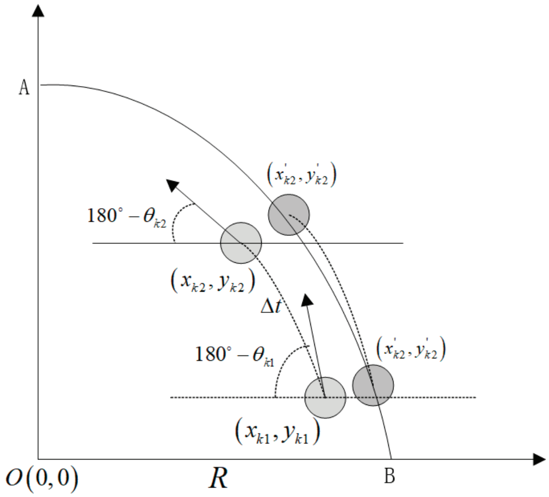


Figure 9. Communication latency recovery model.

3.3. Observation Masking

Both visual imaging and laser ranging rely on line-of-sight observations. If the camera–target line is obstructed by other objects, the target cannot be detected and positioning information cannot be obtained. In multi-robot formation movement, the main robot and member robots may be occasionally obstructed by other robots or obstacles. Figure 10 illustrates a scenario where the member robots are obscured by each other. In the figure, R1 represents the main robot; *R*2 and *R*3 represent the member robots; and N_1 and N_2 indicate the headings of *R*1 and *R*2, respectively. When all three robots align in a straight line, as shown in the figure, *R*1 and *R*3 are obstructed by *R*2, rendering *R*3 unobservable by *R*1.

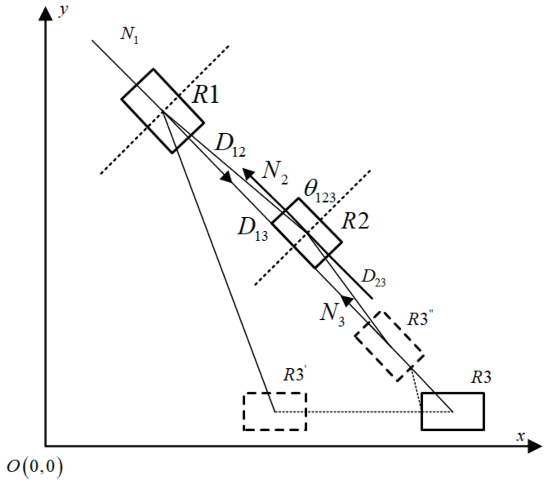


Figure 10. Robotic observation masking model.

This problem can be addressed by considering two cases. The first case occurs when *R*3 moves to its next position *R*3′, implying that *R*2′s obstruction of *R*3 is temporary.

In this situation, $R2$ only relies on its own inertial guidance system to track its motion trajectory during the period of obstruction. Once $R2$ passes the obscured area, it can then use observation information from the main robot to make corrections. The second case occurs when $R3$ maintains the same heading as $R2$ and the three robots move in a straight line for an extended period. In this scenario, $R1$ is unable to observe $R3$ for an extended duration; so, we utilize R3′s own sonar array positioning to make position corrections. The distance D_{12} between $R1$ and $R2$ in the figure can be measured because $R2$ is in the heading direction of $R3$, and its sonar array emits waves in this direction with a certain level of accuracy. Additionally, the small angle formed by the heading direction of $R3$, where $R2$ is located, can be determined through the geometric relationship between global positioning, allowing us to measure D_{23}. Therefore, the distance between $R3$ and the main robot can be deduced as

$$D_{13} = \sqrt{D_{12}^2 + D_{23}^2 - 2D_{12}D_{23}\cos\theta_{123}} \tag{35}$$

3.4. Requirements of the Filtering Process for the Observed Values

The equation for the observation of the main robot to the member robots in relative coordinates is

$$Z_j^d = H_j^d X_j + v_j^g = \sqrt{\left(x_k - x_j\right)^2 + \left(y_k - y_j\right)^2} + v_j^g \tag{36}$$

The observation equation converted to relative orientation is

$$\varphi = \arctan\left(\frac{y_j - y_k}{x_j - x_k}\right) - \theta_j + v_\varphi \tag{37}$$

where θ_j is the current heading of the main robot, and v_φ is the Gaussian white noise with zero mean and variance of ${\delta_\varphi}^2$, when

$$\arctan\left(\frac{y_f - y_k}{x_j - x_k}\right) \to +\frac{\pi}{2}, \frac{\Delta y}{\Delta x} = \frac{y_j - y_k}{x_j - x_k} \to +\infty \tag{38}$$

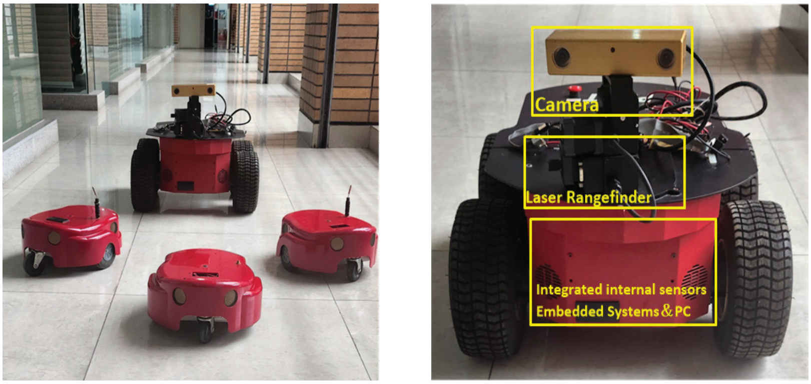
$$\arctan\left(\frac{y_j - y_k}{x_j - x_k}\right) \to -\frac{\pi}{2}, \frac{\Delta y}{\Delta x} = \frac{y_j - y_k}{x_j - x_k} \to -\infty \tag{39}$$

This implies that the Δx between the two robots is very small, and minor measurement errors can lead to significant deviations in the orientation observation, thereby causing challenges in filter convergence [25]. Therefore, when the relative orientation is near ±(π/2), the relative orientation observation does not provide useful localization information. Hence, prior to the filtering update, the observation is evaluated and selected based on the 3δφ law. If the difference between the observation and the expected value exceeds 3δφ, the observation is discarded. Subsequently, if the observation satisfies the condition $\left|Z_j(k+1)\right| \in \left(\frac{\pi}{2} - \theta_t, \frac{\pi}{2} + \theta_t\right)$, it is determined to fall within the range of ±(π/2) and cannot be used for the filtering update. However, the state update of the inertial guidance system is still employed.

4. Results and Discussion

4.1. Test Platform

To evaluate the performance of the multi-robot cooperative localization algorithm in an open indoor environment, we designed a multi-robot experimental platform consisting of a main robot and three smaller robots, as shown in Figure 11a.



(**a**) (**b**)

Figure 11. Test platform and equipment. (**a**) Multi-robot system formation scenario. (**b**) Main robot and main sensors.

As shown in Figure 11b, the main robot is equipped with an onboard computer that possesses processing and computing capabilities. It can be connected to a WiFi device with a USB interface to facilitate communication among the multi-robot systems. The member robots are equipped with microcontrollers responsible for motion system state processing. In the actual experimental setup, the multi-robot platform utilizes a SICK laser range scanner for distance measurement, and member robots are equipped with sonar arrays as their distance sensors. The main robot is equipped with a rotating binocular vision sensor, enabling the target image to be positioned as close to the center of the field of view as possible, thus improving positioning accuracy. All experiments are conducted and performed on this platform, with different experimental scenarios varying in the combinations of robots and sensors. The configuration program of the multi-robot system's experimental platform equipment is shown in Table 2.

Table 2. Multi-robot experimental platform equipment configuration program.

Serial Number	Equipment Name	Main Sensors and Information Processing Units	Robot Function Description
1	Main Robot (Pioneer 3-AT)	• Integrated Odometer • Integrated Gyroscope • SICK Laser Rangefinder • Binocular Vision Sensor • In-car Calculator (1.44G main frequency, 4G memory, 64G hard disk) • WiFi devices	• Maximum movement speed, 3 m/s • Four-wheel construction with four motors with encoders
2	Member Robot (AmigoBot)	• Integrated Odometer • Integrated Gyroscope • Integrated Sonar Array (10) • microcontroller (complete motion information processing) • WiFi devices	• Maximum movement speed, 2 m/s • Maximum weight, 1 kg • Four-wheel structure—two driving wheels, two balance wheels

4.2. Test Scenarios

Three field tests were designed and conducted to assess the performance of the proposed scheme and analyze error characteristics. Test Case 1 aims to evaluate the fusion of odometry and gyroscope information using discrete Kalman filtering. Test Cases 2 and 3 are implemented in a multi-robot system performing formation marching, as illustrated in

Figure 12, where the black circle represents the main robot and the white circles represents the member robots.

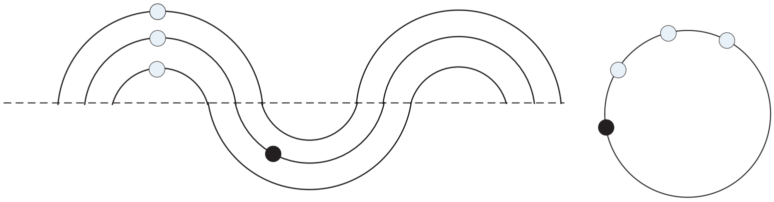


Figure 12. Positioning experimental trajectory of Test Cases 2 (the left part) and 3 (the right part).

Test Case 2 examines the performance of the joint filtering localization algorithm during S-curve motion, where the master robot leads the member robots in formation. Test Case 3 investigates the actual localization error in the presence of observation occlusion when the multi-robot system moves in a circular trajectory. The deviation is calculated by comparing the actual robot motion trajectory with a specified high-precision reference trajectory.

All robots are equipped with WiFi communication devices that support IEEE802.11 series protocols, enabling the formation of a dynamic mesh network for data communication among them. The specific experimental case scenario is presented in Table 3.

Table 3. Summary of the Test Cases.

Test Design	Test Case 1	Test Case 2	Test Case 3
Test scenario description	Circular motion of single robot	The main robot leads the members in an S-curve movement in formation according to a fixed relative position	The member robots follow the main robot in a circular motion along the same trajectory formation
Environment	Empty indoor environment with member robots moving in formation with the main robot along a designated trajectory		
Fusion domain	Positioning domain with open indoor environments Robot movement speed $\leq$ 800 mm/s		
Total Time Duration	50 s	120 s	120 s
Competing Solutions	Odo only Odo/Gyroscope	Odo/Gyroscope Joint filtering algorithm (Proposed)	Odo/Gyroscope Joint filtering algorithm (Proposed, presence of observation masking)
Reference	The reference trajectory in the experiment is obtained from a set of high-precision combined navigation equipment H Guide N580 and post-processed by Novatel commercial software IE 8.9.		

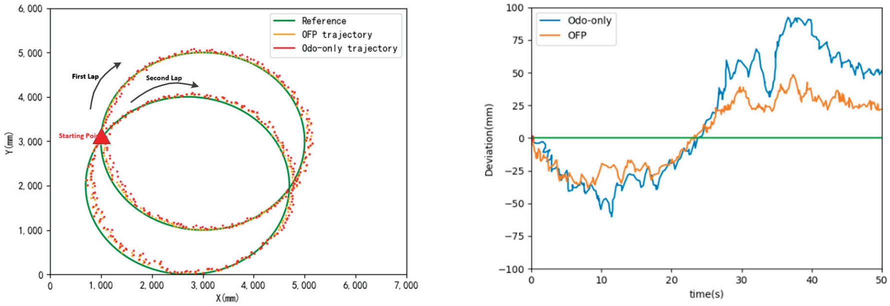
4.3. Test Results and Discussion

4.3.1. Odometer and Gyroscope Information Fusion

Figure 13 illustrates a graph comparing the data fusion positioning (OFP) using both the odometer and gyroscope with the trajectory deviation of using only the single sensor odometer in the experimental scenario Test Case 1. The results demonstrate that the fused trajectory exhibits greater proximity to the real trajectory compared to that when relying solely on the odometer navigation, resulting in an accuracy improvement of 43.3%.

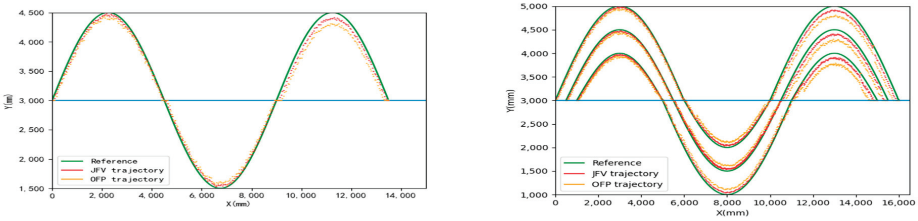
4.3.2. Joint Filter Positioning

Figure 14 presents the actual experimental results comparing the trajectories of odometer fusion positioning (OFP) and joint filter model fusion positioning (JFV) in Test Case 2. The figure contains three sets of curves corresponding to the three member robots. In Figure 14a, the mean deviation of the navigation trajectory using odometer fusion is 11.3 mm. Over time, the deviation of the RI developed trajectory shows a significant increasing trend due to cumulative errors in the internal sensors. In contrast, the joint-filtered navigation profile closely aligns with the reference trajectory without significant cumulative errors. Figure 14b showcases the trajectory comparison between the main robot's odometer fusion localization (OFP) and joint-filtered model fusion localization (JFV). The JFV curve effectively mitigates the cumulative error observed in OFP.



(**a**) (**b**)

Figure 13. Comparison of track deviation before and after odometry fusion. (**a**) Comparison trajectories before and after fusion. (**b**) Calculation of bias before and after fusion.



(**a**) (**b**)

Figure 14. Multi-robot system trajectory chart. (**a**) Trajectory of three member robots. (**b**) Trajectory of mail robot.

Figure 15 displays the peak curves derived from the trajectory deviations calculated during positioning experiments for both the main robot and the member robots. The JFV trajectory of the main robot exhibits greater accuracy compared with that of the member robots. This can be attributed to the main robot's self-positioning in the global coordinate system, which experiences fewer observation errors. On the other hand, the global positioning of the member robots incorporates the observation errors of the main robot.

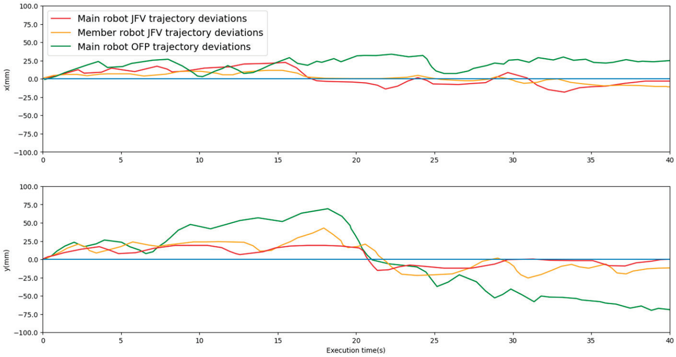


Figure 15. Comparison of positioning error in Test Case 2.

4.3.3. Observation Occlusion Error Handling

The previous experimental procedure considered three error factors: target detection delay, network delay, and filter requirements for observation orientation. However, no treatment for masking observation errors has been applied. In this section, we use the experimental scenario of Test Case 3 to evaluate the masking error in observations. Theorem-type environments (including propositions, lemmas, corollaries, etc.) can be formatted as follows:

Figure 16 illustrates a scenario where the robot follows a small arc trajectory. *R*1 represents the main robot, while *R*2 and *R*3 are member robots. Throughout the motion, *R*3 remains obscured by *R*2 and cannot be observed by *R*1. The accuracy of *R*2, which relies on vision and laser navigation, is higher compared with that of *R*3, which relies on sonar array ranging data.

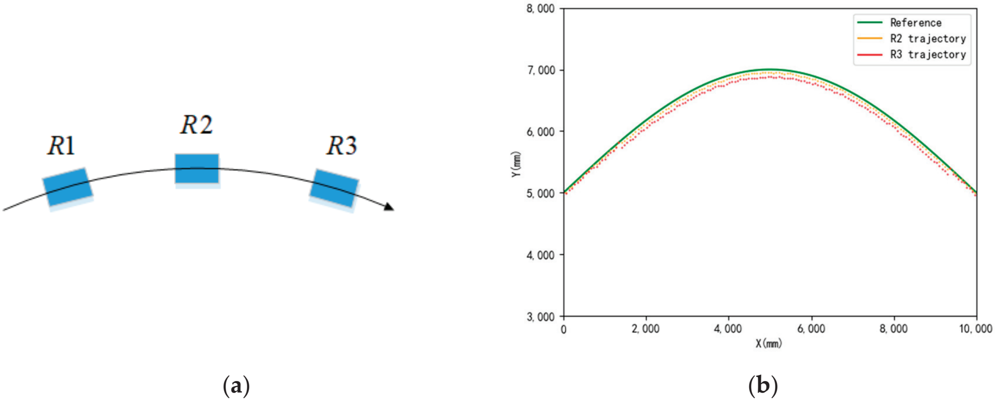


Figure 16. Observation occlusion error handling. (**a**) Observation occlusion scene. (**b**) Track error handling.

5. Conclusions

This paper introduces a cooperative localization method based on a joint filtering model for practical multi-robot formation navigation. The method integrates sensor data from various sources to achieve precise localization. The model incorporates observation information from other robots, enhancing localization accuracy through cooperative efforts. The fusion of a laser rangefinder and vision sensor enables unique laser-ranging-based positioning by obtaining the target's horizontal projection and the camera's optical axis angle. Using a discrete Kalman filter to merge internal sensor data demonstrates superior real-time performance. The application of a visual target detection algorithm based on a convolutional neural network, along with visual and laser localization, employs a joint filtering model to fuse multiple sensors as a sub-filtering system. The resulting estimated deviation of robot position has a root mean square value of 22.8 mm in the x direction and 24.6 mm in the y direction, comparable to current research results based on collaborative autonomous SLAM positioning algorithms for multiple mobile robots that meet high accuracy requirements [26]. Moreover, it surpasses the co-localization errors of existing systems with multiple robot clusters in different formations. Lastly, an in-depth analysis of error sources is conducted in the actual formation application of a multi-robot system [27]. Experimental results demonstrate that the proposed positioning method achieves precise positioning with a small error range and negligible cumulative errors, making it suitable for real-world robot formation applications.

In our future work, we will engage in research in four challenging directions, including the improvement of navigation performance of multi-robot systems, formation control, and cooperative communication.

(1) Leveraging current technology [28], we aim to enhance and optimize our scheme. This involves addressing observation errors arising from the information fusion process and tackling time-delay errors in diverse formation configurations, varying robot numbers, and investigating different communication conditions. This challenging aspect contributes to the refinement of the existing architectural scheme.
(2) We will examine the impact of applying the existing algorithms in height-transformed environments and develop 3D positioning information fusion schemes accordingly, which should enable a more accurate estimation of trip energy consumption to support some advanced future applications.
(3) Based on the accurate position information of member robots and obstacles, the autonomous mobile robot group formation planning, holding, and control algorithms will be further investigated in combination with the currently available algorithmic models [29,30], and the optimal efficiency solution will be proposed based on different task scenarios.
(4) It is necessary to conduct research on the interaction strategies of multi-robot systems under specific topologies and to design transmission strategies for the information in the interaction network, such as sensor data, communication protocols, control information, and images, which will meet the requirements of formation missions with high communication performance.

Author Contributions: Z.C. and J.L. conceived and designed the algorithm; Z.C. and W.C. performed the experiments; W.C. and B.Z. analyzed the data; and Z.C. and J.L. wrote the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the sponsorship of the National Natural Science Foundation of China under Grant 42274051, in part by the Open foundation of Key Laboratory of Maritime Intelligent Cyberspace Technology, Ministry of Education under Grant MICT202306.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors thank the Ministry of Science and Technology (MOST) for its financial support. We also thank the editor and anonymous reviewers for their constructive comments on this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lupton, T.; Sukkarieh, S. Visual-Inertial-Aided Navigation for High-Dynamic Motion in Built Environments Without Initial Conditions. *IEEE Trans. Robot.* **2011**, *28*, 61–76. [CrossRef]
2. Kelly, J.; Sukhatme, G.S. Visual-Inertial Sensor Fusion: Localization, Mapping and Sensor-to-Sensor Self-calibration. *Int. J. Robot. Res.* **2010**, *30*, 56–79. [CrossRef]
3. Jones, E.S.; Soatto, S. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *Int. J. Robot. Res.* **2011**, *30*, 407–430. [CrossRef]
4. Shen, S. *Autonomous Navigation in Complex Indoor and Outdoor Environments with Micro Aerial Vehicles*; University of Pennsylvania ProQuest Dissertations Publishing: Philadelphia, PA, USA, 2014.
5. Lu, Y.; Lee, J.; Yeh, S.H.; Cheng, H.M.; Chen, B.; Song, D. Sharing Heterogeneous Spatial Knowledge: Map Fusion Between Asynchronous Monocular Vision and Lidar or Other Prior Inputs. *Robot. Res.* **2019**, *10*, 727–741.
6. Howard, A.; Matarić, M.J.; Sukhatme, G.S. Localization for Mobile Robot Teams: A Distributed MLE Approach. *Exp. Robot. VIII* **2003**, *5*, 146–155.
7. Sun, R.; Yang, Y.; Chiang, K.-W.; Duong, T.-T.; Lin, K.-Y.; Tsai, G.-J. Robust IMU/GPS/VO Integration for Vehicle Navigation in GNSS Degraded Urban Areas. *IEEE Sens. J.* **2020**, *20*, 10110–10122. [CrossRef]
8. Anbu, N.A.; Jayaprasanth, D. Integration of Inertial Navigation System with Global Positioning System using Extended Kalman Filter. In Proceedings of the International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 27–29 November 2019; pp. 789–794.
9. Mourikis, A.I.; Roumeliotis, S.I. A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation. In Proceedings of the International Conference on Robotics and Automation, Rome, Italy, 10–14 April 2007; pp. 3565–3572.
10. Dou, L.; Li, M.; Li, Y.; Zhao, Q.Y.; Li, J.; Wang, Z. A novel artificial bee colony optimization algorithm for global path planning of multi-robot systems. In Proceedings of the IEEE International Conference on Robotics and Biomimetics, Bali, Indonesia, 5–10 December 2014.

11. Howard, A.; Matark, M.J.; Sukhatme, G.S. Localization for mobile robot teams using maximum likelihood estimation. In Proceedings of the IEEE International Workshop on Intelligent Robots and Systems (IROS), Lausanne, Switzerland, 30 September–4 October 2002.
12. Chhatpar, S.R.; Branicky, M.S. Particle filtering for localization in robotic assemblies with position uncertainty. In Proceedings of the IEEE International Workshop on Intelligent Robots and Systems (IROS), Edmonton, AB, Canada, 2–6 August 2005.
13. Burgard, W.; Cremers, A.B.; Fox, D.; Hähnel, D.; Lakemeyer, G.; Schulz, D.; Steiner, W.; Thrun, S. Experiences with an interactive museum tour-guide robot. *Artif. Intell.* **1999**, *114*, 3–55. [CrossRef]
14. Fox, D.; Burgard, W.; Kruppa, H.; Thrun, S. A Probabilistic Approach to Collaborative Multi-Robot Localization. *Auton. Robots* **2000**, *8*, 325–344. [CrossRef]
15. Rekleitis, I.M.; Dudek, G.; Milios, E.E. Multi-robot cooperative localization: A study of trade-offs between efficiency and accuracy. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 30 September–4 October 2002.
16. Ning, B.; Han, Q.L.; Zuo, Z.; Jin, J.; Zheng, J. Collective Behaviors of Mobile Robots Beyond the Nearest Neighbor Rules With Switching Topology. *IEEE Trans. Cybern.* **2017**, *48*, 1577–1590. [CrossRef] [PubMed]
17. Shucker, B.; Murphey, T.; Bennett, J.K. A method of cooperative control using occasional non-local interactions. In Proceedings of the 2006 IEEE International Conference on Robotics and Automation (IRCA), Orlando, FL, USA, 15–19 May 2006.
18. Garcia, M.A.; Solanas, A. 3D simultaneous localization and modeling from stereo vision. In Proceedings of the IEEE International Conference on Robotics and Automation (IRCA), New Orleans, LA, USA, 26 April–1 May 2004.
19. Kannala, J.; Heikkilä, J.; Brandt, S.S. Geometric Camera Calibration. *Wiley Encycl. Comput. Sci. Eng.* **2008**, *13*, 1–20.
20. Zong, G.; Deng, L.; Wang, W. Robust Localization algorithms for outdoor mobile robot. *J. Beijing Univ. Aeronaut. Astronaut.* **2007**, *33*, 454–458.
21. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y. Scaled-YOLOv4: Scaling Cross Stage Partial Network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13029–13038.
22. Wang, C.Y.; Liao, H.Y.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A New Backbone That Can Enhance Learning Capability of CNN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Seattle, WA, USA, 13–19 June 2020; pp. 390–391.
23. Hu, G.X.; Yang, Z.; Hu, L.; Huang, L.; Han, J.M. Small Object Detection with Multiscale Features. *Int. J. Digit. Multimed. Broadcast.* **2018**, *2018*, 4546896. [CrossRef]
24. Jilkov, V.P.; Angelova, D.S.; Semerdjiev, T.A. Design and comparison of mode-set adaptive IMM algorithms for maneuvering target tracking. *IEEE Trans. Aerosp. Elecreonis Syst.* **1999**, *35*, 343–350. [CrossRef]
25. Wang, L.; Liu, Y.H.; Wan, J.W.; Shao, J.X. Multi-Robot Cooperative Localization Based on Relative Bearing. *Chin. J. Sens. Actuators* **2007**, *20*, 794–799.
26. Wang, J. Multi-Robot Collaborative SLAM Algorithm Research. Master's Thesis, Xi'an University of Technology, Xi'an, China, 2023; pp. 47–63.
27. Zhu, K.; Wen, Z.; Di, S.; Zhang, F.; Guo, G.; Kang, H. Observability and Co-Positioning Accuracy Analysis of Multi-Robot Systems. *Telecommun. Eng.* **2023**, *51*, 4–8.
28. Chiang, K.W.; Duong, T.T.; Liao, J.K. The Performance Analysis of a Real-Time Integrated INS/GPS Vehicle Navigation System with Abnormal GPS Measurement Elimination. *Sensors* **2013**, *13*, 10599–10622. [CrossRef] [PubMed]
29. Yang, S.; Li, T.; Shi, Q.; Bai, W.; Wu, Y. Artificial Potential-Based Formation Control with Collision and Obstacle Avoidance for Second-order Multi-Agent Systems. In Proceedings of the 2020 7th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS), Guangzhou, China, 13–15 November 2021.
30. Lu, Q.; Han, Q.L.; Zhang, B.; Liu, D.; Liu, S. Cooperative Control of Mobile Sensor Networks for Environmental Monitoring: An Event-Triggered Finite-Time Control Scheme. *IEEE Trans. Cybern.* **2016**, *47*, 4134–4147. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Communication

Advancing Ultra-High Precision in Satellite–Ground Time–Frequency Comparison: Ground-Based Experiment and Simulation Verification for the China Space Station

Yanming Guo [1,2,†], Shuaihe Gao [1,*,†], Zhibing Pan [1,†], Pei Wang [1], Xuewen Gong [1], Jiangyu Chen [3], Kun Song [3], Zhen Zhong [3], Yaoli Yue [4], Lishu Guo [1], Yan Bai [1], Yuping Gao [1], Xiaochun Lu [1,2] and Shougang Zhang [1,2]

[1] National Time Service Center, Chinese Academy of Sciences, Xi'an 710600, China; guoyanming@ntsc.ac.cn (Y.G.); panzhibing@ntsc.ac.cn (Z.P.); wangpei@ntsc.ac.cn (P.W.); gongxuewen@ntsc.ac.cn (X.G.); guolishu@ntsc.ac.cn (L.G.); by@ntsc.ac.cn (Y.B.); gaoyp@ntsc.ac.cn (Y.G.); luxc@ntsc.ac.cn (X.L.); szhang@ntsc.ac.cn (S.Z.)

[2] University of Chinese Academy of Sciences, Beijing 100039, China

[3] The 29th Research Institute of China Electronics Technology Group Corporation, Chengdu 611731, China; chjiangy@sina.com (J.C.); songkun1122@foxmail.com (K.S.); 15608172230@163.com (Z.Z.)

[4] The 34th Research Institute of China Electronics Technology Group Corporation, Guilin 541010, China; 13635199784@163.com

* Correspondence: gaoshuaihe@ntsc.ac.cn

† These authors contributed equally to this work.

Abstract: Establishing an ultra-high-precision link for time–frequency comparisons between satellites and ground stations is critically important. This endeavor is fundamental to the advancement of pioneering space science exploration and the development of a robust space-based time–frequency system featuring ultra-high-precision space atomic clocks. In response to the requirements for assessing the long-term stability of high-precision space atomic clocks, we have designed and implemented a satellite–ground microwave time–frequency comparison system and method based on a three-frequency mode. Ground-based experimental results demonstrate that the equipment layer can achieve a satellite–ground time comparison accuracy better than 0.4 ps (RMS), with the equipment delay stability (ADEV) for all three frequencies being better than 8×10^{-18} at 86,400. By leveraging the ground-based experimental results, we constructed a satellite–ground time–frequency comparison simulation and verification platform. This platform realizes ultra-high-precision satellite–ground time–frequency comparison based on the China Space Station (CSS). After correcting various transmission delay errors, the satellite–ground time comparison achieved an accuracy better than 0.8 ps and an ADEV better than 2×10^{-17} at 86,400. This validation of our novel satellite–ground time–frequency comparison system and method, capable of achieving an 10^{-17} magnitude stability, is not only a significant contribution to the field of space time–frequency systems but also paves the way for future advancements and applications in space science exploration.

Keywords: satellite–ground microwave time–frequency comparison; three-frequency mode; China Space Station (CSS); transmission delay errors; stability

Citation: Guo, Y.; Gao, S.; Pan, Z.; Wang, P.; Gong, X.; Chen, J.; Song, K.; Zhong, Z.; Yue, Y.; Guo, L.; et al. Advancing Ultra-High Precision in Satellite–Ground Time–Frequency Comparison: Ground-Based Experiment and Simulation Verification for the China Space Station. *Remote Sens.* **2023**, *15*, 5393. https://doi.org/10.3390/rs15225393

Academic Editor: Michael E. Gorbunov

Received: 19 October 2023
Revised: 8 November 2023
Accepted: 14 November 2023
Published: 17 November 2023

1. Introduction

Globally, the development of high-precision space atomic clocks is regarded as a vital direction in the exploration of cutting-edge space science. These sophisticated timekeepers are capable not only of maintaining precise time on Earth but also of facilitating precise time–frequency comparisons at any location in space, which is particularly crucial for fundamental physics research. The capability for spatial time–frequency comparison enables the detection of spacetime variations in different gravitational fields, playing a pivotal role in advanced fundamental physical research such as the measurement of gravitational redshift and the determination of the fine-structure constant. Current GNSS systems such as BDS, Galileo, and GPS III achieve 10^{-14} to 10^{-15} stability at one day while offering enhanced time and frequency transfer via commercial services, while the Two-Way Satellite Time and Frequency Transfer (TWSTFT) method employs two-way satellite signals to compare terrestrial clocks, achieving stability of the order of 10^{-15} by leveraging the symmetry of the two-way transmission to enhance measurement stability. Led by the European Space Agency, the European Advanced Atomic Clock Program (Atomic Clock Ensemble in Space, ACES) plans to equip the International Space Station (ISS) with a combination of cold atomic microwave clocks and active hydrogen maser clocks, and the ACES MWL system is aimed at achieving a satellite–ground comparison capability with an uncertainty and daily stability reaching the 10^{-16} level. Furthermore, this will pave the way for advanced experiments in cold atomic physics and relativity tests [1]. In comparison with the ACES initiative, the high-precision time–frequency scientific experiment system of the China Space Station (CSS) intends to deploy and operate an ultra-high-precision atomic clock combination, including cold atomic strontium optical clocks, cold atomic microwave clocks, and active hydrogen maser clocks. Simultaneously, by leveraging both microwave and laser transmission links, the CSS aspires to achieve ultra-high-precision comparisons between space- and ground-based clocks, enhancing long-term stability [2]. Additionally, the European Space Agency has laid out its plans for the development of space optical clocks (SOC). In the project's second phase, they have realized a space optical clock prototype with a long-term stability of 5×10^{-17}. Plans are in place to install this on the ISS [3]. Hence, establishing a high-precision satellite–ground time–frequency comparison link, objectively assessing the performance of space atomic clock groups, and promoting the application of high-precision time–frequency references are important research directions.

In the realm of scientific experimentation, optical techniques, including frequency combs and continuous-wave lasers, have shown remarkable progress, and measurement techniques based on free-space optical frequency comb signal comparisons have garnered significant attention and development [4–7], with early systems achieving 10^{-16} instability in one second and recent advancements pushing this to 3×10^{-18} [8]. Dual-branch comb designs have yielded Allan deviations of approximately 2×10^{-18} over one second and an ultimate stability of approximately 5×10^{-21} over 100,000 s after noise reduction [9]. Although lab-based free-space optical frequency comb techniques have shown high accuracy, their performance is greatly diminished in atmospheric conditions, making long-distance clock comparisons challenging. Therefore, the laser satellite–ground time–frequency transmission, implemented using the optical frequency comb, inherently exhibits vulnerabilities. It is severely disrupted by atmospheric conditions that compromise its continuous operational capability. Moreover, because of the high-speed relative motion between satellites and the Earth, the simplistic relativistic correction models currently in use substantially limit the precision of correcting time delay errors caused by relative positional changes of satellite payloads [10]. Consequently, the application of lasers to achieve space optical frequency comb signal comparisons in the complex satellite–ground environment may be constrained. There is the potential that it might not meet the demands for ultra-high precision in satellite–ground time–frequency comparisons. This underscores the importance of improving microwave time–frequency comparison methods, which may better withstand atmospheric disturbances and maintain a stable long-term space–ground link. Advancements in microwave time–frequency comparison methods may hold the key to realizing a future global network of ultra-high-precision clock comparisons.

In response to the requirements for evaluating the long-term stability of satellite–ground time–frequency in the high-precision time–frequency experimental system of the China Space Station (CSS), this paper focuses on the microwave time–frequency transmission payload planned to be on board the space station. In collaboration with a ground station (GS) system, we have constructed a simulated satellite–ground time–frequency comparison system. Utilizing this system, we validated the design of a terrestrial three-frequency mode (1 uplink + 2 downlink) for time–frequency comparisons. Through system architecture, error modeling and correction, and equipment control, we demonstrate the feasibility of achieving ps-level satellite–ground time–frequency comparison. Additionally, we show that long-term continuous operation can attain stability of the order of 10^{-17} for satellite–ground time–frequency comparison.

2. Satellite (CSS)–Ground Time–Frequency Comparison System and Methodology

The CSS is equipped with an array of time–frequency payload equipment, which includes advanced onboard atomic clocks—such as a hydrogen maser, a microwave clock, and an optical clock—integral for maintaining the station's time standards with ultra-high precision. Alongside these clocks, the CSS hosts external payloads like the microwave link [11] payload, laser link payload, and precise orbit determination antenna payload. These payloads, in combination with the ground station equipment, form a comprehensive satellite–ground time–frequency comparison system [12]. This system, as depicted in Figure 1, is designed to facilitate ultra-high-precision time–frequency comparisons between the satellite and ground, enabling critical operations and experiments that require stringent time synchronization and frequency standards.

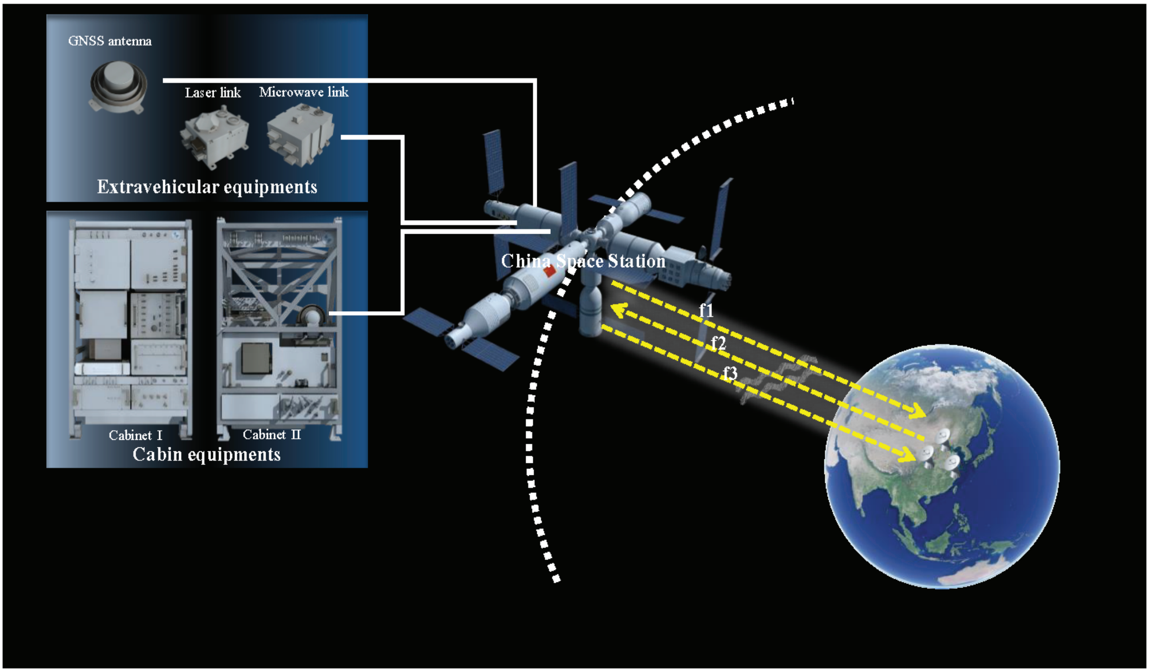


Figure 1. Satellite (CSS)–Ground Time–Frequency Comparison System.

In response to the stability assessment requirements of the high-precision time–frequency system on the CSS, a three-frequency mode has been designed for satellite–ground time–frequency comparison based on the aforementioned comparison system. When taking into full consideration the signal path effects, the three-frequency mode (1 uplink + 2 downlinks) of the satellite–ground microwave two-way link is expected to achieve ultra-high precision in satellite–ground time–frequency comparison. The uplink and downlink (f1 and f2) are used to decouple satellite–ground relative clock differences, while the two downlinks (f1 and f3) are for atmospheric error modeling. The observation model for this three-frequency-mode satellite–ground time–frequency comparison can be represented as

$$\begin{cases} p_1(t,f_1) = r_G^S(t) + \Delta T(t) + H_G^S(f_1) + \tau_{ion}(t,f_1) + \tau_{tro}(t,f_1) + \tau_{mov}(t,f_1) + \tau_{rel}(t,f_1) + \varepsilon_G^S(t,f_1) \\ p_2(t,f_2) = r_S^G(t) - \Delta T(t) + H_S^G(f_2) + \tau_{ion}(t,f_2) + \tau_{tro}(t,f_2) + \tau_{mov}(t,f_2) + \tau_{rel}(t,f_2) + \varepsilon_S^G(t,f_2) \\ p_3(t,f_3) = r_S^G(t) - \Delta T(t) + H_S^G(f_3) + \tau_{ion}(t,f_3) + \tau_{tro}(t,f_3) + \tau_{mov}(t,f_3) + \tau_{rel}(t,f_3) + \varepsilon_S^G(t,f_3) \end{cases} \quad (1)$$

where $f_1 \sim 25$ GHz, $f_2 \sim 30$ GHz, and $f_3 \sim 20$ GHz represent one uplink signal frequency and two downlink signal frequencies, respectively. The variable p stands for the carrier observation value at different frequency points, while r signifies the geometric distance equivalent time delay between satellite and ground receivers at different moments, and $\Delta T(t)$ is the relative clock offset between satellite and ground at different times. H_G^S is the sum of the ground equipment transmission channel delay and the space equipment reception channel delay. Conversely, H_S^G is the combined delay of the space equipment transmission channel and the ground equipment reception channel. The variables τ_{ion}, τ_{tro}, τ_{mov} and τ_{rel} represent the equivalent time delays for ionospheric error, tropospheric error, movement error, and relativistic effects, respectively. While some error factors are fundamentally independent of frequency, frequency is related to the signal path that is affected by frequency. In our work, frequency serves to differentiate the error factors along distinct signal paths, thus reflecting the indirect relationship between the two. ε denotes measurement error.

In two-way time–frequency transmission under satellite–ground conditions, the uplink and downlink propagation signals have an approximate symmetry in their paths. By leveraging the difference in ranging values from the uplink and downlink two-way links, we can decouple the relative clock difference between satellite and ground. Most of the transmission errors introduced by the propagation path can be either canceled out or significantly attenuated. The decoupled relative clock difference between the satellite and the ground is given by

$$\Delta T(t) = \frac{p_1(t,f_1) - p_2(t,f_2)}{2} + \Delta H_{hardware}^{f_1,f_2} + \Delta\varepsilon_{obs}^{f_1,f_2} + \Delta\tau_{ion}^{f_1,f_2} + \Delta\tau_{tro}^{f_1,f_2} + \Delta\tau_{mov}^{f_1,f_2} + \Delta\tau_{rel}^{f_1,f_2} \quad (2)$$

where p_1 and p_2 represent the ranging values of the uplink and downlink, respectively, typically obtained through transceiver devices that mutually exchange carrier waves and pseudocode ranging signals. $\Delta H_{hardware}$ denotes the combined delay error intrinsic to the hardware, usually determined via a self-closed loop calibration. $\Delta\tau_{ion}$, $\Delta\tau_{tro}$,$\Delta\tau_{mov}$ and $\Delta\tau_{rel}$ signify transmission errors along the link, corresponding respectively to ionospheric, tropospheric, motion time delay, and relativistic errors, typically estimated through precise modeling. $\Delta\varepsilon_{obs}$ is the measurement error, predominantly determined by the inherent hardware capabilities of the transmitting and receiving devices.

The precision of equipment measurement errors is the most pivotal factor in determining the accuracy of satellite–ground time delay measurements. Apart from the errors in measurement equipment, other error sources such as platform orbiting accuracy and atmospheric transmission conditions also affect the final two-way time comparison results [13]. Hence, the correction of transmission delay errors for distance measurement signals is vital for achieving high-precision two-way time comparisons. Strategies and methods for modeling various types of errors are shown in Table 1.

Table 1. Satellite–ground transmission delay errors and correction methods.

Error	Error-Handling Method
Time delay caused by motion	Corrected by model [14]
Hardware delay	Closed-loop self-calibration [15]
Time delay caused by troposphere effect	Using microwave radiometer data and model [16]
Time delay caused by ionosphere effect	Corrected by dual-downlink model [16]
Time delay caused by relativistic effect	Corrected by model [17]

Assuming the hardware capabilities meet the required standards, to attain an ultra-high time-transfer link accuracy, it is imperative to thoroughly consider the mentioned error sources and adopt appropriate methods to precisely counteract these errors to ensure the stability of the link.

3. Ground-Based Experiment

In high-precision link systems, the inherent measurement errors of the equipment significantly determine its measurement accuracy. To assess the intrinsic metrics of the link equipment developed for measurements, a co-source test scenario is contemplated. This involves using a single time–frequency reference signal for both the satellite-borne and ground-based terminals, thus eliminating errors introduced by the clock group and assessing the additional noise of the microwave time–frequency transfer link. A ground-based bidirectional test system is established as shown in Figure 1, simulating both the space and ground segments. The GNSS receiver is employed to initialize the pulse-per-second synchronization between the space and ground segments, assisting in the rapid establishment of the satellite–ground link. Subsequently, the local optical-frequency reference signal maintains the pulse moments for both the satellite-borne and ground-based terminals.

The test system utilizes optical-frequency signal transmission to achieve precise time-delay control and local oscillator signal synthesis, as shown in the equipment composition in Figure 2. The frequency signal from the cesium atomic clock is converted to an optical signal via an optical comb. Using an optical-frequency-signal phase-stable transmission device, precise time-delay control is achieved during the optical-frequency signal transmission process. Pulse repetition frequency multiplication is implemented at the optical frequency, which, after photodetection, filtering, and amplification, is converted into the desired local oscillator signal. Stable tracking and precise measurement are realized through radio frequency conversion and baseband signal processing. Furthermore, for the assurance of the test equipment's long-term stability, the experimental setup incorporates comprehensive thermal regulation measures: externally, it utilizes protective radomes and precision air conditioning to manage the ambient temperature, while internally, a secondary precision temperature control system, consisting of liquid cooling pipelines, heating films, and thermoelectric coolers (TECs), meticulously maintains the equipment's thermal state.

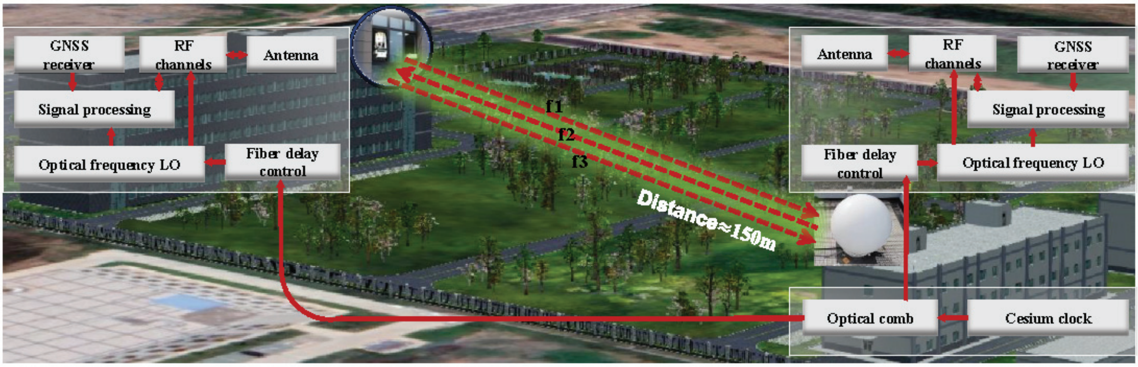


Figure 2. Ground-Based Test Experiment Setup.

The measurement accuracy at the equipment level is a critical factor constraining the high-precision time–frequency comparison between the satellite and the ground. This primarily includes random measurement errors of the equipment, equipment time delay caused by temperature changes, and jitter in the antenna phase center, among others. Based on the aforementioned ground test system, measurements obtained from satellite-borne and ground-based transceiver equipment in terrestrial conditions showed time delay measurement results for the transmission links at three frequency points. Figure 3 and Table 2 display the one-way time delay measurement results for two uplinks and one downlink in a co-source test scenario. The time deviation (TDEV) values for the three frequency points were found to be 3.72 × 10^{-3} ps at 86,400 s, 6.50 × 10^{-2} ps at 86,400 s, and 1.32 × 10^{-1} ps at 86,400 s, respectively. The root mean square (RMS) values of their link measurement errors are all less than 0.4 ps. The corresponding Allan deviation (ADEV) values were 5.84 × 10^{-19} at 86,400 s, 2.27 × 10^{-18} at 86,400 s, and 7.65 × 10^{-18} at 86,400s, respectively. This shows that the ground-based experiment for the three-frequency microwave transfer link in near space has high stability, indicating that the currently designed time delay measurement system possesses the fundamental hardware capability of achieving satellite–ground time–frequency comparisons at the 10^{-17} level. This represents nearly a 3-order-of-magnitude improvement over the precision of the BDS-3 satellite–ground time comparison system [18,19].

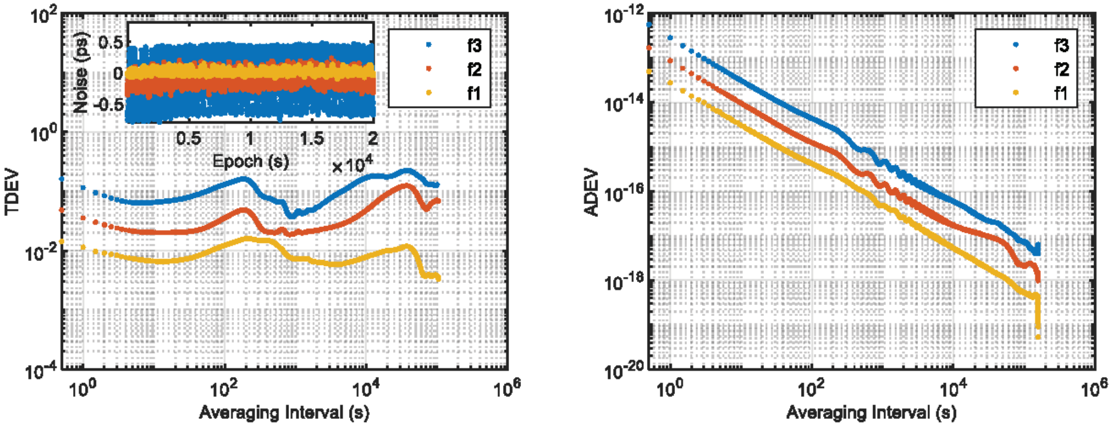


Figure 3. Experimental results of ground time–frequency transfer. Left: The TDEV (ps) of three−requency links. Right: The ADEV of three-frequency links.

Table 2. Statistics on stability of three-frequency links.

Type	Averaging Interval	CSS Receiving Link (f1)	GS Receiving Link (f2)	GS Receiving Link (f3)
TDEV	300 s 86,400 s	1.50 × 10^{-14} 3.72 × 10^{-14}	4.05 × 10^{-14} 6.50 × 10^{-14}	9.76 × 10^{-14} 1.32 × 10^{-13}
ADEV	300 s 86,400 s	1.74 × 10^{-16} 5.84 × 10^{-19}	4.73 × 10^{-16} 2.27 × 10^{-18}	1.62 × 10^{-15} 7.65 × 10^{-18}

4. Simulation and Verification of Satellite–Ground Time–Frequency Comparison

In conditions where satellite and ground equipment have functional states, the primary limiting factor of the accuracy of satellite–ground time–frequency comparison is the correction of transmission delay errors. The design of a three-frequency satellite–ground time–frequency comparison system requires modeling and correcting transmission delay errors item by item. Therefore, the orbital products and microwave radiometer monitoring parameters (pressure P, temperature T, water vapor pressure e_w) used for correction are key factors influencing correction accuracy. Among them, the accuracy of orbital products is affected by three factors: errors in precise orbit determination (POD), spacecraft attitude errors, and calibration errors in the conversion process from the tracking antenna phase center to the microwave antenna phase center. Meanwhile, the primary cause of errors in atmospheric parameter monitoring values by microwave radiometers is instrument error. According to current actual ground test results, the aforementioned error factors can generally be controlled within the index range listed in Table 3. Therefore, the simulation and analysis of satellite–ground time–frequency comparison are based on the index capabilities listed in the table.

Table 3. Error indicator settings.

Attitude error	$\leq 40''$ (3σ)
Phase center calibration error	≤ 3 mm (3σ)
POD error	≤ 10 cm (3σ)
Atmospheric parameter error	$P \leq 0.5$ hPa; T ≤ 0.5 °C; $e_w \leq 0.5$ hPa

4.1. Satellite–Ground Simulation and Verification Platform

In order to simulate and validate the satellite–ground time–frequency comparison system and its related key technologies, this paper constructs a satellite–ground time–frequency simulation and verification platform. The construction is based on the CSS orbital parameters provided by the China Manned Space Agency office (http://www.cmse.gov.cn/gfgg/zgkjzgdcs/ accessed on 16 July 2023) as shown in Table 4, as well as related parameters obtained from ground-based experiments. Utilizing this simulation platform, we conducted satellite–ground simulations to validate ultra-high-precision satellite–ground microwave time–frequency comparison technology. The architecture of the simulation and verification platform is illustrated in Figure 4.

Table 4. CSS orbital configurations.

Epoch	**24 February 2022 00:00:00 (UTC)**
Orbital parameters	Semi-major axis: 6759.9132 m, Orbit inclination: 41.4680 (degree) Eccentricity: 0.0005007, Right ascension of ascending node: 188.6126 (degree) Argument of perigee: 357.7510 (degree), Mean anomaly: 1.8149 (degree)

Within the framework of the simulation and verification platform, parameters are strictly set according to the actual conditions of the CSS [20]. In this simulation, we have comprehensively considered the Earth's model and the forces acting on the spacecraft and modeled various error factors, including geometric distance, motion time delay (distance error and clock error), atmospheric time delay, and relativistic time delay, among others [13]. This process almost accurately replicates the entire procedure of the satellite–ground time–frequency comparison system achieving two-way measurement. The simulation generates dual one-way pseudo-range observation data between the CSS and the ground (with a sampling rate of 1 Hz). Lastly, according to the simulation data, we further analyze the impact of various error factors under this new method of satellite–ground time comparison. We also validate the feasibility of achieving time stability for the satellite–ground comparison, thereby providing technical support for the processing and analysis of subsequent in-orbit experimental data.

4.2. Analysis of Transmission Delay Errors in Satellite–Ground Time–Frequency Comparison

Utilizing the CSS-to-ground simulation, we replicated the establishment of a CSS-to-ground measurement link over the course of a single day (0~86,400 s) under realistic environmental conditions. This simulation provided us with a detailed understanding of various two-way transmission delay errors in the satellite–ground comparison. As illustrated in Figure 5, each arc represents one of the successive passes of the CSS, indicating the periods when the space station is visible from a ground observer's perspective. Typically, these visibility arcs last for approximately 4 to 6 min, a time frame that is critical for the successful establishment and maintenance of a measurement link between the CSS and ground station.

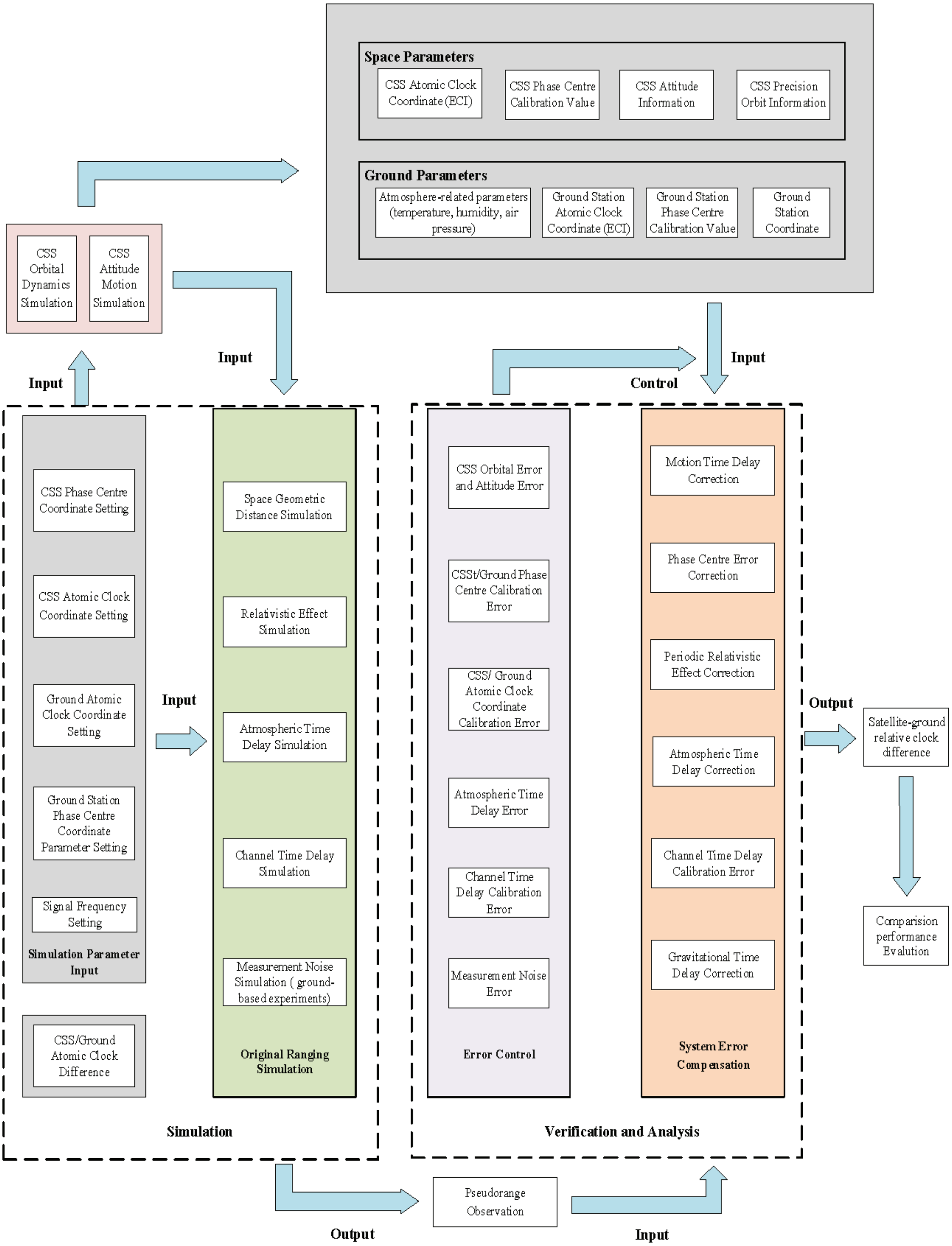

Space Parameters
CSS Atomic Clock Coordinate (ECI)
CSS Phase Centre Calibration Value
CSS Attitude Information
CSS Precision Orbit Information
Ground Parameters
Atmosphere-related parameters (temperature, humidity, air pressure)
Ground Station Atomic Clock Coordinate (ECI)
Ground Station Phase Centre Calibration Value
Ground Station Coordinate
CSS Orbital Dynamics Simulation
CSS Attitude Motion Simulation
Input
Input
Control
Input
CSS Phase Centre Coordinate Setting
CSS Atomic Clock Coordinate Setting
Ground Atomic Clock Coordinate Setting
Ground Station Phase Centre Coordinate Parameter Setting
Signal Frequency Setting
Simulation Parameter Input
CSS/Ground Atomic Clock Difference
Input
Space Geometric Distance Simulation
Relativistic Effect Simulation
Atmospheric Time Delay Simulation
Channel Time Delay Simulation
Measurement Noise Simulation (ground-based experiments)
Original Ranging Simulation
Simulation
CSS Orbital Error and Attitude Error
CSSt/Ground Phase Centre Calibration Error
CSS/ Ground Atomic Clock Coordinate Calibration Error
Atmospheric Time Delay Error
Channel Time Delay Calibration Error
Measurement Noise Error
Error Control
Motion Time Delay Correction
Phase Centre Error Correction
Periodic Relativistic Effect Correction
Atmospheric Time Delay Correction
Channel Time Delay Calibration Error
Gravitational Time Delay Correction
System Error Compensation
Verification and Analysis
Output
Satellite-ground relative clock difference
Comparision performance Evalution
Pseudorange Observation
Output
Input

Figure 4. Platform architecture of the satellite–ground simulation and verification.

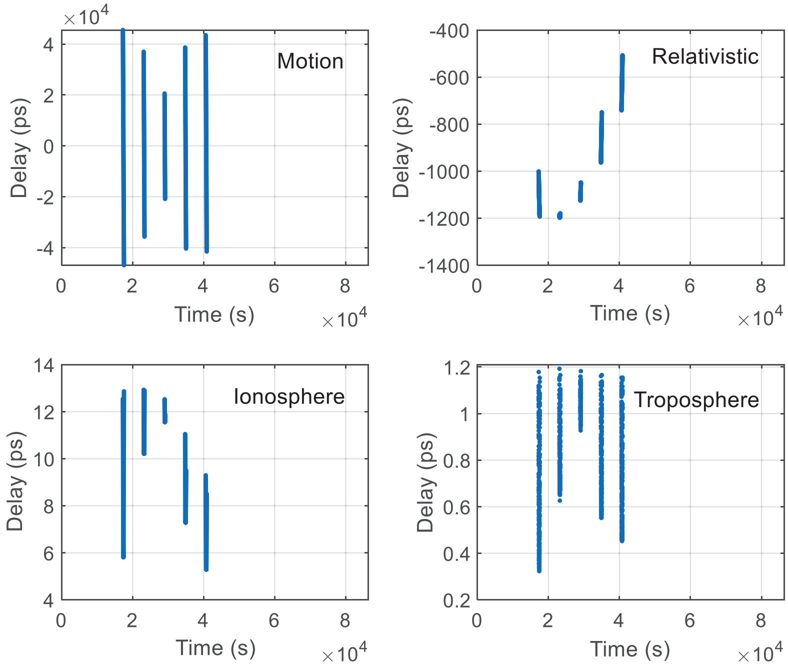


Figure 5. Two—way transmission delay errors of satellite–ground comparison.

In Figure 5, the x-axis represents time within the day in seconds, outlining the evolution of two-way time delays throughout the link establishment between the space station and the ground station. These delays encompass motion-induced delay, relativistic effects, and atmospheric interferences such as ionospheric and tropospheric delays, which align with the error terms outlined in Equation (2). Each data point plotted signifies a separate simulation instance, reflecting the variation in time delays across different runs. Because of the low Earth orbit and high velocity of the space station, the motion delay error introduced during signal transmission significantly impacts the satellite–ground time comparison. As shown, the error can accumulate to tens of nanoseconds in the process of the two-way decoupling of clock differences, thereby underscoring the necessity of correcting for the motion delay error. This is typically achieved through the use of precise ephemeris data for correction. Our simulations indicate that when applying precise ephemeris modeling to correct for motion delay error on low-Earth-orbit spacecraft, the residual uncertainty in the error can be reduced to better than picoseconds.

According to the theory of relativity, the simultaneity of spatially separated events is not absolute. In satellite–ground time–frequency comparison, relativistic errors mainly arise from the high-speed relative motion between the targets. This includes the effects of relativity on frequency (nominal frequency offset and periodic relativistic time delay) and relativistic path effects (Shapiro time delay). The nominal frequency offset caused by relativity can be precisely calibrated by lowering the frequency of the atomic clocks on the spacecraft or ground station, and it is generally insufficient to affect the stability of long-distance time comparisons. The periodic relativistic effects during the space-station-to-ground time comparison are mainly due to the different speeds and gravitational potentials of the atomic clocks on the space station and ground station in inertial space. In low Earth orbit, the spacecraft is subject to complex forces, and the gravitational potential modeling must consider various perturbing forces (mainly, Earth's gravity, tides, lunar and solar gravity, and solar radiation pressure). Gravitational time delay is caused by changes in the relative positions of the ground station, spacecraft, and Earth's center during the propagation of the satellite–ground ranging signal. As shown in Figure 5, in the high-precision time–frequency comparison experiment system of the CSS, the time-delay errors caused by relativity reached several nanoseconds. To achieve accurate estimation and correction of this part of the error, we consider establishing a high-precision model for the forces acting on the space station to mitigate the impact of periodic relativistic time-delay perturbations on time comparison stability.

Atmospheric errors (ionosphere and troposphere) during signal propagation also significantly limit the achievement of ultra-high-precision microwave satellite–ground time comparison. In the process of the bi-directional decoupling of clock differences in three-frequency mode, the error can reach tens of picoseconds, which requires careful consideration and correction. To address atmospheric errors, we propose a novel approach that eliminates tropospheric errors by modeling historical data and microwave radiometer monitoring data. In combination with two downlink links, we accurately calculate the total electron content of the ionosphere along the same path, thereby correcting the time delay errors caused by the ionosphere. Utilizing the current design of high-precision error-correction methods (see Table 1), we corrected various link delay errors in the three-frequency mode. The residual errors of the two-way time–frequency comparison after correction are illustrated in Figure 6. After correction, the RMS of the residual error of the satellite–ground two-way transmission delay errors is no more than 0.8 ps. The corresponding TDEV can achieve daily stability better than 1.33 × 10^{-3} ps, and ADEV can achieve daily stability better than 1.50 × 10^{-17}. It is evident that even after two-way differencing and correction, the transmission delay errors still have significant impacts on the stability of the satellite–ground time comparison compared to the equipment delay stability (Table 2). This makes transmission delay errors one of the main factors affecting the stability of time–frequency comparison [21]. The handling of transmission delay errors directly affects the final results of satellite–ground time comparison.

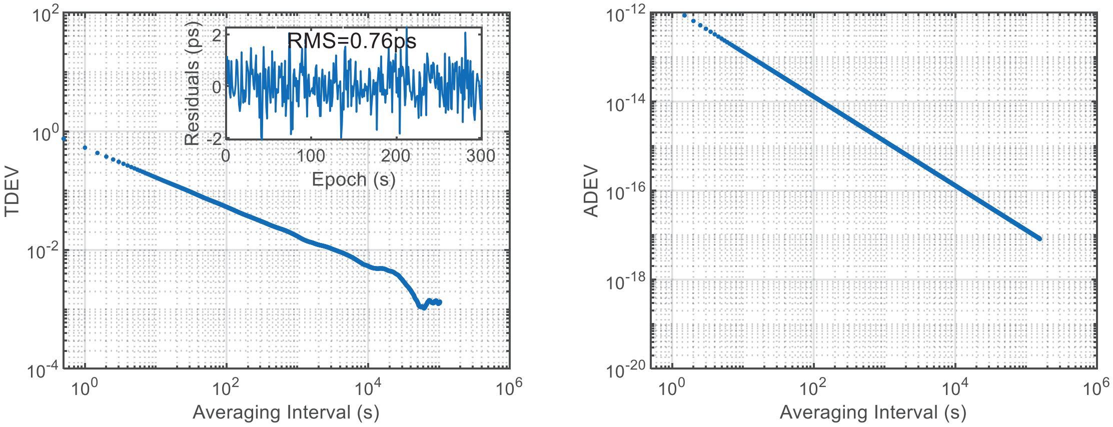


Figure 6. Stability of the residual error after correcting various transmission−delay errors in two−way time−frequency comparison.

4.3. Analysis of Satellite–Ground Time–Frequency Comparison Results

Based on the current ground experiments, and taking into account the various error factors in satellite–ground comparison as mentioned above, a semi-physical satellite–ground time comparison processing platform has been established. The satellite–ground time comparison results in the three-frequency mode are shown in Figure 7. With the help of the ultra-high−precision atomic clock on board the space station, we expect to achieve a satellite–ground time comparison precision of 0.77 ps. The corresponding TDEV can achieve 2.94 × 10^{-2} ps at 86,400 s, and ADEV can achieve 1.51 × 10^{-17} at 86,400. Therefore, considering this satellite–ground time–frequency comparison system, it is expected to achieve an 10^{-17} magnitude of stability in the CSS–ground time–frequency comparison system.

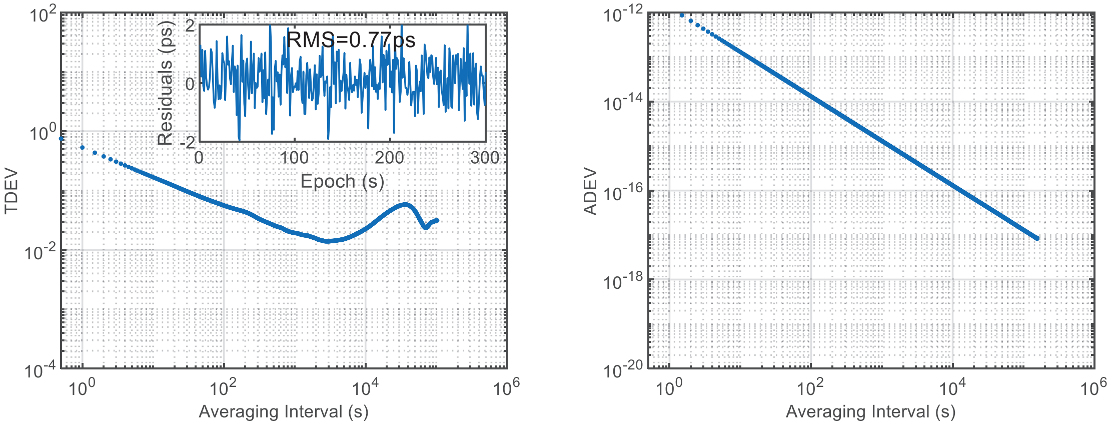


Figure 7. Final stability of the satellite (CSS)−ground time comparison.

5. Conclusions and Future Work

In response to the need for long-term stability assessment of satellite–ground time–frequency comparison in the CSS, this paper proposes a three-frequency-mode satellite–ground time difference measurement system and method. According to ground experiment results, the equipment is capable of achieving a measurement error with the RMS of less than 0.4 ps, as well as stability better than 8 × 10^{-18} at 86,400 s. Taking into account the impact of transmission delay errors and other factors (including link residual errors, hardware measurement inaccuracies, and clock noise, effectively reflecting the total uncertainty in our measurements), our comprehensive analysis and evaluation indicate that the precision of satellite–ground time comparison is expected to be better than 0.8 ps, meeting the requirements of 10^{-17} magnitude stability for satellite–ground time–frequency comparison.

Our initial use of different frequencies was a strategic choice to maintain signal integrity while developing these isolation techniques. To further enhance the precision of satellite–ground time–frequency comparison, future work will focus not only on refining corrections for atmospheric and relativistic errors but also on addressing the technical challenges associated with same-frequency uplink and downlink. We plan to optimize the system and scheme design for same-frequency communications, by implementing effective isolation measures to mitigate the potential interference of signals in space. This approach is aimed at ensuring the symmetry of the propagation paths for uplink and downlink signals, which is expected to inherently cancel out the effects of the space environment on the signals. At the same time, we will consider the impact of higher-order perturbing forces in the inertial frame (such as J2 terms and above) on the precision of two-way time comparison. We aim to establish a precision relativistic correction model based on higher-order terms to correct the satellite–ground time comparison errors caused by signal propagation, thereby achieving higher precision in satellite–ground time comparison.

Author Contributions: Conceptualization, Y.G. (Yanming Guo) and Z.P.; data curation, Y.G. (Yanming Guo); formal analysis, Y.G. (Yanming Guo) and X.G.; funding acquisition, S.G., Y.B., Y.G. (Yuping Gao), X.L. and S.Z.; methodology, Y.G. (Yanming Guo); project administration, Y.G. (Yanming Guo), S.G. and Z.P.; resources, J.C., K.S., Z.Z., Y.Y. and L.G.; validation, Y.G. (Yanming Guo), Z.P. and X.G.; writing—original draft, Y.G. (Yanming Guo) and P.W.; writing—review and editing, Y.G. (Yanming Guo) and Z.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (12273045, 42030105).

Data Availability Statement: The simulation data supporting the findings of this study are available upon request from the authors. These data are not publicly available due to privacy or ethical restrictions. For specific inquiries regarding the data, please contact the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Laurent, P.; Abgrall, M.; Clairon, A.; Lemonde, P.; Santarelli, G.; Salomon, C.; Picard, F.; Sirmain, C.; Massonnet, D.; Cacciapuoti, L. The space program PHARAO/ACES. *Proc. Spie* **2007**, *6673*, 50–62.
2. Liu, Y. *Study on Space Station and Loran Common-View Time Comparison Method*; University of Chinese Academy of Sciences: Beijing, China, 2019.
3. Origlia, S.; Pramod, M.S.; Schiller, S.; Singh, Y.; Viswam, S.; Bongs, K.; Häfner, S.; Herbers, S.; Dörscher, S.; Al-Masoudi, A.; et al. An Optical Lattice Clock Breadboard Demonstrator for the I-SOC Mission on the ISS. In Proceedings of the Lasers Electro-Optics Europe European Quantum Electronics Conference, Munich, Germany, 25–29 June 2017; p. 1.
4. Giorgetta, F.R.; Swann, W.C.; Sinclair, L.C.; Baumann, E.; Coddington, I.; Newbury, N.R. Optical two-way time and frequency transfer over free space. *Nat. Photonics* **2012**, *7*, 434–438. [CrossRef]
5. Deschênes, J.D.; Sinclair, L.C.; Giorgetta, F.R.; Swann, W.C.; Baumann, E.; Coddington, I.; Newbury, N.R. Synchronization of Optical Oscillators Over a Free-Space Link at the Femtosecond Level. In Proceedings of the Conference on Lasers and Electro-Optics (CLEO), San Jose, CA, USA, 10–15 May 2015.
6. Shen, Q.; Guan, J.Y.; Ren, J.G.; Zeng, T.; Hou, L.; Li, M.; Cao, Y.; Han, J.J.; Lian, M.Z.; Chen, Y.W.; et al. Free-space dissemination of time and frequency with 10−19 instability over 113 km. *Nature* **2022**, *610*, 661–666. [CrossRef] [PubMed]
7. Dix-Matthews, B.P.; Schediwy, S.W.; Gozzard, D.R.; Savalle, E.; Esnault, F.X.; Lévèque, T.; Gravestock, C.; D'Mello, D.; Karpathakis, S.; Tobar, M.; et al. Point-to-point stabilized optical frequency transfer with active optics. *Nat. Commun.* **2021**, *12*, 515. [CrossRef] [PubMed]
8. Leopardi, H.; Davila-Rodriguez, J.; Quinlan, F.; Olson, J.; Sherman, J.A.; Diddams, S.A.; Fortier, T.M. Single-branch Er:fiber frequency comb for precision optical metrology with 10−18 fractional instability. *Optica* **2017**, *4*, 879–885. [CrossRef]
9. Li, P.; Rolland, A.; Jiang, J.; Fermann, M.E. Coherent frequency transfer with < 5*10−21 stability via a multi-branch comb with noise cancellation. *Opt. Exp.* **2022**, *30*, 22957–22962.
10. Hećimović, Ž. Relativistic Effects on Satellite Navigation. *Tehnički Vjesnik/Technical Gazette.* **2013**, *20*, 195–203.
11. Meng, Z.; Guo, X.; Yang, J.; Yan, Q.; Zhang, Y.; Yang, F.; Wang, G.; Liu, K.; Chen, L.; Liu, Y.; et al. Method for Designing a Signal System for High-Precision Microwave Time-Frequency Transfer from Space to Ground. CN202110554869.0, 5 May 2023. Changsha Guoke Tianhe Intellectual Property Agency Co., Ltd. Available online: https://d.wanfangdata.com.cn/patent/CN202310101879.8 (accessed on 12 July 2023).
12. Shen, W.; Zhang, P.; Shen, Z.; Xu, R.; Sun, X.; Ashry, M.; Ruby, A.; Xu, W.; Wu, K.; Wu, Y.; et al. Testing Gravitational Redshift Based on Microwave Frequency Links Onboard the China Space Station. *Phys. Rev. D* **2023**, *108*, 064031. [CrossRef]
13. Stone, W. *Two-Way Satellite Time and Frequency Transfer (TWSTFT): Principle, Implementation, and Current Performance*; Wiley-IEEE Press: Hoboken, NJ, USA, 2009.
14. Pan, J.Y.; Hu, X.G.; Tang, C.P.; Zhou, S.S.; Li, R.; Zhu, L.F.; Tang, G.F.; Hu, G.M.; Chang, Z.Q.; Wu, S.; et al. System error calibration for time division multiple access inter-satellite payload of new-generation Beidou satellites. *Chin. Sci. Bull.* **2017**, *62*, 2671–2679. [CrossRef]
15. Yang, Y.; Yang, Y.; Hu, X.; Tang, C.; Guo, R.; Zhou, S.; Xu, J.; Pan, J.; Su, M. BeiDou-3 broadcast clock estimation by integration of observations of regional tracking stations and inter-satellite links. *GPS Solut.* **2021**, *25*, 57. [CrossRef]
16. Zhang, S.; Gao, S.; Bai, Y.; Guo, Y.; Pan, Z.; Lu, X. A High-Precision Satellite-Ground Time Comparison Method and System Based on Tri-frequency Mode. CN202110554869.0. 21 May 2022. Available online: https://xueshu.baidu.com/usercenter/paper/show?paperid=1q5a0pc0x2630jh0ay0k0pq09g273141&site=xueshu_se (accessed on 24 July 2022).
17. Han, C.; Cai, Z. Relativistic effects to the onboard BeiDou satellite clocks. *Navigation* **2019**, *66*, 49–53. [CrossRef]
18. Sun, L.; Gao, Y.; Huang, W.; Li, P.; Zhou, Y.; Yang, J. Autonomous Time Synchronization Using BeiDou Inter-satellite Link Ranging. In Proceedings of the 2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP), Chongqing, China, 11–13 December 2019; pp. 1–5. [CrossRef]
19. Pan, J.; Hu, X.; Zhou, S.; Tang, C.; Guo, R.; Zhu, L.; Tang, G.; Hu, G. Time synchronization of new-generation BDS satellites using inter-satellite link measurements. *Adv. Space Res.* **2018**, *61*, 145–153. [CrossRef]

20. Sun, X.; Shen, W.B.; Shen, Z.; Cai, C.; Xu, W.; Zhang, P. Formulation to test gravitational redshift based on the tri-frequency combination of ACES frequency links. *Eur. Phys. J. C.* **2021**, *81*, 634. [CrossRef]
21. Delva, P.; Meynadier, F.; Le Poncin-Lafitte, C.; Laurent, P.; Wolf, P. Time and frequency transfer with a MicroWave Link in the ACES/PHARAO Mission. In Proceedings of the 2012 European Frequency and Time Forum, Gothenburg, Sweden, 23–27 April 2012; pp. 28–35.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Article

An Improved Carrier-Smoothing Code Algorithm for BDS Satellites with SICB

Qichao Zhang [1], Xiaping Ma [1,*], Yuting Gao [1], Gongwen Huang [2] and Qingzhi Zhao [1]

[1] School of Geomatics, Xi'an University of Science and Technology, Xi'an 710054, China; 1004917400zqc@gmail.com (Q.Z.); ygao@xust.edu.cn (Y.G.); zhaoqingzhia@xust.edu.cn (Q.Z.)
[2] Geodetic Data Processing Centre of Ministry of Natural Resources, Xi'an 710054, China; hgwbds@126.com
* Correspondence: xpmaxkd16@xust.edu.cn

Abstract: Carrier Smoothing Code (CSC), as a low-pass filter, has been widely used in GNSS positioning processing to reduce pseudorange noise via carrier phases. However, current CSC methods do not consider the systematic bias between the code and carrier phase observation, also known as Satellite-induced Code Bias (SICB). SICB has been identified in the BDS-2 and the bias will reduce the accuracy or reliability of the CSC. To confront bias, an improved CSC algorithm is proposed by considering SICB for GEO, IGSO, and MEO satellites in BDS constellations. The correction model of SICB for IGSO/MEO satellites is established by using a 0.1-degree interval piecewise weighted least squares Third-order Curve Fitting Method (TOCFM). The Variational Mode Decomposition combined with Wavelet Transform (VMD-WT) is proposed to establish the correction model of SICB for the GEO satellite. To verify the proposed method, the SICB model was established by collecting 30 Multi-GNSS Experiment (MGEX) BDS stations in different seasons of a year, in which the BDS data of ALIC, KRGG, KOUR, GCGO, GAMG, and SGOC stations were selected for 11 consecutive days to verify the effectiveness of the algorithm. The results show that there is obvious SICB in the BDS-2 Multipath (MP) combination, but the SICB in the BDS-3 MP is smaller and can be ignored. Compared with the modeling in the references, TOCFM is more suitable for IGSO/MEO SICB modeling, especially for the SICB correction at low elevation angles. After the VMD-WT correction, the Root Mean Square Error (RMSE) of SICB of B1I, B2I, and B3I in GEO satellites is reduced by 53.35%, 63.50%, and 64.71% respectively. Moreover, we carried out ionosphere-free Single Point Positioning (IF SPP), Ionosphere-free CSC SPP (IF CSC SPP), CSC single point positioning with the IGSO/MEO SICB Correction based on the TOCFA Method (IGSO/MEO SICB CSC), and CSC single point positioning with the IGSO/MEO/GEO SICB correction based on VMD-WT and TOCFA (IGSO/MEO/GEO SICB CSC), respectively. Compared to IF SPP, the average improvement of the IGSO/MEO/GEO SICB CSC algorithm in the north, east, and up directions was 24.42%, 27.94%, and 24.98%, respectively, and the average reduction in 3D RMSE is 24.54%. Compared with IF CSC SPP, the average improvement of IGSO/MEO/GEO SICB CSC is 7.03%, 6.50%, and 10.48% in the north, east, and up directions, respectively, while the average reduction in 3D RMSE was 9.86%. IGSO/MEO SICB mainly improves the U direction positioning accuracy, and GEO SICB mainly improves the E and U direction positioning accuracy. After the IGSO/MEO/GEO SICB correction, the overall improvement was about 10% and positioning improved to a certain extent.

Keywords: BeiDou navigation satellite system; carrier-smoothing code; satellite-induced code bias; multipath combination; variational mode decomposition; wavelet transform

Citation: Zhang, Q.; Ma, X.; Gao, Y.; Huang, G.; Zhao, Q. An Improved Carrier-Smoothing Code Algorithm for BDS Satellites with SICB. *Remote Sens.* **2023**, *15*, 5253. https://doi.org/10.3390/rs15215253

Academic Editor: Michael E. Gorbunov

Received: 13 September 2023
Revised: 2 November 2023
Accepted: 3 November 2023
Published: 6 November 2023

1. Introduction

The development of China's BeiDou Navigation Satellite System (BDS) is divided into three stages: verification system (BDS-1), regional service system (BDS-2), and global coverage service system (BDS-3) [1]. By 2020, BDS-3 had 30 satellites (three in Geostationary Orbit (GEO), three in Inclined Geosynchronous Orbit (IGSO), and 24 in Medium Earth

Orbit (MEO)) to expand the coverage of the BDS from regional to global [2]. According to the development plan, BDS-2 and BDS-3 will continue to provide services in the coming years. Therefore, joint processing of BDS-2/BDS-3 data is crucial.

Currently, Single Point Positioning (SPP) and Differential Global Navigation Satellite Systems (DGNSS) are widely employed in navigation and positioning applications, providing accuracy at the meter and sub-meter levels [3]. Compared to carrier phase observation, the integer ambiguity is not solved in pseudorange observation; however, the noise of pseudorange observation and cycle slips limit the positioning accuracy and reliability. Carrier Smoothing Code (CSC) combines the advantages of pseudorange and carrier phase observations to improve the positioning accuracy.

CSC, also known as Hatch filtering, reduces the noise in the pseudorange without resolving the ambiguities [4]. In fact, it uses the delta range to obtain a predicted pseudorange value, which is then weighted by the average of the predicted value and the original pseudorange observation [5–7]. The delta range is derived from the carrier observations of consecutive epochs. However, Hatch filtering has problems such as ionospheric delay accumulation. For this reason, many scholars have proposed improved Hatch algorithms. Overall, these improvement strategies can broadly be categorized into two types: optimal smoothing windows and ionospheric delay compensation. To solve the problem that the ionospheric delay cannot be obtained via a single frequency receiver, Park [8,9] calculates the optimal smoothing window by introducing the Klobuchar model or the external ionospheric delay information combined with the noise model based on the elevation angle. Zhang et al. [10] used a Satellite-based Augmentation System (SBAS) technique with an ionospheric grid model combined with satellite elevation angle adaptation to determine the optimal smoothing window. Based on the theory of optimal parameter estimation, Guo et al. [11] proposed an optimal dual-frequency carrier smoothing algorithm. The results indicate that the optimal CSC algorithm outperforms traditional algorithms. Doppler observations can be computed in the delta range and are not affected by cycle slips. Liu et al. [12] proposed an optimal CSC algorithm that considers both satellite signal strength and ionospheric delay with the assistance of Doppler. Zhou and Li [13] designed pure and continuous Doppler smoothing based on the principle of minimum variance. Through experimental verification, they demonstrated their effectiveness and efficiency. Another strategy involved compensating for ionospheric delay. Zhang et al. [14] proposed an ionospheric delay self-modeling compensation single-frequency CSC algorithm specifically for single-frequency users. The effectiveness of the algorithm was validated via ship model experiments and trolley experiments. Mcgraw [5] summarized mainstream non-divergent CSC algorithms, among which dual-frequency users can compensate for ionospheric delay using dual-frequency data. In essence, the pseudorange in the CSC algorithm has correlation to solve this problem. Chen et al. [15] proposed a real-time dynamic ionospheric delay model for CSC considering colored noise based on the Kalman filter and least squares theory. This algorithm can adapt to various situations including different sampling intervals and ionospheric anomalies. Tang et al. [16] proposed a dual-frequency non-divergent BDS CSC differential positioning method. The results showed that as the baseline length increased, the positioning accuracy of B3I decreased at a higher rate than B1I. And, the Hatch algorithm was optimized for challenging environments [17,18]. Most of the research above is based on the establishment of CSC algorithms using GNSS data. Due to the unique constellation and development strategy of BDS, there has been relatively less research on CSC specifically focused on BDS.

CSC can mitigate multipath and noise, but is subject to systematic bias in pseudo-range and carrier phase observations, which means that systematic errors need to be eliminated in advance. This systematic bias has been found in BDS-2, known as Satellite-Induced Code Bias (SICB). The SICB of BDS can be divided into two categories: the first category is SICB, which varies with elevation angles for IGSO/MEO satellites, and the other category is SICB for GEO satellites. These biases have significant impacts on single point positioning and wide/narrow-lane ambiguity resolution [19]. Hauschild et al. [20] pointed out that BDS-2

IGSO/MEO satellites have SICB that result in code phase divergence exceeding 1.0 m. This error is referred to as SCIB. Based on two years of data, Gou et al. [21] developed a model for SCIB and provided correction values and accuracy indicators. This model helps refine the random model of observation. The experiments indicate that the correction model is more suitable for BDS-2. For accurate modeling of SICB, Pan et al. [22] modeled each satellite individually, while also considering the impact of inconsistent single-difference ambiguity parameters and hardware delay for Multipath (MP) mitigation. Additionally, a one-degree elevation node was used to accurately describe SICB. Their experiments demonstrated centimeter-level variations in SCIB for BDS-3 satellites. For the SICB model of the GEO satellite, Wu et al. [23] proposed the code noise and multipath correction algorithm and the results showed a 42% reduction in the standard deviation of the MP time series for GEO satellites. Ning et al. [24] analyzed GEO satellites using correlation analysis, Fourier transformation, and wavelet decomposition. The results showed that the error characteristics of C01, C02, and C04 differed from those of C03 and C05. The error sequences of C01, C02, and C04 exhibited high-frequency variations. Hu [25] et al. used the characteristic of BDS-3 frequency homogeneity to realize a one-step modeling of SICB considering the correlation. Chen [26] used EMD-WT to model GEO satellites. B1/B2 IF-PPP verified the effectiveness of the algorithm. The aforementioned model can improve the SPP accuracy. However, challenges such as insufficient data volume and the unique nature of GEO satellite orbits make it difficult to obtain high-precision models.

In the above literature, the CSC algorithm does not consider the SICB correction of BeiDou IGSO/MEO/GEO, where the SICB modeling data is less and there is less research on the characteristics of GEO SICB. In this contribution, we briefly review the basic model and error analysis of CSC. To address the ionosphere delay, we employ the ionosphere-free combination. Furthermore, we conducted extensive analysis and modeling of the SICB of the IGSO/MEO/GEO satellite based on data from 30 global Multi-GNSS Experiment (MGEX). This modeling is accomplished using the piecewise weighted least squares Third-order Curve Fitting Method (TOCFM), yielding accurate correction models. To effectively model GEO SICB, we introduce the Variational Mode Decomposition-Wavelet Transform (VMD-WT) model. This method considers the accurate characterization and correction of GEO SICB. Finally, the experimental results are analyzed and a meaningful conclusion is drawn.

2. Methods

2.1. Mathematical Model of Carrier-Smoothed Code

CSC can improve pseudorange accuracy, also known as Hatch filtering. The Hatch filter is a recursive filter that uses the current measurement and previous estimates without requiring any dynamic models or additional information and can be implemented for real-time operation in low-cost single-frequency GNSS receivers. The equation for the Hatch filter is as follows:

$$\begin{cases} \overline{P}_{CSC}\,(k) = \alpha P(k) + (1-\alpha)\left(\overline{P}_{CSC}\,(k-1) + \lambda(\phi(k) - \phi(k-1))\right) \\ \overline{P}_{CSC}\,(1) = P(1) \end{cases} \tag{1}$$

where $P(k)$ represents the raw pseudorange of epoch k. $\overline{P}_{CSC}\,(k)$ and $\overline{P}_{CSC}\,(k-1)$ represent the smoothed pseudorange of epoch k, $k-1$, where the unit is a meter. $\phi(k)$ and $\phi(k-1)$ represent carrier phase observations of epoch k, $k-1$, where the unit is a cycle; α represents the smoothing factor; and λ is the wavelength. It can be clearly seen that the essence of Hatch filtering is a weighted averaging method. The precision of smoothing pseudorange is improved continuously with the epoch iteration, and the filter must be reset when cycle slips occur.

The effect of ionospheric delay on the pseudorange and carrier phase is equal and opposite. The Hatch filter is divergent due to the influence of the ionosphere. Assuming carrier compensation for ionospheric delay, we can obtain a non-divergent Hatch filter.

$$\mathit{\Phi}(k) = \varphi(k) + 2I(k) \tag{2}$$

$$\begin{aligned} P_{CSC}(k) &= \alpha\overline{P}(k) + (1-\alpha)\big(\overline{P}_{CSC}\ (k-1) + \lambda(\mathit{\Phi}(k) - \mathit{\Phi}(k-1)\big) \\ &= \alpha\overline{P}(k) + (1-\alpha)\big(\overline{P}_{CSC}\ (k-1) + \lambda(\varphi(k) - \varphi(k-1) + 2\Delta I(k)\big) \end{aligned} \tag{3}$$

where $I(k)$ represents the variation in ionospheric delay of epoch k, and $\Delta I(k) = I(k) - I(k-1)$ represents the ionospheric delay variation from epoch $k-1$ to epoch k. In this paper, ionospheric delay is eliminated by using ionosphere-free combinations:

$$\begin{cases} P_{IF} = \frac{P_i f_i^2}{f_i^2 - f_j^2} - \frac{P_j f_j^2}{f_i^2 - f_j^2} \\ \varphi_{IF} = \frac{\varphi_i f_i^2}{f_i^2 - f_j^2} - \frac{\varphi_j f_j^2}{f_i^2 - f_j^2} \end{cases} \tag{4}$$

Replace the $\overline{P}, \mathit{\Phi}$ in Equation (4) with the ionosphere-free combination:

$$\overline{P}_{IF,CSC}(k) = \alpha\overline{P}_{IF}(k) + (1-\alpha)\big(\overline{P}_{IF,CSC}(k-1) + \lambda(\phi_{IF}(k) - \phi_{IF}(k-1)\big) \tag{5}$$

where $(\cdot)_{IF}$ represents the ionosphere-free combination; f are the frequencies; and the subscripts i, j are used to denote different frequencies.

The ionosphere-free combination requires Timing Group Delay (TGD) correction [27],

$$\begin{cases} P_{12} = \rho - \frac{\alpha}{\alpha-1}TGD_1 + \frac{1}{\alpha-1}TGD_2 + T + \varepsilon_{12} \\ P_{13} = \rho - \frac{\beta}{\beta-1}TGD_1 + T + \varepsilon_{13} \\ P_{23} = \rho - \frac{k}{k-1}TGD_2 + T + \varepsilon_{23} \end{cases} \tag{6}$$

where $P_{ij}(i,j = 1,2,3)$ is the ionosphere-free pseudorange observable in meters; ρ is the true geometric distance between the satellite and receiver; T is the slant troposphere delay; α, β, and k are constant frequency-dependent multiplier factors ($\alpha = f_1^2/f_2^2$, $\beta = f_1^2/f_3^2$, $k = f_2^2/f_3^2$); and ε_{ij} $(i,j = 1,2,3)$ are the ionosphere-free noise. Note that the TGD parameter is provided in the broadcast ephemeris.

2.2. Error Analysis

We assume that the observation variances in the ionosphere-free combination carrier phase and the pseudorange are Q_P and Q_ϕ. The observed values are mutually uncorrelated. Continuous epoch data can be obtained according to Equation (5),

$$\begin{pmatrix} \overline{P}_{IF,CSC}(1) \\ \overline{P}_{IF,CSC}(2) \\ \vdots \\ \overline{P}_{IF,CSC}(k-1) \\ \overline{P}_{IF,CSC}(k-1) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 \\ \frac{1}{2} & -\frac{\lambda}{2} & \frac{1}{2} & \frac{\lambda}{2} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \cdots & \vdots & \vdots \\ \frac{1}{k-1} & -\frac{\lambda}{k-1} & \cdots & \cdots & \frac{1}{k-1} & -\frac{(k-2)\lambda}{k-1} & 0 & 0 \\ \frac{1}{k} & -\frac{\lambda}{k} & \cdots & \cdots & \frac{1}{k} & -\frac{\lambda}{k} & \frac{1}{k} & \frac{(k-1)\lambda}{k} \end{pmatrix} \begin{pmatrix} P_{IF}(1) \\ \phi_{IF}(1) \\ P_{IF}(2) \\ \phi_{IF}(2) \\ \vdots \\ P_{IF}(k-1) \\ \phi_{IF}(k-1) \\ P_{IF}(k) \\ \phi_{IF}(k) \end{pmatrix} \tag{7}$$

According to the error covariance propagation law, we can derive the variance matrix of the smoothed pseudorange.

$$Q_{IF,CSC} = \begin{pmatrix} Q_P & \frac{Q_P}{2} & \cdots & \frac{Q_P}{k-1} & \frac{Q_P}{k} \\ \frac{Q_P}{2} & \frac{Q_P}{2} & & \vdots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{Q_P}{k-1} & \frac{Q_P}{k-1} & \cdots & \frac{Q_P}{k-1} & \frac{Q_P}{k} \\ \frac{Q_P}{k} & \frac{Q_P}{k} & \cdots & \frac{Q_P}{k} & \frac{Q_P}{k} \end{pmatrix} + \begin{pmatrix} 0 & & & 0 \\ & \frac{\lambda^2 Q_\phi}{2} & & \\ & & \frac{\lambda^2 (k-2) Q_\phi}{k-1} & \\ 0 & & & \frac{\lambda^2 (k-1) Q_\phi}{k} \end{pmatrix} \tag{8}$$

From the above equation, it is apparent that there is a correlation between the epochs of the smoothed pseudorange. As the epochs increase, the variance in the smoothed pseudorange approaches $\frac{Q_P}{k}$, signifying a continuous enhancement in accuracy.

2.3. Analysis of the SICB of the Satellite

The multipath combination is a special linear combination of single-frequency code and dual-frequency phase measurements which has been widely used to analyze the SICB [22,26,27]. The following formula shows the MP combination observation value at the frequency i. It should be noted that in practical applications, the carrier phase observation value usually adopts the same frequency as the pseudorange observation; that is, $i = j$ or $i = q$:

$$\begin{aligned} MP_i &= P_i - (m_{ijq} - 1) \cdot \varphi_j - m_{ijq} \cdot \varphi_q \\ &= N_{e,ijq} + B_i \end{aligned} \tag{9}$$

with

$$\begin{cases} m_{ijq} = (\lambda_i^2 + \lambda_j^2)/(\lambda_i^2 - \lambda_q^2) \\ N_{e,ijq} = (m_{ijq} - 1)N_j - m_{ijq} * N_q \\ B_i = M_I + \varepsilon_I + (m_{ijq} - 1)(m_j + \delta_j) - m_{ijq}(m_j + \delta_j) \end{cases} \tag{10}$$

where i, j, q denotes the carrier frequencies, m_{ijq} denotes the linear factor, MP is the MP combination, $N_{e,ijq}$ refers to carrier phase ambiguity involving constant satellite and receiver-dependent hardware delays, and B is in respect to the sum of the multipath and measurement noise.

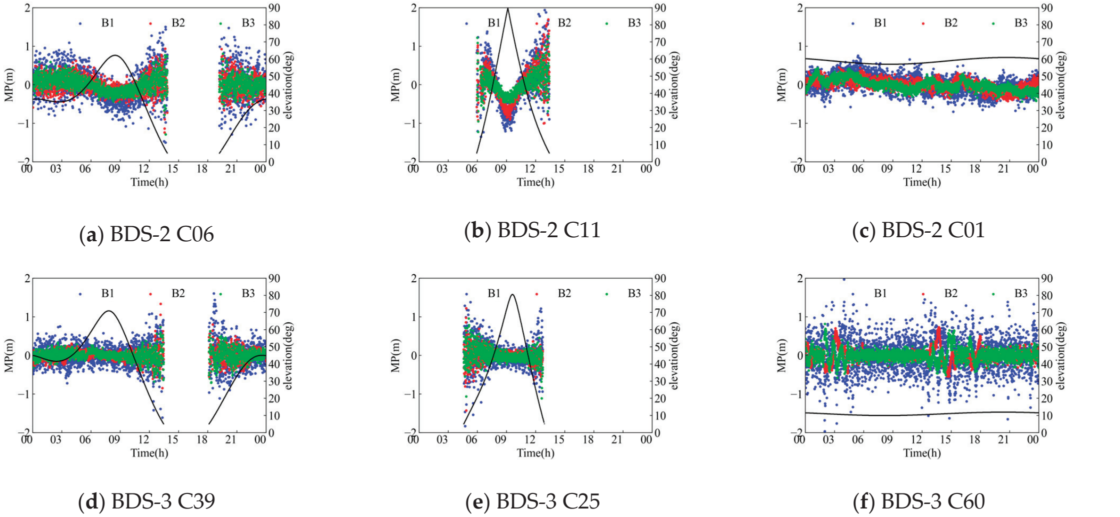
The above model eliminates first-order ionospheric delay and geometric distance. In addition, the carrier phase multipath can be ignored because it is smaller than the pseudorange. Therefore, Equation (10) is suitable for establishing SICB characteristics. Since the carrier ambiguity and hardware delay remain relatively constant in the absence of cycle slips, the ambiguity and hardware delay in the multipath can be weakened by subtracting the average value of a time series.

$$MP_i = MP_i - \frac{\sum_n^i MP_i}{n} \tag{11}$$

where i represents the epoch count, and n represents the length of the time series without cycle slips.

Figure 1 shows the MP time series of BDS2/3 for 7 June 2023 from Figure 1a–f.

Figure 1 shows the MP time series of BDS-3 is more dispersed due to the large observation noise when the elevation angle is small. In the MP time series of IGSO and MEO satellites, the systematic bias of BDS-2 satellite is actually SICB, which increases with the increase in elevation angle, especially at B1I and B2I frequencies.



(**a**) BDS-2 C06

(**b**) BDS-2 C11

(**c**) BDS-2 C01

(**d**) BDS-3 C39

(**e**) BDS-3 C25

(**f**) BDS-3 C60

Figure 1. MP time series and elevation angle (black line) at different frequencies of the BDS2/3: (**a**) C06 satellite of BDS-2, (**b**) C11 satellite of BDS-2, (**c**) C01 satellite of BDS-2, (**d**) C39 satellite of BDS-3, (**e**) C25 satellite of BDS-3, and (**f**) C60 satellite of BDS-3.

3. Results and Discussion

3.1. Data Analysis and Model Establishment

In order to analyze the characteristics of BDS SICB, the BDS observation data of more than 30 Multi-GNSS Experiment (MGEX) stations from the International GNSS Service (IGS) in different seasons were selected uniformly, as shown in Table 1. The stations' observation environment is ideal with minimal multipath effects. Only the stations located in East Asia and Australia can obtain complete and high quality IGSO satellite observation data. Therefore, when analyzing IGSO satellites, only the stations in East Asia and Australia are used. The distribution of the stations is shown in Figure 2. The red dots represent IGSO modeling stations, where all stations are used for MEO satellite modeling, and the outbound stations are labeled as subsequent algorithm verification stations.

Table 1. Observation data period.

SICB	Data Period
Modeling Datasets	2022: Day of Year (DOY) 251–257, DOY 315–312, 2023: DOY 001–007, DOY 158–164,
Validation Dataset	2023: DOY 200–210

In order to explore the relationship between SICB and stations, seasons and satellites, we compare the MP and elevation angle of different satellites of same station, the MP and elevation angle of different stations, and the satellite elevation angle and MP of same station in different periods. Figure 3 shows the relationship between MP and the elevation of different satellites from the PPTG station. Figure 4 shows the relationship between MP and the elevation angle of the ALIC, PTGG, CUSV, and ULAB stations at the same time. Figure 5 shows the relationship between MP and the elevation angle of PTGG stations at different dates (DOY001, 158, 251, and 315). SICB shows the same trend at different stations, with different satellites and times; however, it clearly differs at different frequencies. It is concluded that SICB is not related to the seasons, stations, and satellites, but is related to frequency.

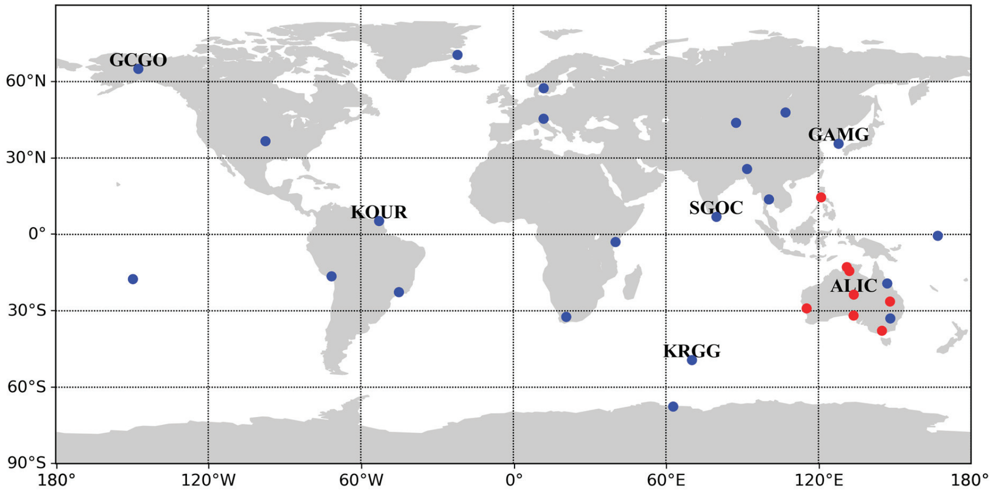
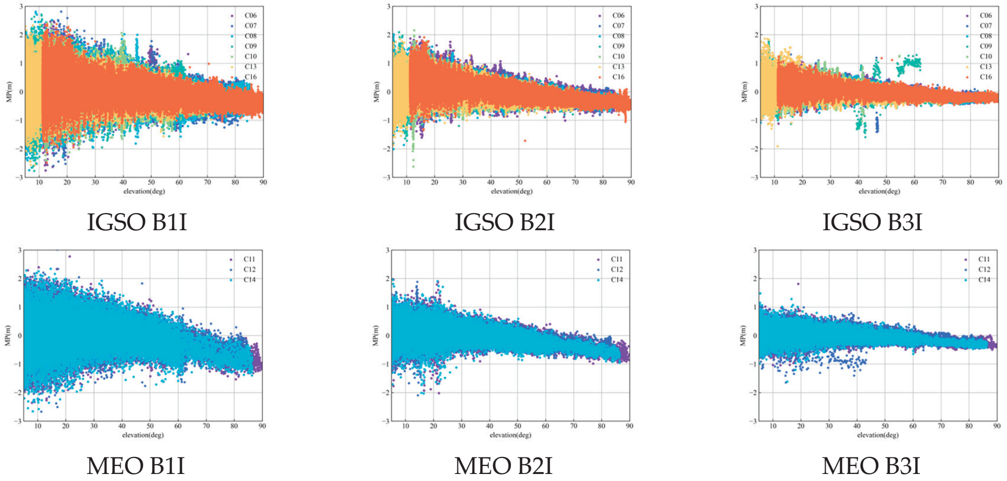


Figure 2. Stations distribution (The blue and red dots represent the 30 stations, of which the blue plus red dots model the MEO satellite, the red dots model the IGSO satellite, and the dots marked with the name are the verification stations).



IGSO B1I

IGSO B2I

IGSO B3I

MEO B1I

MEO B2I

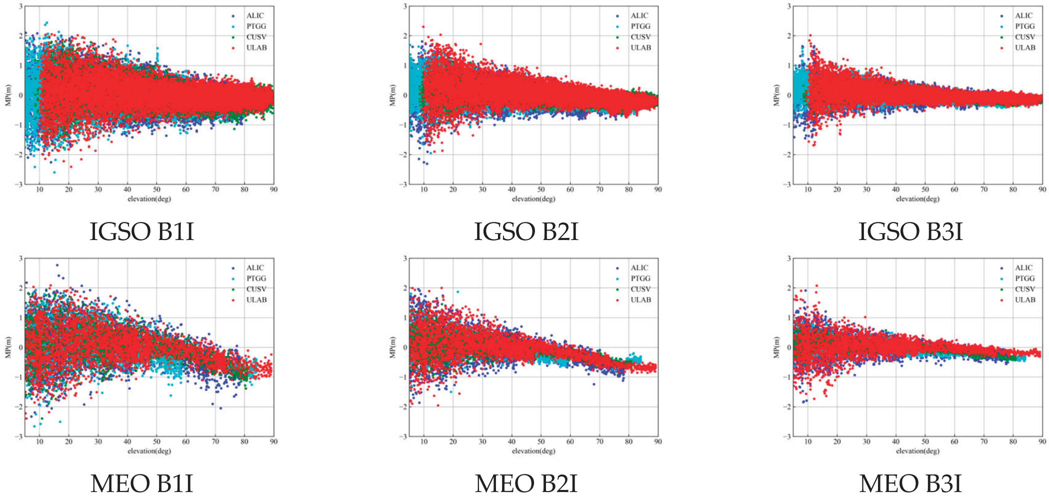
MEO B3I

Figure 3. The relationship between MP and elevation angle of different satellites at PPTG stations.

To further investigate the correlation between the SICB and elevation angle, the Pearson correlation coefficient is calculated for all selected IGSO/MEO satellites at the chosen stations. The formula for the calculation is as follows:

$$Person = \frac{\sum MP \cdot El - \frac{\sum MP \sum El}{N}}{\sqrt{(\sum MP^2 - \frac{(\sum MP)^2}{N}(\sum El^2 - \frac{(\sum El)^2}{N})}} \tag{12}$$

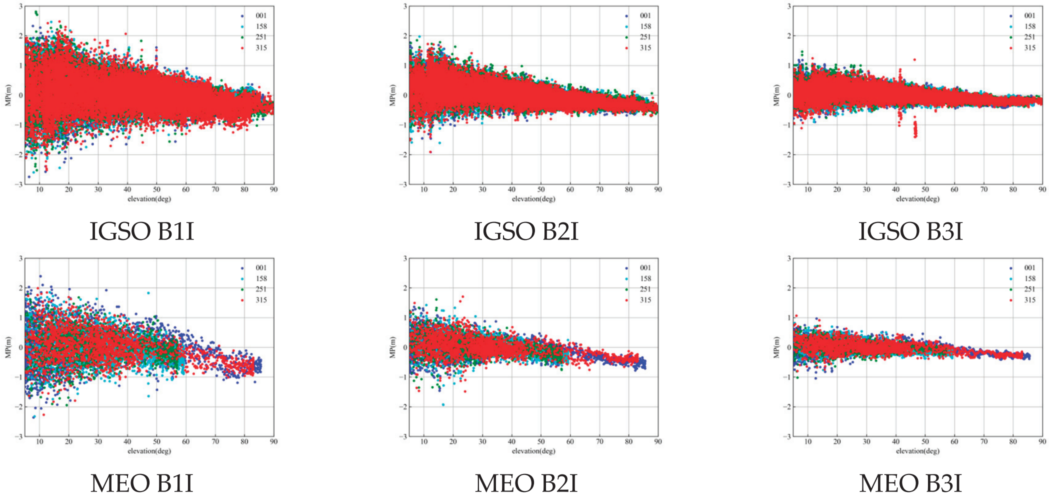
where *El* and *N* represent the elevation angle and the length of data, and *Person* represents the correlation coefficient, with a range from −1 to 1. The larger the absolute value, the stronger the correlation. Coefficients between 0.2 and 0.4 indicate weak correlation, 0.4–0.6 indicate moderate correlation, 0.6–0.8 indicate strong correlation, and values greater than 0.8 indicate extremely strong correlation. The correlation coefficients are shown in Table 2.



IGSO B1I IGSO B2I IGSO B3I

MEO B1I MEO B2I MEO B3I

Figure 4. The relationship between MP and elevation angle at different stations.



IGSO B1I IGSO B2I IGSO B3I

MEO B1I MEO B2I MEO B3I

Figure 5. The relationship between MP and elevation angle of PPTG stations at different periods.

Table 2. Statistical table of elevation angle and MP correlation coefficient.

System	IGSO			MEO		
	B1I	**B2I/B2a**	**B3I**	**B1I**	**B2I/B2a**	**B3I**
BDS-2	−0.2312	−0.3518	−0.2998	−0.3428	−0.3932	−0.2733
BDS-3	−0.0240	0.0143	0.0512	−0.0103	0.0422	−0.0181

The MP of BDS-2 IGSO/MEO is weakly negatively correlated with the elevation angle of the satellite. The MP of B2I is slightly higher than that of B1I and B3I. The MP between B1I, B2a, and B3I in BDS-3 is not correlated, indicating that the SICB of BDS-3 can be ignored. Therefore, this paper models and corrects the observation quantity of the BDS-2 satellite according to the IGSO/MEO satellite type.

This paper classifies satellite data based on IGSO/MEO, using an interval of 0.1 degrees, and takes the average value of MP that does not exceed 0.1 degrees. The Piecewise Third-Order Curve Fitting Method (TOCFM) with 30-degree nodes is used to fit the MP sequence, while also ensuring the global continuity of the nodes.

$$y = a_0 + a_1 \cdot El + a_2 \cdot El^2 \tag{13}$$

where y is the MP, El is the elevation angle, and a_0, a_1, a_2 are the three fitting coefficients. When the n epochs are observed, the MP value and elevation angle corresponding to each epoch can be obtained. The parameters (a_0, a_1, a_2) can be solved by using the least square method. The global fitting needs to satisfy the continuity of the boundary point. The minimum variance and the formula is as follows:

$$\begin{cases} \sum\limits_{k}^{m-1} \sum\limits_{l}^{n_k} W_{k,l} \cdot (y_{k,l} - MP_{k,l})^2 = \min \\ y_k(El_k) - y_{k+1}(El_k) = 0 \end{cases} \tag{14}$$

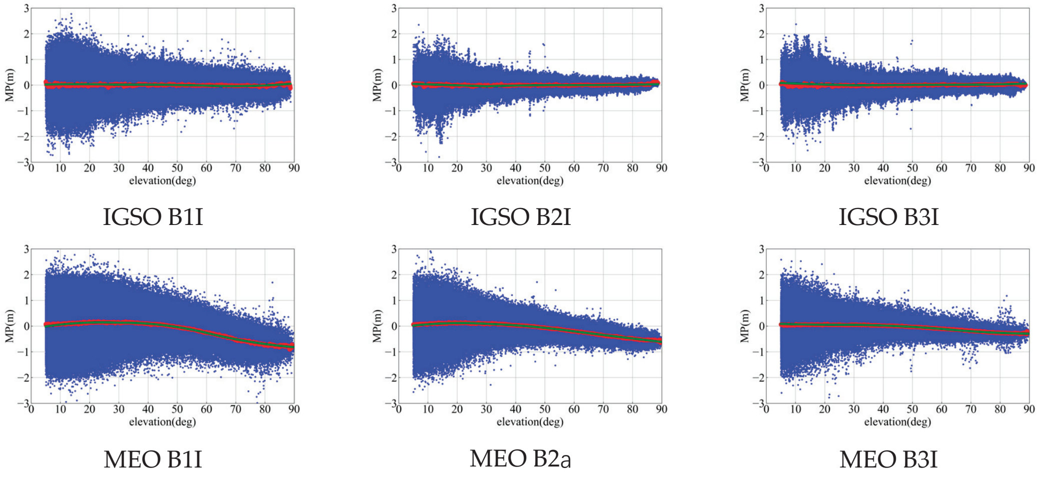
where k represents the segment point, l represents the data epoch of the segment, and m and n represent the number of segments and the length of each segment, respectively.

The BDS MP time series shows significant noise at a low elevation. Considering the different accuracy of MP under different elevation angles, different weights are set for MP observations corresponding to different elevation angles in the fitting process. The weight design scheme is as follows:

$$W_{k,l} = a^2 + \frac{a^2}{\sin^2(El_{k,l})} \tag{15}$$

where a is usually 0.3, which is the pseudorange variance of the BDS observation, and $El_{k,l}$ represents the elevation angle corresponding to point l in piecewise k.

Figure 6 and Table 3 show that the fitting curve of BDS-3 is obviously close to 0, indicating that the BDS-3 SICB is small and negligible. The fitting curve of BDS-2 has different trends. Piecewise fitting can refine the SICB modeling at low elevation angles.



IGSO B1I

IGSO B2I

IGSO B3I

MEO B1I

MEO B2a

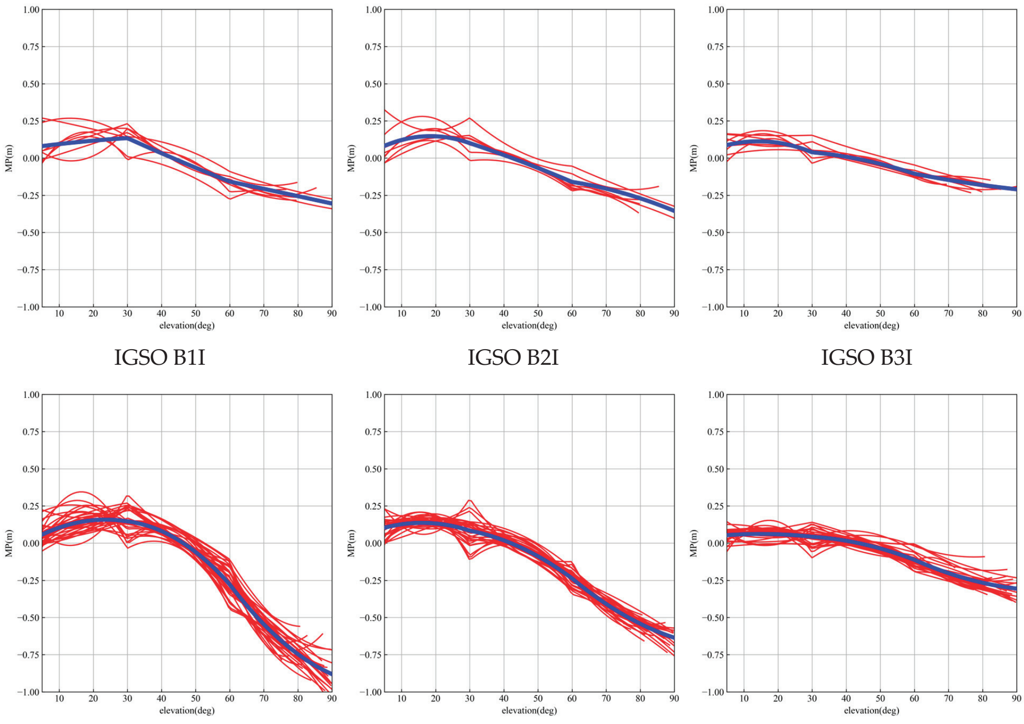
MEO B3I

Figure 6. Partial BDS 2/3 SICB fitting results (blue represents raw data, red represents redefined data, and green represents third-order fitting curve).

Table 3. Correction coefficients for SICB related to elevation angle.

Segmentation (°)	Parameter	IGSO			MEO		
		B1I	B2I	B3I	B1I	B2I	B3I
5–30	a_0	0.0510	0.0883	0.0705	0.0434	0.0673	−0.0036
	a_1	0.0050	0.0074	0.0044	0.0024	0.0086	0.0137
	a_2	-6.14×10^{-5}	-1.94×10^{-4}	-1.54×10^{-4}	-8.21×10^{-5}	-2.69×10^{-4}	-2.89×10^{-4}
30–60	a_0	0.5297	0.5374	0.1706	−0.0233	0.0273	0.1153
	a_1	−0.0148	−0.0157	−0.0031	0.0059	0.0082	0.0201
	a_2	6.60×10^{-5}	7.71×10^{-5}	-1.79×10^{-5}	-1.23×10^{-4}	-2.10×10^{-4}	-3.80×10^{-4}
60–90	a_0	1.7007	0.7604	0.6366	0.9484	1.6042	2.7324
	a_1	−0.0513	−0.0219	−0.0188	−0.0252	−0.0423	−0.0702
	a_2	3.50×10^{-4}	1.19×10^{-4}	1.15×10^{-4}	1.25×10^{-4}	1.93×10^{-4}	3.34×10^{-4}

Figure 7 shows the comparison between the modeling results of a single station and multiple stations. The modeling trend of the single station and multi-station is the same, but there is a certain deviation in the value. The MP of B1I, B2I, and B3I have the same code bias trend and the value is slightly different.



IGSO B1I IGSO B2I IGSO B3I

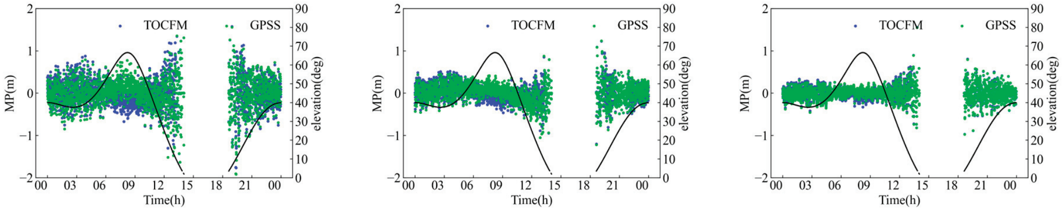
MEO B1I MEO B2I MEO B3I

Figure 7. Comparison of single- and multi-station modeling (red represents the modeling results of a single station, and blue represents the modeling results of all stations).

Table 3 shows correction coefficients related to the MP time series and the elevation angle calculated by Equation (13) based on all observation data of the 30 stations around the world.

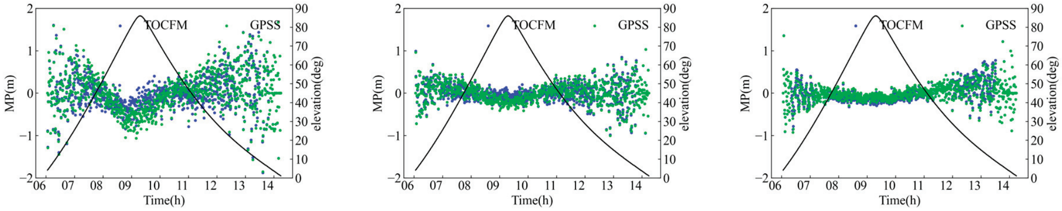
3.2. Analysis of the IGSO/MEO SICB Model Results

MP was corrected according to the correction coefficients of SICB calculated in Table 3. In order to verify the validity of the model, the correction coefficient of the reference [28] was compared (the model in the reference was named GPSS). Figures 8 and 9 show the comparison of the correction effect between TOCFM and GPSS. The correction effect of TOCFM was better than that of GPSS at different elevation angles due to TOCFM adopting 0.1-degree interval piecewise fitting. Different correction effects indicate that different modeling data will produce different modeling coefficients.



B1I B2I B3I

Figure 8. MP and elevation series before and after the correction of C16 SICB.



B1I B2I B3I

Figure 9. MP and elevation series before and after the correction of C11 SICB.

In Figures 3–9, it is evident that the MP of B1I exhibits the largest bias. B2I follows and B3I shows the smallest bias. After the correction, the SICB of IGSO/MEO satellites was effectively improved. Additionally, it can be observed that under the piecewise fitting method, the SICB at low elevation angles is mitigated to some extent.

3.3. SICB Correction Model of GEO Satellites

SICB modeling methods for GEO satellites, such as wavelet transform, regularization and machine learning methods. Variational Mode Decomposition (VMD) is a signal decomposition method that decomposes nonlinear non-stationary signals into a discrete number of modes. Compared to the widely used Empirical Mode Decomposition (EMD) method, it has the following advantages:

(1) Through rigorous mathematical derivation, the theory is rigorous.
(2) It overcomes the problem of modal aliasing in EMD.
(3) It overcomes the breakpoint effect in EMD.
(4) It has good noise and sampling robustness.

For detailed theory, please refer to [29].

In this paper, we used the VMD-WT model to analyze and correct GEO SICB. It decomposes the original signal using VMD. Since each Intrinsic Mode Function (IMF) component has a different center frequency, different wavelet threshold functions are needed to improve the denoising effect. Therefore, different wavelet threshold functions are used for different IMF components.

The wavelet coefficients of the soft threshold function and adaptive threshold function are expressed as:

$$\hat{\omega}_{j,k}^{A} = \begin{cases} \operatorname{sgn}(\omega_{j,k}^{A})(\left|\omega_{j,k}^{A}\right| - \mu_j^{A}), & \left|\omega_{j,k}^{A}\right| \geq \mu_j^{A} \\ 0, & \left|\omega_{j,k}^{A}\right| < \mu_j^{A} \end{cases} \tag{16}$$

$$\hat{\omega}_{j,k}^{B} = \begin{cases} \operatorname{sgn}(\omega_{j,k}^{B})(\left|\omega_{j,k}^{B}\right| - \frac{\mu_j^{B}}{(\left|\omega_{j,k}^{B}\right|^2 - \left|\mu_j^{B}\right|^2)^{J} + 1}), & \left|\omega_{j,k}^{B}\right| \geq \mu_j^{B} \\ 0, & \left|\omega_{j,k}^{B}\right| < \mu_j^{B} \end{cases} \tag{17}$$

where $\operatorname{sgn}(\cdot)$ is the sign function, $\hat{\omega}_{j,k}^{A}$, $\hat{\omega}_{j,k}^{B}$ are the k-th wavelet coefficients of the j-th layer after the DWT, and J is the total number of wavelet reconstruction layers. In addition, the threshold μ_j^{A} and threshold μ_j^{B} are calculated from:

$$\begin{cases} \mu_j^{A} = \sigma_j \sqrt{2\ln(N_j)} \\ \mu_j^{B} = \sigma_j \sqrt{2\ln(N_j)} / \ln(j+1) \end{cases} \tag{18}$$

where the current number of reconstruction layers is j, the noise standard deviation is $\sigma_j = median(\left|\omega_{j,k}\right|)/0.6745$, and N_j is expressed as the length of the current reconstruction layer.

The principle of VMD-WT is to maximize the noise reduction performance while maintaining the local characteristics of the original signal. Soft thresholding can reduce signal loss in high-frequency signals. The adaptive threshold function is used to remove noise components in the MP time series. It can also avoid the constant bias of soft thresholding and the discontinuity of hard thresholding. The recommended frequency is 50 mHZ under good observation conditions. This particular setup is the best choice for obtaining precise results in this situation. The data processing flow is shown in Figure 10.

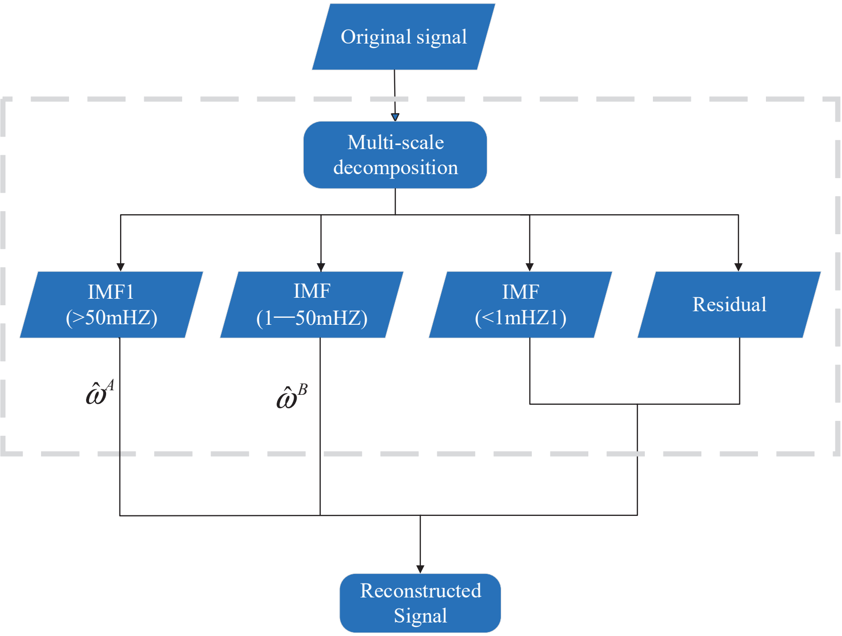
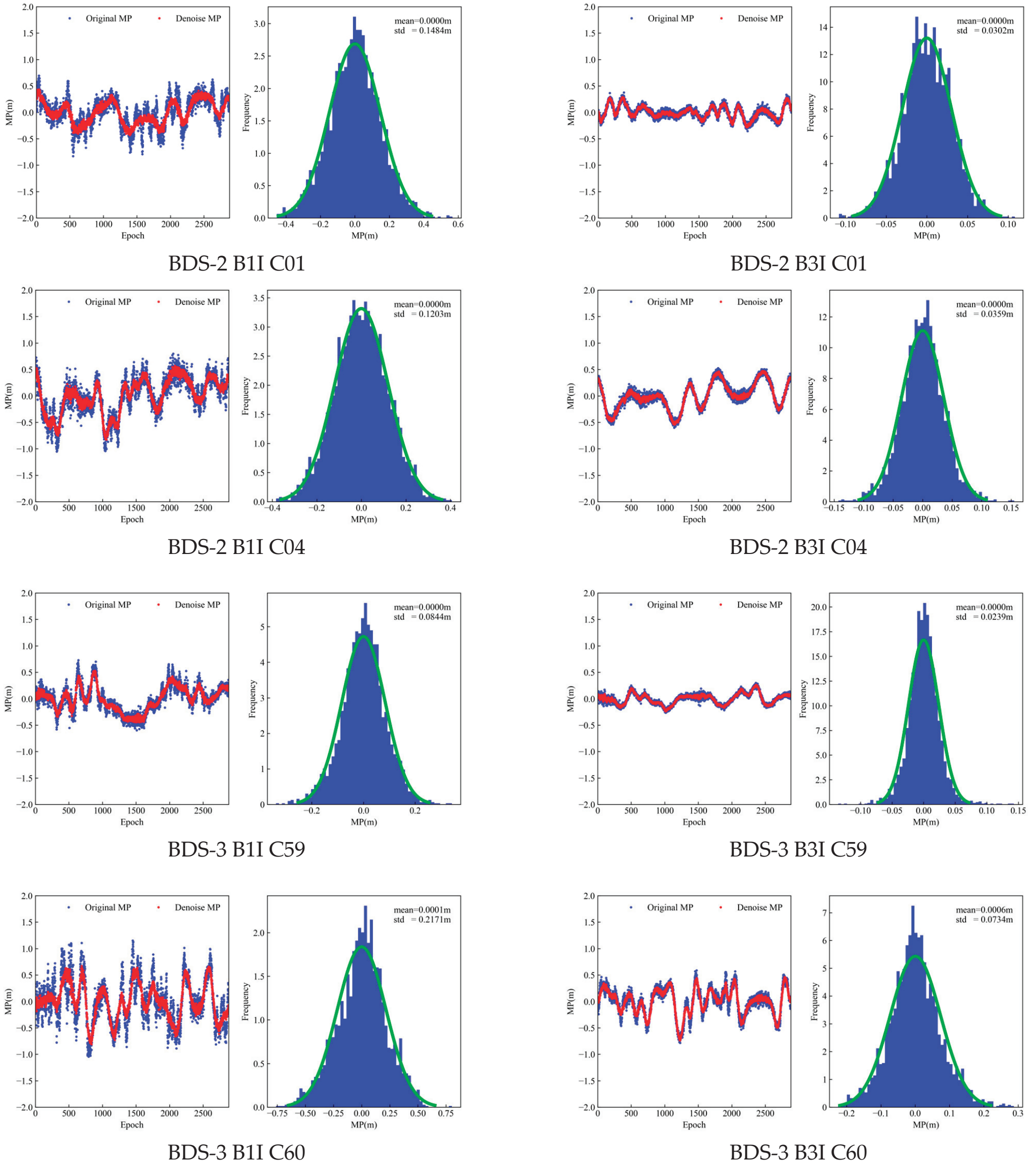


Figure 10. VMD-WT coupled model noise reduction flow chart.

3.4. Analysis of the GEO Correction Results

Figure 11 shows the MP result of C01, C04, C59, and C60 processed by VMD-WT with ALIC at DOY 200. The left side shows the original MP and the denoised MP. The right side shows the residual histogram of MP. The corrected residuals approximate normal distribution and are very close to white noise. Table 4 provides the detailed improvement ratio before and after the correction. The Root Mean Square Error (RMSE) of the modified MP time series decreased significantly and the RMSE of B1I, B2I, and B3I decreased by 53.35%, 63.50%, and 64.71%, respectively. Figure 12 shows the uncorrected and corrected MP RMSE of six GEO satellites at all stations (C05 was not corrected due to missing data at some stations).



BDS-2 B1I C01

BDS-2 B3I C01

BDS-2 B1I C04

BDS-2 B3I C04

BDS-3 B1I C59

BDS-3 B3I C59

BDS-3 B1I C60

BDS-3 B3I C60

Figure 11. MP time series and residual histogram (ALIC DOY 200).

Table 4. Reduction percentage of the MP (%).

Frequency	C01	C02	C03	C04	C05	C59	C60	C61	Average
B1I	60.6	50.6	28.9	65.9	47.9	66.0	50.7	56.2	53.35
B2I	69.0	57.8	51.8	71.9	67.0				63.50
B3I	69.9	64.5	62.5	79.5	65.0	70.5	64.7	41.1	64.71

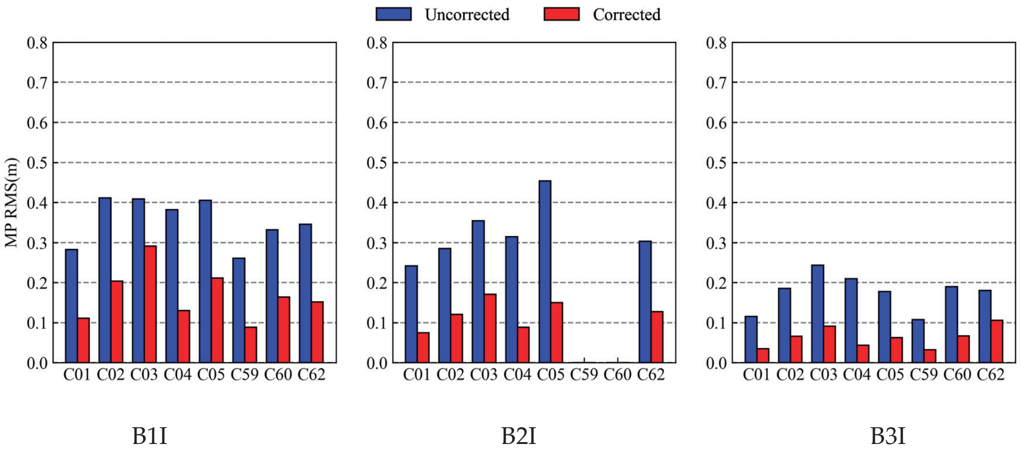


Figure 12. Comparison of the RMSE of the MP after uncorrected and corrected GEO satellites.

Figures 11 and 12 show that VMD-WT can perform better noise reduction on GEO satellites and the residual error after the correction is basically in a normal distribution, with an average increase of about 60%. At the same time, it was found that the multipath of the BDS-3 satellite was different from that of the BDS-2 satellite.

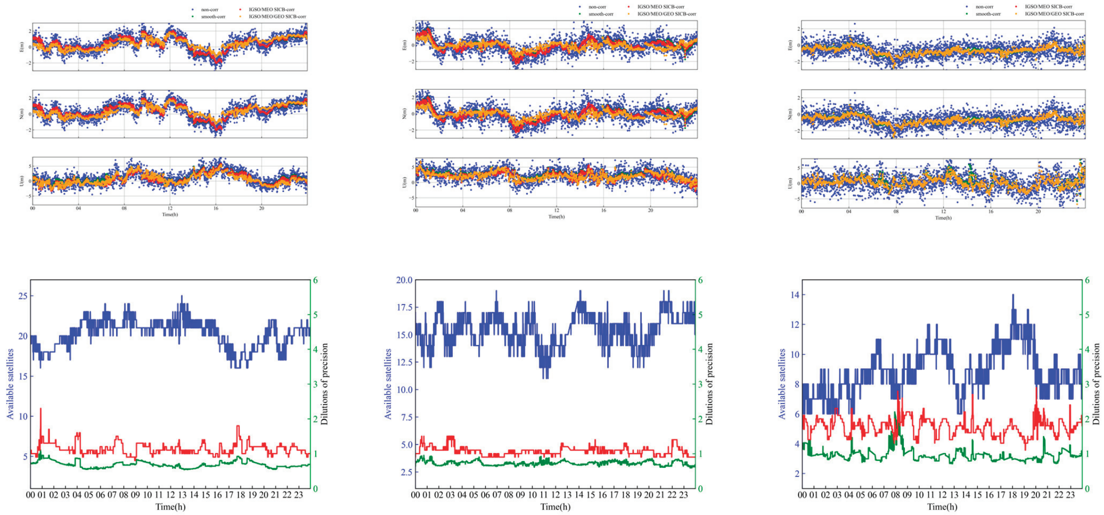
3.5. Correction of the SICB to Improve the CSC of the BDS

To verify the effectiveness of the algorithm, different time periods from the SICB modeling data were selected. The experiment was conducted based on the experimental data from 6 stations, ALIC, KRGG, KOUR, GCGO, GAMG, and SGOC, from 19 July 2023 to 29 July 2023. Four strategies were used for BDS SPP: Scheme 1: Ionosphere-free single point positioning (IF SPP), Scheme 2: ionosphere-free CSC single point positioning (IF CSC SPP), Scheme 3: CSC single point positioning with the IGSO/MEO SICB Correction based on the TOCFA Method (IGSO/MEO SICB CSC), and Scheme 4: CSC single point positioning with the IGSO/MEO/GEO SICB correction based on VMD-WT and TOCFA (IGSO/MEO/GEO SICB CSC). The processing strategy is shown in Table 5.

Table 5. Processing strategies of the SPP for the BDS.

Parameters	Models
Observations	Raw observations
Signal	BDS B1I/B3I
Sampling rate	30 s
Cycle slip detection	GF and MW
Smooth window	20 (min)
Smoothing factor	20
Elevation cutoff	10°
Relativistic effect	Corrected
Ionospheric delay	Ionosphere-free combination
Tropospheric delay	Saastamoinen
Estimator	Weighted least squares

The E, N, and U direction residuals, the number of available satellites, Horizontal Dilution of Precision (HDOP), and Vertical Dilution of Precision (VDOP) of the three stations are shown in Figure 13. The weekly solution document coordinates were used as reference values. The satellite elevation angle and pseudorange residuals are shown in Figure 14. Table 6 shows the RMSE in the E, N, and U directions of four different schemes for 11 consecutive days and the improvement ratio of the current scheme compared to the previous scheme.



ALIC

SGOC

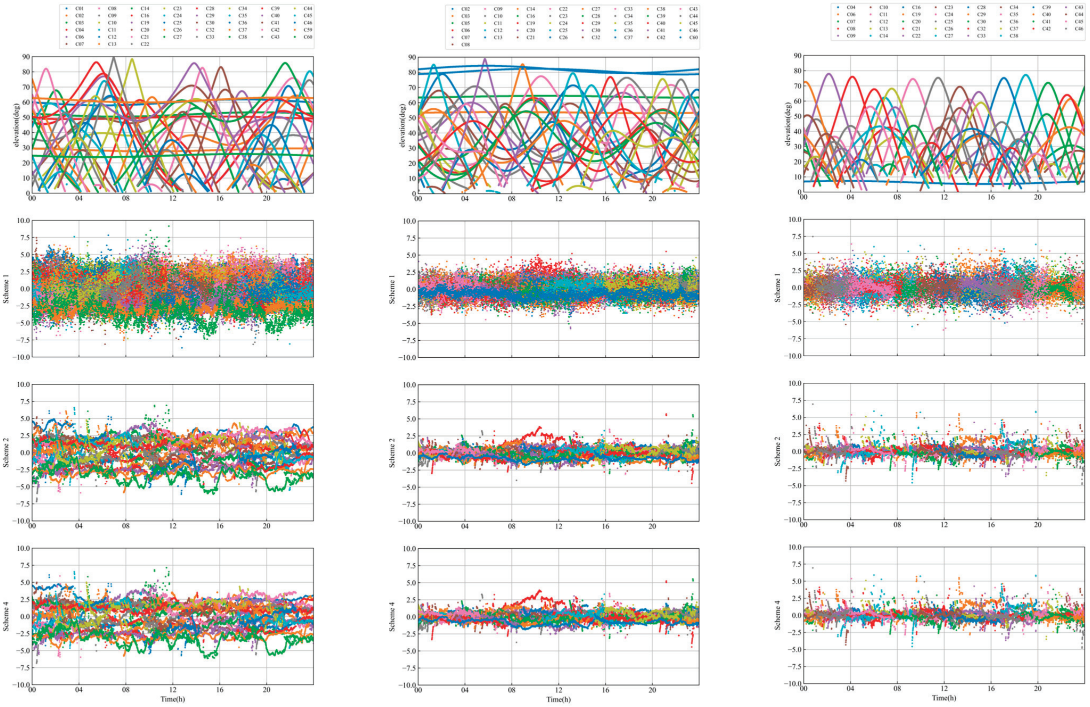
GCGO

Figure 13. The residual of the three stations in the E, N, and U directions and the number of available satellites, HDOP, and VDOP on 19 July 2023 (number of satellites in blue, VDOP in red, HSOP in green).

Figures 13 and 14 and Table 6 show that the positioning accuracy gradually improves with the addition of the IGSO/MEO/GEO SICB correction. The improvement effect was evident in the U direction. At the same time, SICB negatively impacts the accuracy of the E direction because SICB greatly impacts on the U direction more than the E direction, which is consistent with the characteristics of SICB in previous studies.

In the case of only adding the IGSO/MEO SICB correction, the positioning accuracy of the ALIC and SGOC stations has the best improvement. Figure 14 shows that the elevation angles of the satellites of the two stations are generally higher, so the pseudorange was greatly affected by SICB. The effect of the SICB correction is remarkable. Due to the difference in elevation of IGSO/MEO satellites at the KOUR and GCGO stations, the improvement effect was also different. The 3D RMSE of the two stations was reduced to 7.824% and 8.516%, respectively.

After adding the GEO SICB correction, the positioning accuracy of the E and U directions was improved. The highest improvement was the ALIC station and the E, N, and U directions were improved by 14.762%, 6.765%, and 6.880%, respectively, due to the fact that the ALIC station observes more GEO satellites. The improvement effect of the KRGG station was the worst, with an increase of 14.762%, 6.765%, and 6.880% in the E, N, and U directions, respectively. This is due to the lack of available GEO satellites at the KRGG station. Since KOUR and GCGO stations are not located in Asia or Australia, GEO satellites cannot be observed or are below the cut-off satellite elevation angle, so Scheme 4 has no change compared with Scheme 3. The SICB correction of GEO satellites was significant when multiple GEO satellites are observed.



ALIC SGOC GCGO

Figure 14. Satellite elevation angle of the three stations on 19 July 2023 and pseudo-range residuals of Scheme 1, Scheme 2, and Scheme 4 (since the residual values of Scheme 3 and Scheme 4 are close, the result of Scheme 3 is not shown).

Table 6. Positioning errors for the BDS SPP with four different strategies.

Station		RMSE				Improvement Rate#-Previous (%)			
		E	N	U	3D	E	N	U	3D
ALIC	Scheme 1	0.939	0.770	2.210	2.521				
	Scheme 2	0.784	0.579	1.751	2.004	16.414	24.774	20.749	20.498
	Scheme 3	0.807	0.570	1.569	1.854	−2.893	1.623	10.406	7.496
	Scheme 4	0.688	0.531	1.461	1.700	14.762	6.765	6.880	8.310
KRGG	Scheme 1	0.579	0.877	1.508	1.838				
	Scheme 2	0.502	0.786	1.346	1.637	13.381	10.333	10.776	10.930
	Scheme 3	0.491	0.772	1.207	1.515	2.228	1.812	10.264	7.478
	Scheme 4	0.486	0.791	1.199	1.507	0.866	−2.511	0.681	0.522
KOUR	Scheme 1	1.210	0.896	2.871	3.242				
	Scheme 2	1.043	0.733	2.547	2.849	13.741	18.239	11.275	12.129
	Scheme 3	1.032	0.723	2.501	2.800	1.132	1.330	1.836	1.707
	Scheme 4	1.032	0.723	2.501	2.800	0.000	0.000	0.000	0.000
GCGO	Scheme 1	1.019	1.312	3.083	3.502				
	Scheme 2	0.764	0.714	2.133	2.375	25.041	45.591	30.826	32.179
	Scheme 3	0.777	0.691	1.927	2.189	−1.721	3.235	9.657	7.824
	Scheme 4	0.777	0.691	1.927	2.189	0.000	0.000	0.000	0.000
GAMG	Scheme 1	0.735	0.706	2.324	2.538				
	Scheme 2	0.609	0.558	2.087	2.245	17.044	20.992	10.189	11.538
	Scheme 3	0.635	0.563	1.870	2.054	−4.207	−1.010	10.411	8.516
	Scheme 4	0.593	0.500	1.820	1.978	6.579	11.186	2.682	3.664

Table 6. *Cont.*

Station		**RMSE**				**Improvement Rate#-Previous (%)**			
		E	**N**	**U**	**3D**	**E**	**N**	**U**	**3D**
SGOC	Scheme 1	0.964	0.789	3.242	3.473				
	Scheme 2	0.714	0.495	2.819	2.949	25.980	37.247	13.064	15.084
	Scheme 3	0.679	0.468	2.515	2.647	4.857	5.553	10.762	10.255
	Scheme 4	0.620	0.459	2.468	2.585	8.731	1.983	1.885	2.324

In the case of the IGSO/MEO/GEO satellite SICB correction, the average improvement rate between Scheme 4 and Scheme 2 in the E, N, and U directions was 7.03%, 6.49%, and 10.48%, respectively. The IGSO/MEO/GEO CSC algorithm can improve the positioning accuracy by 9%~10% compared to the traditional IF SPP CSC algorithm.

In summary, the improvement effect of SICB is related to the elevation angle of the satellite and indirectly to the station’s position. Under normal circumstances, the accuracy after the IGSO/MEO SICB correction can be increased by 7% to 10% and the accuracy after the GEO SICB correction can be increased by 2% to 8%.

In particular, we also analyze the effect of SICB on the BDS B1/B3 ionosphere-free combination PPP model and find that the impact was not significant. Therefore, the calculation process and results are not given in this paper.

4. Conclusions

This paper reviews the basic principles of CSC and performs error analysis. Through the correlation analysis between MP and elevation angle, the MP of BDS-2 IGSO/MEO has a weak negative correlation with the elevation angle of the satellite, which indirectly indicates that there was a systematic bias between the pseudorange observation and the carrier observation of the BDS 2 IGSO/MEO satellite; that is, SICB. The proposed TOCFM is slightly better than the methods in reference [28]. The IGSO/MEO piecewise weighted least squares TOCFM SICB correction model was established by acquiring the MP time series from 30 stations worldwide. For GEO satellite SICB, a variational mode decomposition combined with the wavelet transform (VMD-WT) method is proposed. The MP time series RMSE of GEO B1I, B2I, and B3I are reduced by 53.35%, 63.50%, and 64.71%, respectively. Based on the above, we propose the CSC algorithm, considering that IGSO/MEO/GEO SICB is proposed in this paper.

In order to verify the effectiveness of the SICB correction, the effects on IF SPP CSC before and after the SICB correction were analyzed and compared. The results show that the SICB correction model effectively weakens the SICB and improves the positioning accuracy of CSC SPP. The accuracy after the IGSO/MEO SICB correction can be increased by 7%~10%, and the accuracy after the GEO SICB correction can be increased by 2%~8%. Therefore, The SICB plays an important role in improving the positioning accuracy. The SICB correction algorithm applies to all smooth pseudorange algorithms and only the dual-frequency IF-CSC algorithm is computed in this paper.

However, there are some shortcomings. It should be noted that the accuracy improvement may be limited in challenging environments due to the influence of cycle slips. Additionally, the model of IGSO/MEO SICB has yet to be modeled for each satellite, leaving room for further improvement in the positioning accuracy. Based on previous research, the characteristics of the different GEO satellite SICB is different and the effect of the different denoising strategy is not the same. Therefore, the Doppler smoothing pseudorange and IGSO/MEO SICB accurate modeling will be the focus of further research. Also, since CCL/CSC is widely used in GNSS ionospheric modeling, we will focus on evaluating the application performance of these two methods in the future.

Author Contributions: Methodology, Q.Z. (Qichao Zhang) and X.M.; software, Q.Z. (Qichao Zhang); validation, Y.G.; writing—original draft preparation, Q.Z. (Qingzhi Zhao). and X.M.; writing—review and editing, X.M. and Y.G.; visualization, G.H.; funding acquisition, Y.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (42204034, 42274039).

Data Availability Statement: Not applicable.

Acknowledgments: Special thanks to IGS for the BDS observatory data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yang, Y.X.; Li, J.L.; Xu, J.Y.; Tang, J.; He, H.B. Contribution of the Compass Satellite Navigation System to Global PNT Users. *Chin. Sci. Bull.* **2011**, *56*, 7. [CrossRef]
2. Yang, Y.; Mao, Y.; Sun, B. Basic Performance and Future Developments of BeiDou Global Navigation Satellite System. *Satellite Navigation* **2020**, *1*, 1. [CrossRef]
3. Spilker, J.J., Jr.; Axelrad, P.; Parkinson, B.W.; Enge, P. *Global Positioning System: Theory and Applications*; Axelrad, P., Parkinson, B.W., Enge, P., Spilker, J.S., Jr., Eds.; American Institute of Aeronautics and Astronautics: Washington, DC, USA, 1996; Volume I, ISBN 978-1-56347-106-3.
4. Hatch, R. The Synergism of GPS Code and Carrier Measurements. In Proceedings of the Third International Geodetic Symposium on Satellite Doppler Positioning, Las Cruces, NM, USA, 8–12 February 1982; Volume 2, pp. 1213–1231.
5. Mcgraw, G. Generalized Divergence-Free Carrier Smoothing with Applications to Dual Frequency Differential GPS. *Navigation* **2009**, *56*, 115–122. [CrossRef]
6. Le, A.Q.; Tiberius, C. Single-Frequency Precise Point Positioning with Optimal Filtering. *GPS Solut.* **2006**, *11*, 61–69. [CrossRef]
7. Chen, C.; Chang, G.; Zhang, S.; Chen, G.; Luo, F. New Range Domain Carrier-smoothed Code Filtering with Dual-frequency BDS Data. *Asian J. Control* **2019**, *21*, 1496–1505. [CrossRef]
8. Park, B.; Kee, C. Optimal Hatch Filter with a Flexible Smoothing Window Width. In Proceedings of the 18th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2005), Long Beach, CA, USA, 13–16 September 2005; pp. 592–602.
9. Park, B.; Lim, C.; Yun, Y.; Kim, E.; Kee, C. Optimal Divergence-Free Hatch Filter for GNSS Single-Frequency Measurement. *Sensors* **2017**, *17*, 448. [CrossRef] [PubMed]
10. Zhang, X.; Huang, P. Optimal Hatch Filter with an Adaptive Smoothing Time Based on SBAS. In Proceedings of the 2nd International Conference on Soft Computing in Information Communication Technology, Taipei, Taiwan, 31 May–1 June 2014; Atlantis Press: Taipei, Taiwan, 2014.
11. Guo, J.; Ou, J.; Yuan, Y.; Wang, H. Optimal Carrier-Smoothed-Code Algorithm for Dual-Frequency GPS Data. *Prog. Nat. Sci.* **2008**, *18*, 591–594. [CrossRef]
12. Liu, X.; Yang, S.; Yang, J.; Lu, P.; University, S. Optimal Doppler-Aided Autonomous Position with a Flexible Smoothing Window Width. In Proceedings of the 26th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2013), Nashville, TN, USA, 16–20 September 2013; pp. 3521–3527.
13. Zhou, Z.; Li, B. Optimal Doppler-Aided Smoothing Strategy for GNSS Navigation. *GPS Solut.* **2017**, *21*, 197–210. [CrossRef]
14. Zhang, Q.; Chen, Z.; Rong, F.; Cui, Y. An Improved Hatch Filter and Its Application in Kinematic Positioning with Single-Frequency GPS. *Measurement* **2019**, *146*, 868–878. [CrossRef]
15. Chen, C.; Chang, G.; Luo, F.; Zhang, S. Dual-Frequency Carrier Smoothed Code Filtering with Dynamical Ionospheric Delay Modeling. *Adv. Space Res.* **2019**, *63*, 857–870. [CrossRef]
16. Tang, W.; Cui, J.; Hui, M.; Deng, C. Performance Analysis for BDS Phase-Smoothed Pseudorange Differential Positioning. *J. Navig.* **2016**, *69*, 1011–1023. [CrossRef]
17. Bahrami, M.; Ziebart, M. A Kalman Filter-Based Doppler-Smoothing of Code Pseudoranges in GNSS-Challenged Environments. In Proceedings of the 24th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2011), Portland, OR, USA, 20–23 September 2011; pp. 2362–2372.
18. Geng, J.; Jiang, E.; Li, G.; Xin, S.; Wei, N. An Improved Hatch Filter Algorithm towards Sub-Meter Positioning Using Only Android Raw GNSS Measurements without External Augmentation Corrections. *Remote Sens.* **2019**, *11*, 1679. [CrossRef]
19. Lou, Y.; Gong, X.; Gu, S.; Zheng, F.; Feng, Y. Assessment of Code Bias Variations of BDS Triple-Frequency Signals and Their Impacts on Ambiguity Resolution for Long Baselines. *GPS Solut.* **2017**, *21*, 177–186. [CrossRef]
20. Hauschild, A.; Montenbruck, O.; Sleewaegen, J.-M.; Huisman, L.; Teunissen, P.J.G. Characterization of Compass M-1 Signals. *GPS Solut.* **2012**, *16*, 117–126. [CrossRef]
21. Guo, F.; Li, X.; Liu, W. Mitigating BeiDou Satellite-Induced Code Bias: Taking into Account the Stochastic Model of Corrections. *Sensors* **2016**, *16*, 909. [CrossRef]

22. Pan, L.; Guo, F.; Ma, F. An Improved BDS Satellite-Induced Code Bias Correction Model Considering the Consistency of Multipath Combinations. *Remote Sens.* **2018**, *10*, 1189. [CrossRef]
23. Wu, X.; Zhou, J.; Wang, G.; Hu, X.; Cao, Y. Multipath Error Detection and Correction for GEO/IGSO Satellites. *Sci. China Phys. Mech. Astron.* **2012**, *55*, 1297–1306. [CrossRef]
24. Ning, Y.; Yuan, Y.; Chai, Y.; Huang, Y. Analysis of the Bias on the Beidou GEO Multipath Combinations. *Sensors* **2016**, *16*, 1252. [CrossRef]
25. Hu, C.; Wang, Z.; Rao, P.; Cheng, T. One-Step Correction Strategy for BDS-2/BDS-3 Satellite Observation Code Bias and Multipath Delay. *Acta Geod. Geophys.* **2021**, *56*, 29–59. [CrossRef]
26. Chen, J.; Yue, D.; Zhu, S.; Chen, H.; Liu, Z.; Zhao, X. Correction Model of BDS Satellite-Induced Code Bias and Its Impact on Precise Point Positioning. *Adv. Space Res.* **2019**, *63*, 2155–2163. [CrossRef]
27. Ge, Y.; Zhou, F.; Sun, B.; Wang, S.; Shi, B. The Impact of Satellite Time Group Delay and Inter-Frequency Differential Code Bias Corrections on Multi-GNSS Combined Positioning. *Sensors* **2017**, *17*, 602. [CrossRef] [PubMed]
28. Wanninger, L.; Beer, S. BeiDou Satellite-Induced Code Pseudorange Variations: Diagnosis and Therapy. *GPS Solut.* **2015**, *19*, 639–648. [CrossRef]
29. Dragomiretskiy, K.; Zosso, D. Variational Mode Decomposition. *IEEE Trans. Signal Process.* **2014**, *62*, 531–544. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

remote sensing

MDPI

Technical Note

Multi-Antenna Global Navigation Satellite System/Inertial Measurement Unit Tight Integration for Measuring Displacement and Vibration in Structural Health Monitoring

Wujiao Dai [1,*], Xin Li [1], Wenkun Yu [1], Xuanyu Qu [2] and Xiaoli Ding [2]

[1] School of Geosciences and Info-Physics, Central South University, Changsha 410083, China
[2] Department of Land Surveying & Geo-Informatics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China; xl.ding@polyu.edu.hk (X.D.)
* Correspondence: wjdai@csu.edu.cn

Abstract: Large-scale engineering structures deform and vibrate under the influence of external forces. Obtaining displacement and vibration is crucial for structural health monitoring (SHM). Global navigation satellite system (GNSS) and inertial measurement unit (IMU) are complementary and widely used in SHM. In this paper, we propose an SHM scheme where IMU and multi-antenna GNSS are tightly integrated. The phase centers of multiple GNSS antennas are transformed into the IMU center, which increases the observation redundancy and strengthens the positioning model. To evaluate the performance of tight integration of IMU and multiple GNSS antennas, high-rate vibrational signals are simulated using a shaking table, and the errors of horizontal displacement of different positioning schemes are analyzed using recordings of a high-precision ranging laser as the reference. The results demonstrate that applying triple-antenna GNSS/IMU integration for measuring the displacement can achieve an accuracy of 2.6 mm, which is about 33.0% and 30.3% superior than the accuracy achieved by the conventional single-antenna GNSS-only and GNSS/IMU solutions, respectively.

Keywords: multi-antenna GNSS; GNSS/IMU; tight integration; structural health monitoring

Citation: Dai, W.; Li, X.; Yu, W.; Qu, X.; Ding, X. Multi-Antenna Global Navigation Satellite System/Inertial Measurement Unit Tight Integration for Measuring Displacement and Vibration in Structural Health Monitoring. *Remote Sens.* **2024**, *16*, 72. https://doi.org/10.3390/rs16061072

Academic Editor: Baocheng Zhang

Received: 19 February 2024
Revised: 15 March 2024
Accepted: 15 March 2024
Published: 18 March 2024

1. Introduction

For large-scale structures, serious structural damage may occur when vibrating deformation induced by external loads (such as traffic and winds) exceeds the designed bearing capacity. Accurate deformation information is a prerequisite for timely alerts to prevent unnecessary casualties and losses. SHM is a technology for obtaining deformation information and estimating the health condition and the structural characteristics of civil structures and infrastructures [1,2]. On this basis, an SHM strategy first intends to measure responses such as displacements over time, and then obtain insightful information about the current or unknown condition of a civil structure by computational techniques either in time domain or in frequency domain [3,4]. Various types of sensor devices such as linear variable differential transformers, optical fiber sensors, smartphones, vision cameras, and radars are used in SHM to measure the displacement responses, but dynamic deformation monitoring widely adopts GNSS and accelerometers for SHM non-stop or with high periodicity but not requiring gluing/embedding sensors into the structure [5–8]. The RTK-GNSS can achieve a subcentimeter-level measurement accuracy, and it is often selected for structural displacement estimation of large-scale structures such as long-span bridges and high-rise buildings, which usually have at least centimeter-level displacements [9–11]. However, the accuracy of GNSS positioning is typically limited due to environmental perturbations such as occlusion, diffraction, and reflection. In addition, the GNSS-based monitoring method is insensitive to high-frequency vibration signals because the measurement noise is relatively large [12]. The accelerometer method can obtain high-accuracy dynamical displacements over short time periods through the double integration of acceleration measurements. However, significant cumulative errors are ultimately present when the accelerometer is applied to monitor long-period quasi-static displacements [13].

GNSS and accelerometers are complementary and therefore have been commonly integrated in monitoring applications. For example, Meng et al. [14] successfully identified a peak vibration frequency of 10.05 Hz of Nottingham Wilford Bridge by using an GNSS/accelerometer integrated system. Kim et al. [15] combined GPS RTK with an accelerometer in monitoring Yeongjong Grand Bridge and achieved a displacement accuracy of 2 mm in the vertical direction. Xin et al. [16] integrated a strong motion accelerometer with GNSS PPP for seismic deformation monitoring. The root-mean-square error (RMSE) of the difference with respect to the reference was 2 mm, with a cross-correlation coefficient of 0.99. Although GNSS/accelerometer integration has been widely validated in previous studies, the tilt of the accelerometer can cause baseline errors and distort the captured peak amplitude of the displacement signal [14,16].

In addition to displacement, rotational deformation is another critical piece of information for structural health monitoring. Although the accelerometer can measure the rotational deformation by sensing changes in gravitational acceleration [17], the accuracy of accelerometer-derived rotation is typically limited because accelerometers measure specific forces that involve gravitational and vibrational acceleration. To separate the different types of acceleration and finally determine the rotational deformation, it is necessary to use gyroscopes [18]. Rossi et al. [19] corrected the impact of rotation on GNSS/accelerometer monitoring results using rotation information recorded using an inertial measurement unit (IMU). The RMSE of the corrected displacement monitoring results was reduced by half to about 1–2 mm. de Alteriis et al. [20] designed a low-cost and real-time monitoring device that integrated GNSS and Micro Electro Mechanical System (MEMS) IMU. Experimental results showed that the system was able to estimate position and attitude under high-frequency vibrations, and the monitoring results were in good agreement with the fiber-optic sensors. Geng et al. [21] implemented a seismic monitoring system with six degrees of freedom by integrating an accelerometer, gyroscope, and GNSS. In a waveform simulation experiment of the 2010 MW 7.2 EI Mayor-Cucapah earthquake, the displacement obtained using the proposed method achieved an accuracy 68% higher than that obtained from the traditional accelerometer/GNSS integration.

Most previous studies of integrated GNSS/IMU deformation monitoring are based on single-antenna GNSS. Although improved accuracy and robustness have been reported, there are still some obvious limitations of single-antenna GNSS/IMU. For example, the GNSS signal environment in deformation monitoring is typically complex, which worsens the GNSS signal quality and ultimately the positioning accuracy. Moreover, the motion of the monitored structures is generally limited, resulting in a low attitude observability, especially in the heading direction [22,23]. The accumulation of attitude errors eventually affects the monitoring accuracy, especially when low-cost IMUs are used [24]. Multi-antenna GNSS offers a solution to this problem. Double-difference carrier phases are formed between multiple GNSS antennas to estimate inter-antenna baselines. When carrier phase ambiguities are fixed, the baselines can be accurately inverted to attitude information [25]. Zhang et al. [26] evaluated the positioning performance of the traditional single-antenna and dual-antenna GNSS/IMU. They reported that the dual-antenna GNSS/IMU achieved mm level positioning accuracy, which was about 50% higher than the single-antenna GNSS/IMU. Li et al. [27] applied the length constraints of lever arms in multi-antenna GNSS/IMU for monitoring the settlement of high-speed railway, and achieved a positioning accuracy of 1–2 mm. However, multi-antenna GNSS/IMU integration has been rarely studied in structural deformation monitoring.

In this paper, we propose a tight integration of multi-antenna GNSS and IMU for vibration monitoring. The GNSS observations are tightly coupled to the IMU by transforming their phase centers to the IMU center. Thus, the integration reduces the number of unknown parameters, enhances the geometric strength of the model, and increases the observability of attitude errors. To evaluate the performance of the proposed method, different types of vibrational signals are simulated using a shaking table and a laser rangefinder recording as a reference. Section 2 presents the mathematical model of the tight integration of multi-antenna GNSS and IMU, including the measurement model and the system model. In Section 3, the experimental configuration, results, and analysis are described. Section 4 concludes this work.

2. Methods

2.1. Multi-Antenna GNSS/IMU Tight Integration Measurement Model

The measurement model of GNSS/IMU tight integration can be expressed as:

$$\delta\boldsymbol{z} = \boldsymbol{H}\delta\boldsymbol{x} + \boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim N(\boldsymbol{0},\ \boldsymbol{R}) \tag{1}$$

where $\delta\{\cdot\}$ indicates the correction for a variable; $\delta\boldsymbol{z}$ denotes the measurement correction vector that is the difference between GNSS observations (including carrier phases and pseudorange rates) and those derived from IMU; $\boldsymbol{H}$ is the design matrix; $\delta\boldsymbol{x}$ is the state correction vector; $\boldsymbol{\epsilon}$ represents the observation noise vector; and $\boldsymbol{R}$ corresponds to the covariance matrix of observations. The state vector $\boldsymbol{x}$ is:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{r}^T & \boldsymbol{v}^T & \boldsymbol{\psi}^T & \boldsymbol{b}_a{}^T & \boldsymbol{b}_g{}^T & \boldsymbol{n}^T \end{bmatrix}^T \tag{2}$$

where $\boldsymbol{r}$ is the three-dimensional position vector, taking the IMU reference center as the reference point of the platform; $\boldsymbol{v}$ and $\boldsymbol{\psi}$, respectively, indicate the velocity and attitude of the platform; $\boldsymbol{b}_a$ and $\boldsymbol{b}_g$ denote the biases of accelerometer and gyroscope, respectively; and $\boldsymbol{n}$ corresponds to the ambiguities of double-difference carrier phases.

When composing the measurement model for the multi-antenna GNSS and IMU tight integration system, one rover antenna is chosen as the primary antenna and the others are used as auxiliary antennas. The double-difference observation equation is formed between the primary antenna and the reference station, which introduces an absolute position for the whole system. At the same time, the double-difference observation equations are also formed between the primary antenna and the auxiliary antennas to adequately exploit the redundant observations from multiple antennas. The double-difference carrier phase $\boldsymbol{z}_{ij}$ and pseudorange rate $\dot{\boldsymbol{z}}_{ij}$ between antenna i and antenna j can be written as:

$$\begin{cases} \boldsymbol{z}_{ij} = \boldsymbol{A}_i\boldsymbol{r}_i - \boldsymbol{A}_j\boldsymbol{r}_j + \boldsymbol{\Lambda}_{ij}n + \boldsymbol{\epsilon}_{ij} \\ \quad \dot{\boldsymbol{z}}_{ij} = \boldsymbol{A}_i\boldsymbol{v}_i - \boldsymbol{A}_j\boldsymbol{v}_j + \dot{\boldsymbol{\epsilon}}_{\boldsymbol{ij}} \end{cases} \tag{3}$$

where $\boldsymbol{r}_i$ and $\boldsymbol{v}_i$ are, respectively, the position and velocity of antenna i; $\boldsymbol{A}_i$ stands for an $m \times 3$ line-of-sight matrix of between-satellite single-difference at antenna i, where m is the difference between the total number of satellites observed and the number of satellite systems employed; $\boldsymbol{\Lambda}_{ij}$ is a diagonal matrix with non-zero elements being the carrier phase wavelengths of the corresponding satellites; and $\boldsymbol{\epsilon}_{ij}$ denotes the double-difference observation noise. The position vector $\boldsymbol{r}_i$ and velocity vector $\boldsymbol{v}_i$ of antenna i can be transformed to the platform position vector $\boldsymbol{r}$ and velocity vector $\boldsymbol{v}$ at the IMU center through the lever-arm vector and rotation matrix [28]:

$$\boldsymbol{r}_i = \boldsymbol{r} + \boldsymbol{C}\boldsymbol{l}_i \tag{4}$$

$$\boldsymbol{v}_i = \boldsymbol{v} + \boldsymbol{C}(\boldsymbol{\Omega}_\omega\boldsymbol{l}_i) + \boldsymbol{\Omega}_e\boldsymbol{C}\boldsymbol{l}_i,\ \boldsymbol{\Omega}_\omega = [\boldsymbol{\omega}\times] \tag{5}$$

where $\boldsymbol{C}$ is the direction cosine matrix, i.e., the rotation matrix; $\boldsymbol{l}_i$ stands for the lever-arm vector of antenna i; $\boldsymbol{\Omega}_\omega$ indicates the skew-symmetric matrix of three-dimensional angular rate obtained by IMU, i.e., $[\boldsymbol{\omega}\times]$; $[\cdot\times]$ represents a skew-symmetric operator; and $\boldsymbol{\Omega}_e$ denotes the skew-symmetric matrix of earth angle rotation speed vector. Inserting (4) and (5) into (3) yields the double-difference carrier phase and pseudorange rate equations expressed by the position vector and velocity vector of the IMU center:

$$\begin{cases} \boldsymbol{z}_{ij} = \boldsymbol{A}_{ij}\boldsymbol{r} + \boldsymbol{A}_i\boldsymbol{C}\boldsymbol{l}_i - \boldsymbol{A}_j\boldsymbol{C}\boldsymbol{l}_j + \boldsymbol{\Lambda}_{ij}\boldsymbol{n} + \boldsymbol{\epsilon}_{ij} \\ \dot{\boldsymbol{z}}_{ij} = \boldsymbol{A}_{ij}\boldsymbol{v} + \boldsymbol{A}_i\left(\boldsymbol{C}\boldsymbol{\Omega}_\omega\boldsymbol{l}_i + \boldsymbol{\Omega}_e\boldsymbol{C}\boldsymbol{l}_i\right) - \boldsymbol{A}_j\left(\boldsymbol{C}\boldsymbol{\Omega}_\omega\boldsymbol{l}_j + \boldsymbol{\Omega}_e\boldsymbol{C}\boldsymbol{l}_j\right) + \dot{\boldsymbol{\epsilon}}_{ij} \end{cases} \tag{6}$$

where $\boldsymbol{A}_{ij}$ is the difference of $\boldsymbol{A}_i$ and $\boldsymbol{A}_j$, i.e., $\boldsymbol{A}_{ij} = \boldsymbol{A}_i - \boldsymbol{A}_j$. Given that

$$\delta(\boldsymbol{C}\boldsymbol{l}_i) = [(\boldsymbol{C}\boldsymbol{l}_i)\times]\delta\boldsymbol{\psi} \tag{7}$$

$$\delta(\boldsymbol{C}\boldsymbol{\Omega}_\omega\boldsymbol{l}_i + \boldsymbol{\Omega}_e\boldsymbol{C}\boldsymbol{l}_i) = [(\boldsymbol{C}\boldsymbol{\Omega}_\omega\boldsymbol{l}_i - \boldsymbol{\Omega}_e\boldsymbol{C}\boldsymbol{l}_i)\times]\delta\boldsymbol{\psi} \tag{8}$$

the design matrix of (1) can be expressed in terms of submatrix as:

$$\boldsymbol{H}_{ij} = \begin{bmatrix} \boldsymbol{A}_{ij} & \boldsymbol{0}_{m\times 3} & \boldsymbol{H}_{r\psi} & \boldsymbol{0}_{m\times 3} & \boldsymbol{0}_{m\times 3} & \boldsymbol{\Lambda}_{ij} \\ \boldsymbol{0}_{m\times 3} & \boldsymbol{A}_{ij} & \boldsymbol{H}_{v\psi} & \boldsymbol{0}_{m\times 3} & \boldsymbol{H}_{vb} & \boldsymbol{0}_{m\times n} \end{bmatrix} \tag{9}$$

where $\boldsymbol{0}_{m\times n}$ stands for an $m \times n$ matrix with zero elements. The specific expressions of $\boldsymbol{H}_{r\psi}$, $\boldsymbol{H}_{v\psi}$, $\boldsymbol{H}_{vb}$ are:

$$\boldsymbol{H}_{r\psi} = \boldsymbol{A}_i[(\boldsymbol{C}\boldsymbol{l}_i)\times] - \boldsymbol{A}_j\left[\left(\boldsymbol{C}\boldsymbol{l}_j\right)\times\right] \tag{10}$$

$$\boldsymbol{H}_{v\psi} = \boldsymbol{A}_i[(\boldsymbol{C}\boldsymbol{\Omega}_\omega\boldsymbol{l}_i - \boldsymbol{\Omega}_e\boldsymbol{C}\boldsymbol{l}_i)\times] - \boldsymbol{A}_j\left[\left(\boldsymbol{C}\boldsymbol{\Omega}_\omega\boldsymbol{l}_j - \boldsymbol{\Omega}_e\boldsymbol{C}\boldsymbol{l}_j\right)\times\right] \tag{11}$$

$$\boldsymbol{H}_{vb} = \boldsymbol{A}_i\boldsymbol{C}[\boldsymbol{l}_i\times] - \boldsymbol{A}_j\boldsymbol{C}\left[\boldsymbol{l}_j\times\right] \tag{12}$$

Note that $\boldsymbol{A}_j = \boldsymbol{0}$ when j indicates an external base station with known coordinates.

2.2. Multi-Antenna GNSS/IMU Tight Integration System Model

The system model of the integration system of GNSS and IMU can be expressed as:

$$\delta\dot{\boldsymbol{x}}(t) = \boldsymbol{F}(t)\delta\boldsymbol{x}(t) + \boldsymbol{G}(t)\boldsymbol{w}(t) \tag{13}$$

where $\boldsymbol{F}(t)$ is the transition matrix of state vector; $\boldsymbol{G}(t)$ stands for the process noise mapping matrix; and $\boldsymbol{w}(t)$ indicates the process noise vector. The reference coordinate frame adopted in this paper is the Earth-Centered Earth-Fixed (ECEF). The sensor bias errors of the accelerometer and gyroscope are modeled as first-order Gauss–Markov stochastic processes. After the discretization of (13) with update interval τ, the state propagation equation can be rewritten as:

$$\delta\boldsymbol{x}_{k+1}^{-} = \boldsymbol{\Phi}_k\delta\boldsymbol{x}_k + \boldsymbol{w}_k \tag{14}$$

where $\boldsymbol{x}_{k+1}^{-}$ is a prediction of state vector of time t_{k+1} based on the one of time t_k, i.e., $\boldsymbol{x}_k$, and the "$-$" indicates a quantity has not been updated using the latest observation; and $\boldsymbol{\Phi}_k$ and $\boldsymbol{w}_k$ are the transition matrix of state vector and the process noise matrix from time t_k to time t_{k+1}, respectively. The first order approximation of $\boldsymbol{\Phi}_k$ can be obtained as follows:

$$\boldsymbol{\Phi} \approx \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{I}_3\tau & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_{3\times n} \\ \boldsymbol{F}_{vp}\tau & \boldsymbol{I}_3 - 2\boldsymbol{\Omega}_e\tau & [-(\boldsymbol{C}\boldsymbol{f})\times] & \boldsymbol{C}\tau & \boldsymbol{0}_3 & \boldsymbol{0}_{3\times n} \\ \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{I}_3 - \boldsymbol{\Omega}_e\tau & \boldsymbol{0}_3 & \boldsymbol{C}\tau & \boldsymbol{0}_{3\times n} \\ \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{F}_{ba} & \boldsymbol{0}_3 & \boldsymbol{0}_{3\times n} \\ \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{F}_{bg} & \boldsymbol{0}_{3\times n} \\ \boldsymbol{0}_{n\times 3} & \boldsymbol{0}_{n\times 3} & \boldsymbol{0}_{n\times 3} & \boldsymbol{0}_{n\times 3} & \boldsymbol{0}_{n\times 3} & \boldsymbol{I}_n \end{bmatrix} \tag{15}$$

$$\boldsymbol{F}_{vp} = -\frac{2\gamma\boldsymbol{r}^T}{r_{es}|\boldsymbol{r}|} \tag{16}$$

$$\begin{cases} \boldsymbol{F}_{ba} = diag\left(e^{-\tau/T_{c,\,ba}}\right) \\ \boldsymbol{F}_{bg} = diag\left(e^{-\tau/T_{c,\,bg}}\right) \end{cases} \tag{17}$$

where r_{es} is the geocentric radius on the Earth surface; γ indicates the gravity vector; and $T_{c,\,ba}$ and $T_{c,\,bg}$ are the correlation time of accelerometer and gyroscope, respectively, which can be obtained using Allan variance analysis.

Figure 1 shows the flowchart of the tight integration of multi-antenna GNSS and IMU. First, raw observations from multiple antennas are processed, with one primary antenna and the others auxiliary. The double-difference observation equations are formed between the antennas and the reference station (equally, between a selected primary antenna and the reference station, as well as the auxiliary antennas). Relative positioning is performed based on the double-difference observations to obtain the position, velocity, and attitude information of the integrated system. The GNSS-derived information can be used for the initial alignment of the IMU when the ambiguities are fixed. The attitude initialization can also be done using INS self-alignment, with the accelerometer leveling to calculate pitch and roll angles, and gyro-compass to calculate heading angles. The position, velocity, and attitude information of the integrated system can then be updated based on the IMU output. The angular rate of the gyro output is first used to update the attitude, based on which the specific force output from the accelerometer is transformed, and then the velocity and position information are updated. Finally, the equations of state and measurements of the integrated system are formulated, and the Kalman filter is applied to estimate the position, velocity, and attitude of the IMU center at each epoch, together with feedback corrections for sensor errors.

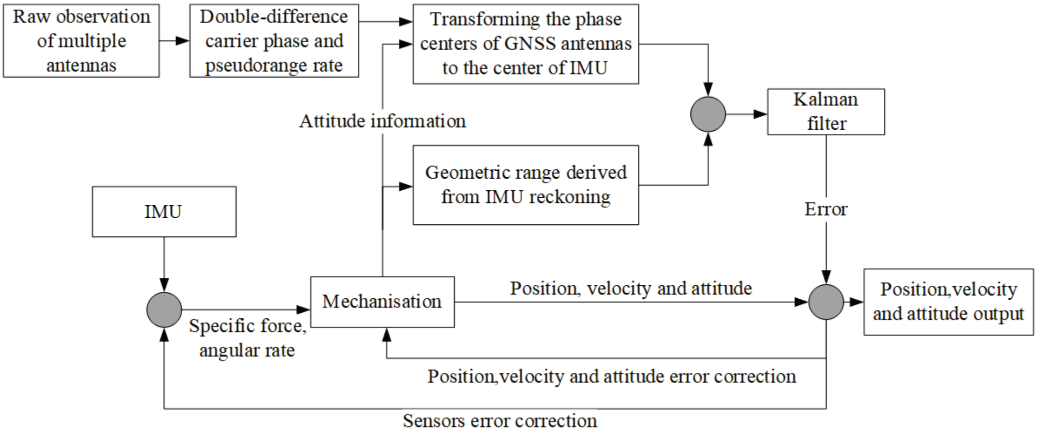
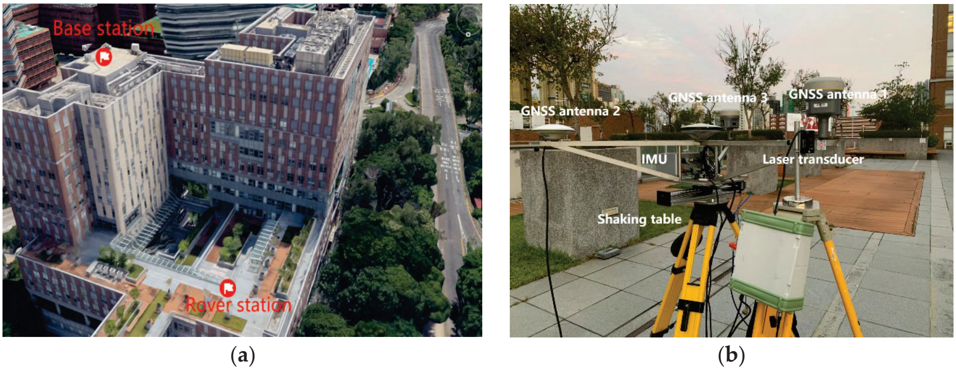

Raw observation of multiple antennas
Double-difference carrier phase and pseudorange rate
Transforming the phase centers of GNSS antennas to the center of IMU
Attitude information
Kalman filter
Geometric range derived from IMU reckoning
IMU
Error
Mechanisation
Position, velocity and attitude
Position,velocity and attitude output
Specific force, angular rate
Position,velocity and attitude error correction
Sensors error correction

Figure 1. Flowchart of the tight integration of multi-antenna GNSS and IMU.

3. Experiments and Analysis

To assess the performance of the tight integration of multi-antenna GNSS and IMU for vibration deformation monitoring, a simulation vibration test was conducted on 13 November 2023 on a half-open platform on the 6th floor of Hong Kong Polytechnic University. Figure 2a illustrates the environment of the experimental site. The monitoring accuracy of integration schemes of different antennas and IMU were analyzed, and the vibration displacement information recorded using a laser rangefinder with a ranging accuracy of 0.01 mm, and a sampling rate of 20 Hz was used as the reference. Figure 2b shows the instrument deployment for the vibration experiment. A shaking table was fixed on a tripod. A rigid triangular platform with one IMU and four GNSS antennas was mounted on the table, and the length of inter-antenna baselines was 1 m. In deformation monitoring, the observability of attitude errors is typically low because of the limited dynamics (including rotation and acceleration) of the deforming objects. The three-dimensional attitude can be determined using at least three non-collinear antennas. Therefore, this study adopts the triple-antenna triangle configuration to strengthen the GNSS/IMU positioning model so that the estimation of parameters such as attitude and IMU biases can be enhanced. A Leica GR25 GNSS geodetic receiver was attached to antenna 2, two Trimble R12 GNSS geodetic receivers were placed for rover antennas 1 and 3, respectively, and a Trimble Net R9 geodetic receiver was used for the base station. We use the configuration of the three non-collinear antennas to enhance the three-dimensional attitude determination, considering that the observability of attitude is typically low due to the limited dynamics of the deforming objects. The base station was located on the roof of the building at a distance of about 50 m. The GNSS sampling rate was 20 Hz. The IMU adopted was iXBLUE ATLANS-C, which has triple-axis accelerometers and gyroscopes, and a self-contained GNSS receiver for time synchronization (the corresponding antenna was installed at the center of the triangular platform). The error characteristics of IMU were estimated using Allan variance analysis [29], as shown in Table 1.



(**a**) (**b**)

Figure 2. Experiment setup. (**a**) Environment of experimental site; (**b**) setup of experimental equipment.

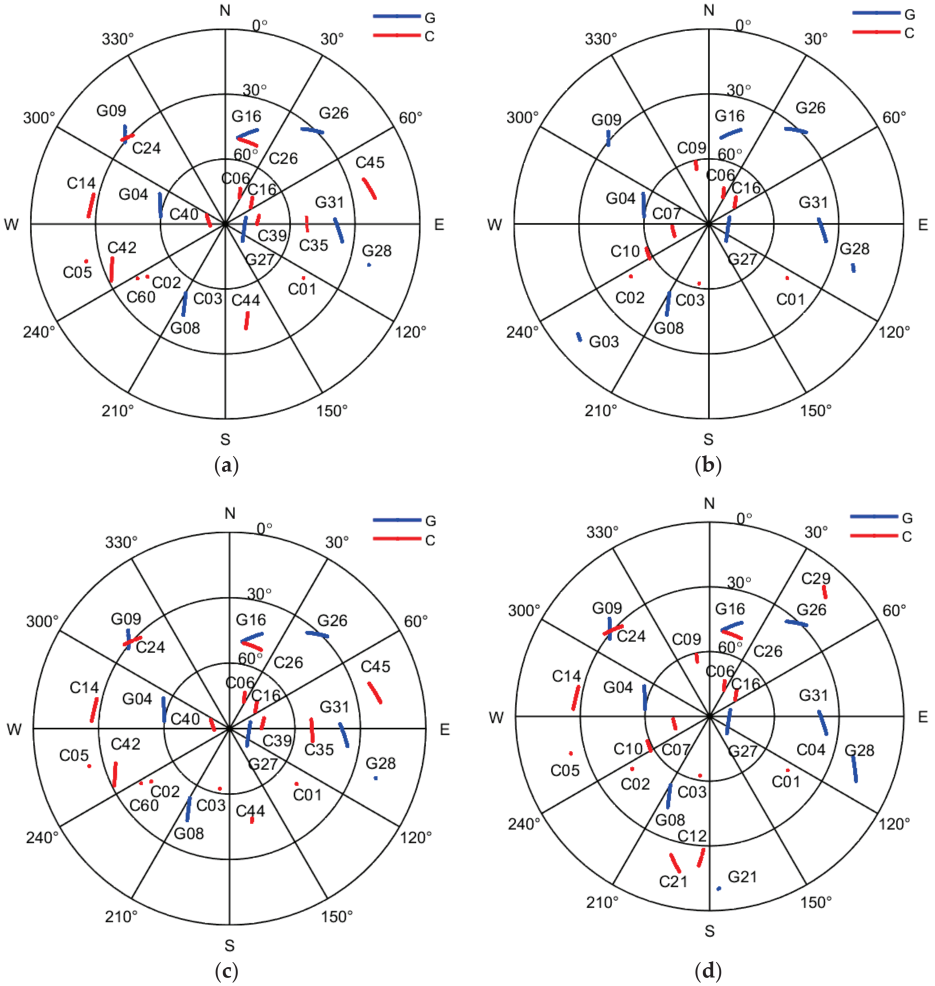
Table 1. Error characteristics of IMU.

Error Characteristics	Accelerometer	Gyroscope
Bias instability	0.0241 mg	0.0084 deg/h
Correlation time	477 s	2957 s
Random walk	0.046 mg/sqrt (Hz)	0.0035 deg/sqrt (h)

A suite of C++ libraries was developed for the data processing, where IMU records are tightly integrated with GNSS observations in double-difference relative kinematic positioning. The double-difference observation equations are formed between the primary antenna and the reference station, as well as the auxiliary antennas. GPS and BDS observations were used for positioning in the experiments. In short baselines, the frequency-dependent ionospheric delays are considerably canceled, and therefore only single-frequency (L1/B1) observations were used in the experiment to reduce the computation of high-rate data. The cut-off angle was set to 10 degrees, and observations were weighted using an elevation-based cosine function.

Figure 3 shows the sky distribution of the observed GPS and BDS satellites, where G denotes the GPS satellite and C the BDS satellite. It can be seen that during the test, about 8–9 GPS satellites and 16–17 BDS satellites were observed by Antennas 1, 3, and the base station. Satellite signals from the southeast direction with low elevation angles (below 30 degrees) were not received due to the blocking of nearby buildings (Figure 2). Note that

Antenna 2 could observe 9 GPS satellites but only 8 BDS-2 satellites because the firmware of the receiver was outdated.



(**a**)

(**b**)

(**c**)

(**d**)

Figure 3. Sky map of GPS and BDS satellites observed by different antennas. (**a**) Antenna 1; (**b**) Antenna 2; (**c**) Antenna 3; and (**d**) antenna of base station.

Table 2 lists eleven positioning schemes that were tested in our experiment, including three single-antenna GNSS-only schemes, one triple-antenna GNSS-only scheme (using the average of the three single-antenna GNSS-only solutions), and seven GNSS/IMU integration schemes with different antennas.

Figure 4 shows the 2D horizontal displacement time series recorded with the laser rangefinder, and the horizontal displacements derived by relative positioning using schemes G1, G2, and G3. As shown in Figure 4a, the shaking table was first kept stationary for about 10 min, which was referred to as Period 1, then generated two 1 min horizontal vibrations with frequencies of 0.50 Hz and 0.67 Hz and amplitudes of 19 mm and 17 mm, respectively. After that, a vibration with a mixed frequency of 0.11 Hz and 0.34 Hz was loaded to the shaking table for about 4 min. Then the shaking table generated a vibration with a frequency of 0.80 Hz and an amplitude of 15 mm. At last, the shaking table returned to the original position in 4 min and then repeated the vibration process. The whole vibration process was named Period 2. We can see that all eight vibration events were successfully captured by the three single-antenna GNSS-only schemes. In both Period 1 and Period 2, the G2 solution was slightly nosier than those of G1 and G3, mainly because of the relatively fewer satellites tracked by Antenna 2 (see also Figure 3).

Table 2. Experimental positioning schemes.

Positioning Scheme	Antennas	Integrated with IMU
G1	Antenna 1	No
G2	Antenna 2	No
G3	Antenna 3	No
G123	Antenna 1, 2, and 3	No
GI1	Antenna 1	Yes
GI2	Antenna 2	Yes
GI3	Antenna 3	Yes
GI12	Antenna 1 and 2	Yes
GI13	Antenna 1 and 3	Yes
GI23	Antenna 2 and 3	Yes
GI123	Antenna 1, 2, and 3	Yes

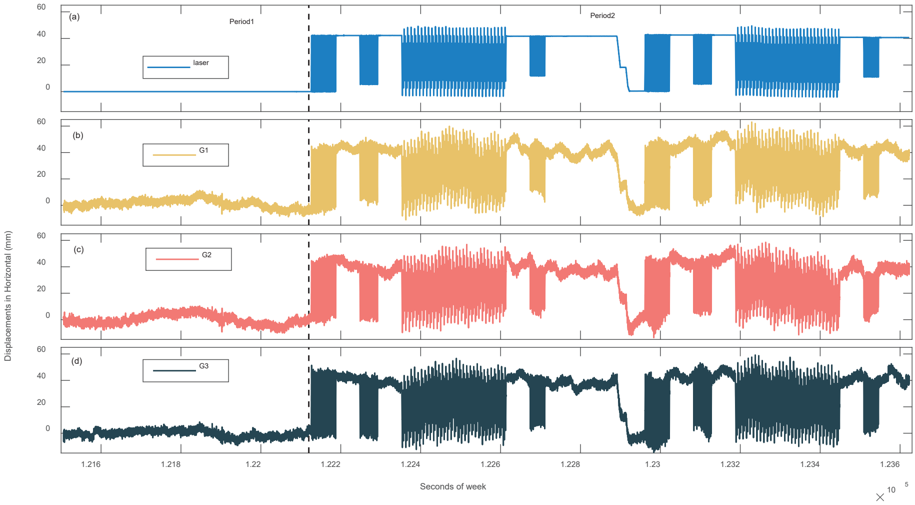


Figure 4. Horizontal displacement time series of the (**a**) ranging laser and (**b**–**d**) single-antenna GNSS-only positioning schemes.

Figure 5 shows the errors of the horizontal displacement of different positioning schemes. We can see from Figure 5a that during Period 1, the positioning errors of single-antenna GNSS-only schemes (i.e., G1, G2, and G3) fluctuate within about 10 mm, and the fluctuations become larger (within 20 mm) in Period 2, maybe due to the coupling of GNSS systematic errors and vibrations. The results of G2 are nosier than those of G1 and G3; for instance, large deviations are present in G2 during 123,400–123,600 s, perhaps due to relatively weaker geometry and undetected outliers. Integrating IMU with single-antenna GNSS data significantly reduces the positioning errors, which become obviously smaller and smoother. However, many low-frequency fluctuations remain visible in the error time series. This implies that the dominant influence is highly correlated with the single-antenna GNSS-only solution errors (Figure 5a). The benefit of multiple antennas, as can be seen in GI12, GI13, GI23, and GI123 solutions, is evident when observations from more GNSS antennas are gradually integrated (Figure 5b–d). Low-frequency systematic errors are mitigated and many error peaks are reduced (see, e.g., the period of 123,400–123,600 s).

This demonstrates that the integration of multiple antennas can enhance the geometry of positioning. Typically, scheme G123, that is based on the average of three corresponding single-antenna GNSS-only solutions (Figure 5d), can achieve similar performance with dual-antenna GNSS/IMU integration (Figure 5c). And the triple-antenna GNSS/IMU solution GI123 further reduces the positioning errors compared with G123 by adding IMU (Figure 5d).

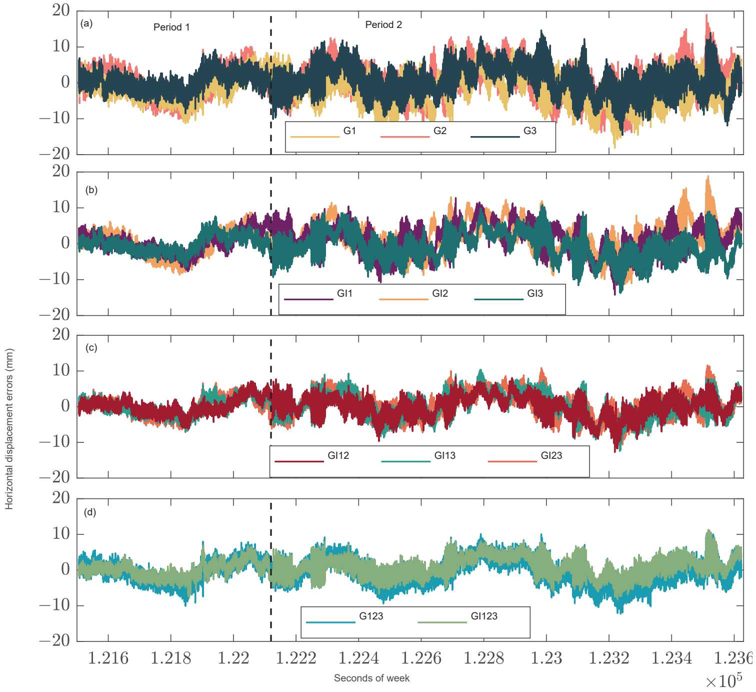


Figure 5. Horizontal displacement errors derived from (**a**) single-antenna GNSS-only, (**b**) single-antenna GNSS/IMU, (**c**) dual-antenna GNSS/IMU, and (**d**) triple-antenna GNSS-only and GNSS/IMU schemes.

Figure 6 shows the histograms of the displacement errors of different positioning schemes. For simplicity, only four typical schemes are shown (including G1, GI1, GI12, and GI123). We can see that the range of positioning errors of G1 is [−18.18, 10.59] mm, which is reduced to [−14.25, 12.79] mm when GNSS/IMU integration is adopted (i.e., GI1). When dual-antenna GNSS/IMU is used (i.e., GI12), the error range is further reduced to [−12.73, 8.38] mm. The triple-antenna GNSS/IMU solution achieves the best results, with the error range of [−8.24, 11.31] mm, and more errors are close to zero.

Table 3 presents the statistics of coordinate standard deviation errors (STDE) for all eleven positioning schemes. For the whole experiment period (including the static period and vibration period), the single-antenna GNSS-only positioning STDE is about 3.88 mm on average, with the STDEs of G1, G2, and G3 being 3.87, 4.21, and 3.57 mm, respectively. When the single-antenna GNSS is tightly integrated with IMU, the corresponding STDE only slightly declines to 3.73 mm (with the STDEs of GI1, GI2, and GI3 being 3.66, 4.12, and 3.41 mm respectively). We can find that the GNSS/IMU positioning performance is related to the positioning quality of the corresponding GNSS antenna used. This effect is less obvious when additional GNSS antennas are used in the integration. The positioning STDE is reduced to about 2.79–3.21 mm by applying dual-antenna GNSS/IMU integration, corresponding to an average accuracy improvement of 21.9% relative to the single-antenna GNSS-only solutions. The triple-antenna GNSS/IMU solution further increases the improvement rate to 33.0% by reducing the STDE to 2.60 mm. Moreover, we notice that the G123 solution is 10.6–24.2% more accurate than the single-antenna solutions (G1, G2, and G3) by means of averaging, and GI123 further reduces the STDE, which is 18.5% smaller than that of G123. This confirms that the use of both IMU and multiple antennas contributes to the improvement in accuracy. As for the statistical results of Period 1 and Period 2, they are similar with those based on the whole data period, while the STDEs of Period 2 increase by about 0.6–1.5 mm compared with those of Period 1, mainly because of the influence of vibrations.

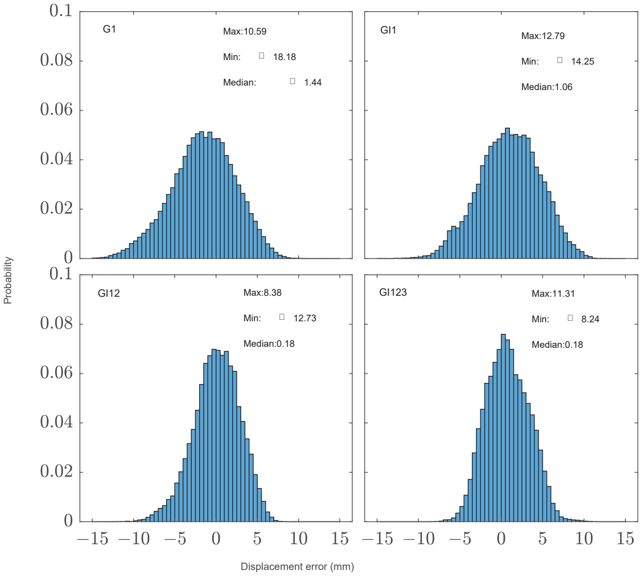


Figure 6. Histograms of displacement errors derived from four typical positioning schemes.

Table 3. Statistical results of 2D horizontal displacement errors of different positioning schemes.

Scheme	Period 1 and 2	Period 1	Period 2
	STDE/mm	STDE/mm	STDE/mm
G1	3.87	3.18	4.10
G2	4.21	3.75	4.36
G3	3.57	2.48	3.92
Mean	3.88	3.14	4.13
GI1	3.66	2.93	3.89
GI2	4.12	3.54	4.22
GI3	3.41	2.24	3.77
Mean	3.73	2.90	3.96
GI12	2.79	2.23	2.98
GI13	3.10	2.31	3.37
GI23	3.21	2.58	3.42
Mean	3.03	2.37	3.26
G123	3.19	2.75	3.35
GI123	2.60	2.08	2.73

Power spectral density (PSD) is calculated to analyze the spectrum of the displacement error sequences from the laser rangefinder and positioning schemes G1, GI1, GI12, and GI123 in the first four vibrations. The results of other positioning schemes, and other vibrations are similar, and thus, for brevity, they are not presented. As can be seen in Figure 7, all positioning schemes can successfully identify the main frequencies of the simulated vibrations, with numerically identical peak-frequency values with respect to those based on laser records. Despite this, compared to the PSD of laser records, the GNSS-only solution G1 is obviously noisier in the high-frequency band (e.g., >1 Hz). The GNSS/IMU solutions exhibit lower noise levels compared to G1, and integrating more GNSS antennas gradually enhances such improvement. This benefits the monitoring of vibrations with high frequencies and small amplitudes, which need further study. As for vibration amplitudes, we applied Fast Fourier Transform to estimate them. Figure 8 shows the errors of amplitudes estimated based on different positioning schemes: they are similar and mostly smaller than 3 mm; those of GI12 present the largest errors among the selected four schemes, perhaps due to the relatively poor data quality of Antenna 2.

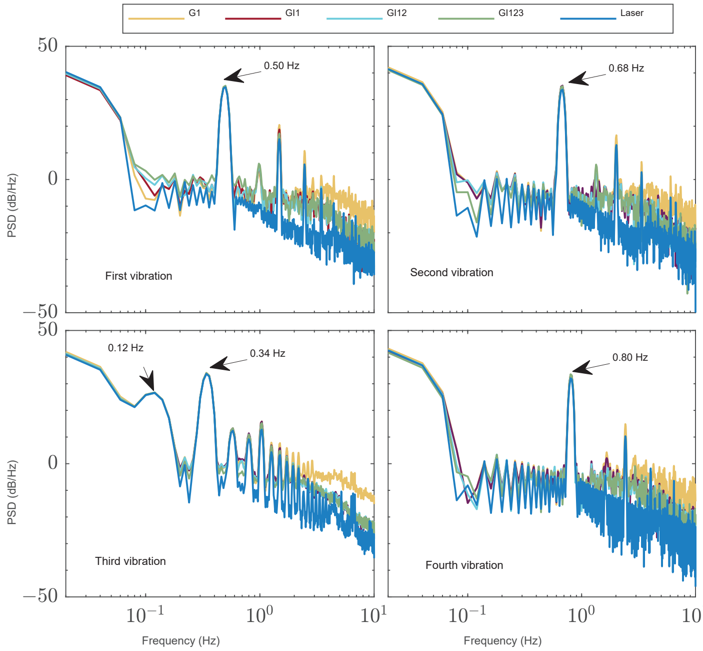


Figure 7. Power spectral density of displacements derived from different positioning schemes.

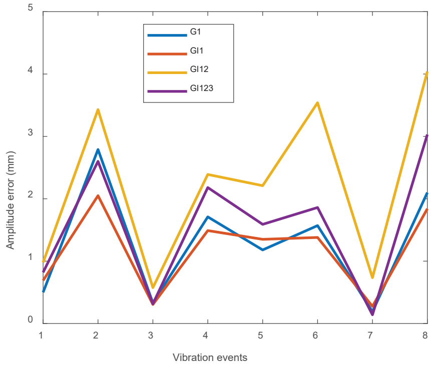


Figure 8. Errors of vibration amplitudes estimated based on displacements of different positioning schemes.

4. Conclusions

In SHM, GNSS can be limited due to signal interference and the accelerometers being subject to rotation-induced baseline errors. Traditional single-antenna GNSS/IMU integrations generally have low observation redundancy and model geometry strength. We propose a tightly integrated multi-antenna GNSS/IMU scheme for structural health monitoring. Vibration signals were simulated with a shaking table to validate the monitoring accuracy of the proposed method, using the laser rangefinder recordings as the reference.

Experimental results show that the STDEs of single-antenna GNSS-only solutions G1, G2, and G3 were 3.87, 4.21, and 3.57 mm, respectively. Compared to the results of single-antenna GNSS-only solutions, the single-antenna GNSS/IMU solutions are affected by the GNSS antenna used; thus, the positioning STDE was only slightly reduced from about 3.88 mm to 3.73 mm on average. Specifically, the STDEs of G1, G2, and G3 were 3.66, 4.12, and 3.41 mm, respectively. Adding more GNSS antennas can effectively increase the improvements. The dual-antenna GNSS/IMU integration reduces the STDE to a level of about 3.03 mm with the STDEs of GI12, GI13, and GI23 being 2.79, 3.10, and 3.21 mm, respectively, and positioning results with an STDE of 2.60 mm were finally obtained by triple-antenna GNSS/IMU integration, which is about 32.99%, 30.29%, and 14.19% more accurate than the single-antenna GNSS-only, single-antenna GNSS/IMU, and dual-antenna GNSS/IMU solutions, respectively. The power spectral density analysis demonstrates that vibration frequencies identified based on the displacements from different positioning schemes are nearly identical. Through Fast Fourier Transform analysis, we found that the benefit of integrating multi-antenna GNSS and IMU in estimating the vibration amplitudes is not obvious. This is perhaps because of the use of poor GNSS data from some antennas.

This study validates the effectiveness of the multi-antenna GNSS/IMU integrated model, and indicates the potential of multi-antenna GNSS/IMU integration for SHM. Further study may apply the proposed method in real SHM environments where more serious signal interference, such as occlusion, multipath, and diffraction errors, is generally present.

Author Contributions: Conceptualization, X.L. and W.Y.; methodology, X.L. and W.Y.; software, X.L. and W.Y.; validation, W.Y. and W.D.; formal analysis, X.L.; investigation, W.Y. and W.D.; data curation, X.Q. and X.D.; writing—original draft preparation, X.L.; writing—review and editing, W.Y. and W.D.; supervision, W.D.; funding acquisition, X.L., W.D., W.Y. and X.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (Grant No. 42104022; 42388102); the Department of Natural Resources of Hunan Province (No. HNGTCH-2023-05); the Innovation and Technology Fund of Hong Kong (Project No. ITP/019/20LP); and the Fundamental Research Funds for the Central Universities of Central South University (Grant No. 2022ZZTS0474).

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ma, Z.; Choi, J.; Sohn, H. Structural displacement sensing techniques for civil infrastructure: A review. *J. Infrastruct. Intell. Resil.* **2023**, *2*, 100041. [CrossRef]
2. Sabato, A.; Dabetwar, S.; Kulkarni, N.N.; Fortino, G. Noncontact Sensing Techniques for AI-Aided Structural Health Monitoring: A Systematic Review. *IEEE Sens. J.* **2023**, *23*, 4672–4684. [CrossRef]
3. Zhang, C.; Mousavi, A.A.; Masri, S.F.; Gholipour, G.; Yan, K.; Li, X. Vibration feature extraction using signal processing techniques for structural health monitoring: A review. *Mech. Syst. Signal Proc.* **2022**, *177*, 109175. [CrossRef]
4. Tefera, B.; Zekaria, A.; Gebre, A. Challenges in applying vibration-based damage detection to highway bridge structures. *Asian J. Civ. Eng.* **2023**, *24*, 1875–1894. [CrossRef]

5. Braunfelds, J.; Senkans, U.; Skels, P.; Porins, J.; Haritonovs, V.; Spolitis, S.; Bobrovs, V. Development of the Strain Measurement Calibration Technique for Road Pavement Structural Health Monitoring Applications Using Optical FBG Sensors. In Proceedings of the 2023 Photonics & Electromagnetics Research Symposium (PIERS), Prague, Czech Republic, 3–6 July 2023; pp. 1060–1065. [CrossRef]
6. Sarmadi, H.; Entezami, A.; Yuen, K.; Behkamal, B. Review on smartphone sensing technology for structural health monitoring. *Measurement* **2023**, *223*, 113716. [CrossRef]
7. Entezami, A.; Arslan, A.N.; De Michele, C.; Behkamal, B. Online Hybrid Learning Methods for Real-Time Structural Health Monitoring Using Remote Sensing and Small Displacement Data. *Remote Sens.* **2022**, *14*, 3357. [CrossRef]
8. Wang, X.; Zhao, Q.; Xi, R.; Li, C.; Li, G.; Li, L. Review of Bridge Structural Health Monitoring Based on GNSS: From Displacement Monitoring to Dynamic Characteristic Identification. *IEEE Access* **2021**, *9*, 80043–80065. [CrossRef]
9. Tamura, Y.; Matsui, M.; Pagnini, L.; Ishibashi, R.; Yoshida, A. Measurement of wind-induced response of buildings using RTK-GPS. *J. Wind Eng. Ind. Aerodyn.* **2002**, *90*, 1783–1793. [CrossRef]
10. Nakamura, S. GPS Measurement of Wind-Induced Suspension Bridge Girder Displacements. *J. Struct. Eng.* **2000**, *126*, 1413–1419. [CrossRef]
11. Xiong, C.; Wang, M.; Chen, W. Data analysis and dynamic characteristic investigation of large-scale civil structures monitored by RTK-GNSS based on a hybrid filtering algorithm. *J. Civ. Struct. Health Monit.* **2022**, *12*, 857–874. [CrossRef]
12. Niu, Y.; Ye, Y.; Zhao, W.; Shu, J. Dynamic monitoring and data analysis of a long-span arch bridge based on high-rate GNSS-RTK measurement combining CF-CEEMD method. *J. Civ. Struct. Health Monit.* **2021**, *11*, 35–48. [CrossRef]
13. Stiros, S.C. Errors in velocities and displacements deduced from accelerographs: An approach based on the theory of error propagation. *Soil Dyn. Earthq. Eng.* **2008**, *28*, 415–420. [CrossRef]
14. Meng, X.; Wang, J.; Han, H. Optimal GPS/accelerometer integration algorithm for monitoring the vertical structural dynamics. *J. Appl. Geod.* **2014**, *8*, 265–272. [CrossRef]
15. Kim, K.; Choi, J.; Chung, J.; Koo, G.; Bae, I.; Sohn, H. Structural displacement estimation through multi-rate fusion of accelerometer and RTK-GPS displacement and velocity measurements. *Measurement* **2018**, *130*, 223–235. [CrossRef]
16. Xin, S.; Geng, J.; Zeng, R.; Zhang, Q.; Ortega-Culaciati, F.; Wang, T. In-situ real-time seismogeodesy by integrating multi-GNSS and accelerometers. *Measurement* **2021**, *179*, 109453. [CrossRef]
17. Sugisaki, K.; Abe, M.; Koshimizu, S. Inclination and vibration measurement by inertial sensing for structural health monitoring. *J. Jpn. Soc. Civ. Eng. A1* **2012**, *68*, 216–225. [CrossRef]
18. Sekiya, H.; Kinomoto, T.; Miki, C. Determination Method of Bridge Rotation Angle Response Using MEMS IMU. *Sensors* **2016**, *16*, 1882. [CrossRef] [PubMed]
19. Rossi, Y.; Tatsis, K.; Awadaljeed, M.; Arbogast, K.; Chatzi, E.; Rothacher, M.; Clinton, J. Kalman Filter-Based Fusion of Collocated Acceleration, GNSS and Rotation Data for 6C Motion Tracking. *Sensors* **2021**, *21*, 1543. [CrossRef]
20. de Alteriis, G.; Conte, C.; Caputo, E.; Chiariotti, P.; Accardo, D.; Cigada, A.; Schiano Lo Moriello, R. Low-Cost and High-Performance Solution for Positioning and Monitoring of Large Structures. *Sensors* **2022**, *22*, 1788. [CrossRef]
21. Geng, J.; Wen, Q.; Chen, Q.; Chang, H. Six-Degree-of-Freedom Broadband Seismogeodesy by Combining Collocated High-Rate GNSS, Accelerometers, and Gyroscopes. *Geophys. Res. Lett.* **2019**, *46*, 708–716. [CrossRef]
22. Cai, X.; Chai, H.; Ding, L.; Hsu, H.; Wang, Y. Multi-Antenna GNSS and INS Integrated Position and Attitude Determination without Base Station for Land Vehicles. *J. Navig.* **2019**, *72*, 342–358. [CrossRef]
23. Wang, Y.; Zhao, X.; Pang, C.; Feng, B.; Zhang, L. The Influence of Attitude Dilution of Precision on the Observable Degree and Observability Analysis with Different Numbers of Visible Satellites in a Multi-Antenna GNSS/INS Attitude Determination System. *IEEE Access* **2018**, *6*, 22156–22164. [CrossRef]
24. Li, X.; Zhou, Y.; Shen, Z.; Song, B.; Li, S. Using a Moving Antenna to Improve GNSS/INS Integration Performance under Low-Dynamic Scenarios. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 17717–17728. [CrossRef]
25. Lu, G. Development of a GPS Multi-Antenna System for Attitude Determination. Ph.D. Thesis, University of Calgary, Calgary, AB, USA, 1995. [CrossRef]
26. Zhang, Q.; Chen, Q.; Xu, Z.; Zhang, T.; Niu, X. Evaluating the navigation performance of multi-information integration based on low-end inertial sensors for precision agriculture. *Precis. Agric.* **2021**, *22*, 627–646. [CrossRef]
27. Li, Q.; Bai, Z.; Chen, B.; Guo, J.; Xin, H.; Chen, Y.; Li, Q.; Wu, F. A novel track measurement system based on GNSS/INS and multisensor for high-speed railway. *Acta Geod. Cartogr. Sin.* **2020**, *49*, 569–579. [CrossRef]
28. Groves, P. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 2nd ed.; Industrial Robot; Artech: McLean, VA, USA, 2013; pp. 191–192.
29. Allan, D. Time and Frequency (Time-Domain) Characterization, Estimation, and Prediction of Precision Clocks and Oscillators. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* **1987**, *34*, 647–654. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

MDPI AG
Grosspeteranlage 5
4052 Basel
Switzerland
Tel.: +41 61 683 77 34

Remote Sensing Editorial Office
E-mail: remotesensing@mdpi.com
www.mdpi.com/journal/remotesensing

Disclaimer/Publisher's Note: The title and front matter of this reprint are at the discretion of the Guest Editors. The publisher is not responsible for their content or any associated concerns. The statements, opinions and data contained in all individual articles are solely those of the individual Editors and contributors and not of MDPI. MDPI disclaims responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

MDPI

Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-2922-4