remote sensing

MDPI

Article

Tightly Coupled LIDAR/IMU/UWB Fusion via Resilient Factor Graph for Quadruped Robot Positioning

Yujin Kuang, Tongfei Hu, Mujiao Ouyang, Yuan Yang * and Xiaoguo Zhang

Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China; 230239009@seu.edu.cn (Y.K.); xgzhang@seu.edu.cn (X.Z.)

* Correspondence: yangyuan@seu.edu.cn

Abstract: Continuous accurate positioning in global navigation satellite system (GNSS)-denied environments is essential for robot navigation. Significant advances have been made with light detection and ranging (LiDAR)-inertial measurement unit (IMU) techniques, especially in challenging environments with varying lighting and other complexities. However, the LiDAR/IMU method relies on a recursive positioning principle, resulting in the gradual accumulation and dispersion of errors over time. To address these challenges, this study proposes a tightly coupled LiDAR/IMU/UWB fusion approach that integrates an ultra-wideband (UWB) positioning technique. First, a lightweight point cloud segmentation and constraint algorithm is designed to minimize elevation errors and reduce computational demands. Second, a multi-decision non-line-of-sight (NLOS) recognition module using information entropy is employed to mitigate NLOS errors. Finally, a tightly coupled framework via a resilient mechanism is proposed to achieve reliable position estimation for quadruped robots. Experimental results demonstrate that our system provides robust positioning results even in LiDAR-limited and NLOS conditions, maintaining low time costs.

Keywords: light detection and ranging; ultra-wide-band; inertial navigation system; resilient factor graph; tightly coupled integration

Citation: Kuang, Y.; Hu, T.; Ouyang, M.; Yang, Y.; Zhang, X. Tightly Coupled LIDAR/IMU/UWB Fusion via Resilient Factor Graph for Quadruped Robot Positioning. *Remote Sens.* **2024**, *16*, 4171. https://doi.org/10.3390/rs16224171

Academic Editor: Kan Wang

Received: 10 October 2024
Revised: 6 November 2024
Accepted: 7 November 2024
Published: 8 November 2024

1. Introduction

In the past three decades, mobile robots have garnered significant attention for performing tasks without human interventions in complex environments, e.g., space exploration, rescue operations, and industrial manufacturing. Based on their modes of movements, robots are classified into wheeled robots, tracked robots, and legged robots [1]. Wheeled and tracked robots appeared earlier, and their technologies are more mature and relatively simple to operate, making them the most widely used types of mobile robots today. However, traditional wheeled or tracked robots are often unsuitable for unstructured terrain [2]. Inspired by the movement patterns of humans and animals in nature, legged robots—specifically those with bipedal or quadrupedal movement—navigate complex terrain more freely, presenting significant research potential [3]. Compared with bipedal robots, quadrupedal robots offer better stability and controllability [4]. In the field of quadrupedal robots, beyond robot structures and control methods, addressing positioning and navigation in complex environments is also crucial [5].

Currently, the integration of global navigation satellite system (GNSS) and inertial navigation system (INS) is considered sufficient to meet the positioning requirements of quadruped robots in most scenarios. However, operating in wide satellite-denied areas poses a challenge for achieving continuous and reliable positioning for quadruped robots [6].

Simultaneous localization and mapping (SLAM) with cameras or light detection and ranging (LiDAR) is a promising positioning solution [7]. LiDAR-based solutions offer

several advantages over camera-based ones. LiDAR is unaffected by lighting conditions, allowing it to maintain excellent positioning performance at night or in environments with drastic light changes [8]. Additionally, LiDAR provides a larger detection field of view and a wider sensing range, enabling the accurate measurement of parameters, i.e., distance, angle, and speed of target [9].

Despite the advantages of LiDAR-based positioning schemes, several challenges persist. One of the main issues is the small vertical resolution of LiDAR, which leads to weak elevation constraints. As a result, elevation errors of LiDAR-based SLAM algorithms accumulate over time, causing biased elevation estimations. To address these problems, researchers propose specialized processing based on ground point clouds to impose stronger constraints on the Z-axis. Since LiDAR point clouds do not inherently label points as belonging to the ground, and considering the real-time requirements of SLAM, the primary research goal becomes the development of methods for fast and robust ground segmentation [10–12]. LeGO-LOAM successfully implements ground segmentation, but the authors focus only on point cloud segmentation without exploring the impact of ground segmentations on SLAM performance [13]. Liu et al. [14] employ random sample consensus (RANSAC) to segment the ground plane and then extract surface features from ground point clouds and edge features from non-ground point clouds. This method performs well in simple environments, but real-world road conditions are complex, and obstacles such as grass easily impact effectiveness. To overcome these challenges, some researchers adopt more advanced ground segmentation algorithms, such as Patchwork, instead of traditional segmentation methods [15]. Xu et al. [16] propose a ground-optimized LiDAR odometry called PaGO-LOAM, which incorporates Patchwork. Although this algorithm lightens the system, it partially leverages ground constraints for global optimizations. To make full use of ground information, Kenji Koide et al. [17] use factor graphs to integrate ground information into the SLAM backend, aiming to correct cumulative drift on the Z-axis. However, the constraints are overly rigid, leading to insufficient system robustness. In summary, the Z-axis of LiDAR SLAM is prone to significant drifts, and further research is needed for developing lightweight ground segmentation algorithms that fully use ground information for global constraints.

As known, LiDAR-based schemes rely heavily on point cloud features, making them vulnerable to estimation drift and tracking loss in sparse feature environments [18]. Additionally, LiDAR-based schemes only estimate the relative position, which poses challenges in specialized environments, e.g., in fire rescue situations, it is necessary to refer to a predefined world frame to follow a specific trajectory. One potential solution is to train neural networks to recognize segmented objects from point clouds for localization [19]. However, this approach requires significant effort in data collection, labeling, and training. Furthermore, changes in environments over time render the system ineffective. Another solution involves establishing global observation constraints using deployed indoor infrastructure, e.g., Bluetooth, Wi-Fi, pseudo-satellites, and ultra-wideband (UWB). Bluetooth technology offers advantages such as low power consumption and cost-effectiveness. However, its operational range is limited and susceptible to obstruction, rendering it unsuitable for accurate indoor positioning. Wi-Fi, on the other hand, typically requires no additional infrastructure and is convenient for most indoor environments; however, its positioning accuracy is often compromised due to signal interference and multipath effects, making it less reliable for high-precision applications. Pseudo-satellite signals can emulate satellite signals in GNSS environments, offering high positioning accuracy. Nonetheless, the establishment of pseudo-satellite systems is complex and costly, necessitating precise synchronization between devices, which poses challenges for practical implementation [20]. In contrast, UWB positioning technology provides centimeter-level accuracy, exhibits strong resistance to multipath interference, and maintains a controllable cost, making it particularly suitable for high-precision indoor positioning and garnering increasing interest within the SLAM community [21].

While the location results from UWB devices are promising, several limitations remain, with non-line-of-sight (NLOS) being one of the most significant challenges. The NLOS issues typically arise due to obstructions, reflections, and other factors during signal transmission. These factors result in an increased propagation time, causing the wirelessly measured distance to be greater than the actual distance. Consequently, this leads to measurement outliers, ultimately resulting in the failure of cooperative localization [22]. Ensuring accurate identification of NLOS signals is crucial for positioning accuracy [23]. One approach involves a signal propagation path loss model, which assumes that UWB path energy under line-of-sight (LOS) conditions is significantly greater than NLOS conditions. However, this method requires modeling the distribution of various statistics, necessitating large, frequently updated training databases [24]. Collecting NLOS data is often cumbersome and resource-intensive. Another strategy is to leverage environmental context information to make NLOS judgments [25–27]. Zhang et al. [28] propose using indoor spatial prior information to create a LOS/NLOS infographic of anchor points, which accurately distinguish LOS/NLOS conditions in complex indoor environments. Though this approach looks feasible, generating the initial environmental map for large buildings is a daunting task [29]. Recently, the development of machine learning and deep learning technologies leads more researchers to explore detecting NLOS signals. However, the effectiveness of these models depends heavily on pre-trained datasets, and their reliability decreases with changes in the environment or dynamic obstacles. Additionally, due to the complexity of these mathematical models, meeting real-time prediction requirements remains a challenge.

Considering the limitations associated with relying solely on a single sensor in real-world applications, and recognizing the complementary advantages of LiDAR, IMU, and UWB sensors, a positioning approach based on sensor fusion offers significant benefits. Firstly, integrating LiDAR and IMU with UWB enhances the reliability of UWB positioning systems in NLOS situations. Secondly, since LiDAR/IMU-based navigation is recursive, errors tend to accumulate as the robot travels farther. Introducing UWB observations helps to suppress this error accumulation, improving overall accuracy. Most importantly, UWB, as an absolute positioning method, converts the LiDAR/IMU results from the absolute coordinate system to the global coordinate. This capability is crucial for indoor navigation tasks that require global referencing.

Most data fusion methods, typified by the Kalman filter (KF), rely on Bayesian filtering theory [30,31]. He et al. [32] propose a LiDAR inertial odometer framework using tightly coupled iterative KF significantly reducing the complexity of Kalman gain calculation. Marković et al. [33] employ extended Kalman filters (EKF) to loosely couple the positioning results of LiDAR, vision, and UWB. Wang et al. [34] introduce a particle filter based on an improved sparrow search algorithm to tightly couple UWB/LiDAR. Compared to filtering-based methods, factor graph-based optimization approaches represent a growing trend [35]. These methods use a probabilistic graph model with nodes linked to the system state, incorporating all historical data. The probability expressions are converted into a nonlinear least-squares (LS) problem for optimizations, employing efficient incremental techniques for optimal navigation state solutions. Zhou et al. [36] propose a LiDAR/UWB fusion algorithm based on graph optimization, achieving sensor degradation detection and adaptive adjustment of fusion parameters. Although Nguyen et al. [37] developed a tightly coupled LiDAR/INS/UWB positioning scheme utilizing Ceres optimization, their system does not account for NLOS errors. In contrast, our system incorporates ground constraint and NLOS recognition techniques, enabling it to address the effects of sparse point clouds and NLOS errors in real-world scenarios. Additionally, we fully use the factor graph's plug-and-play resilient mechanism to discard unreliable sensor data in real time, thereby reducing computational costs.

This paper proposes a LiDAR/IMU/UWB positioning system for quadruped robots, with the main contributions of this work are as follows:

1. To overcome the poor elevation robustness associated with the sparse vertical point clouds in traditional LiDAR positioning, a lightweight point cloud segmentation algorithm is proposed. By effectively utilizing the segmented point clouds, a ground constraint factor that maximizes adaptability to variations in the ground plane is constructed.

2. To mitigate the NLOS issues of UWB caused by obstacle occlusions, a multi-decision NLOS recognition module based on information entropy is developed, which is able to mitigates NLOS errors.

3. Leveraging the compatibility of factor graphs with diverse information, a tightly coupled framework based on a resilient strategy is proposed to ensure reliable position estimation for quadruped robots. Experiments conducted in real-world scenarios validate the superior performance.

The remainder of this paper is structured as follows. Section 2.1 provides an overview of the complete system flow. Section 2.2 introduces a lightweight and efficient point cloud processing module. The details of NLOS identification using information entropy and multiple sensors are described in Section 2.3. Section 2.4 presents an LiDAR/IMU/UWB tightly coupled localization algorithm. Detailed experimental settings and discussions are reported and analyzed in Section 3. Finally, Section 4 summarizes the work by drawing several conclusions and offers a vision for future research.

2. Materials and Methods

2.1. System Overview

The flowchart of the proposed method is illustrated in Figure 1. Initially, high-frequency pose estimation is performed by pre-integrating IMU data, which is then applied to correct point cloud distortion and construct pre-integration residuals. Next, a point cloud segmentation algorithm is employed to segment the LiDAR point cloud. From the segmented ground point cloud, surface features are extracted to construct the ground constraint factor, while inter-frame matching of the non-ground point cloud is used to create the odometry factor, both of which are incorporated into the factor graph optimization. To address the impact of NLOS environmental obstructions on ranging information, a multi-decision NLOS recognition module based on information entropy is introduced. This module effectively distinguishes between LOS and NLOS UWB measurements, eliminating NLOS errors. Finally, we employ a resilient factor graph framework to tightly couple LiDAR, IMU, and UWB data, fully leveraging historical information and available sensor inputs to achieve precise global pose estimation for the quadruped robot.

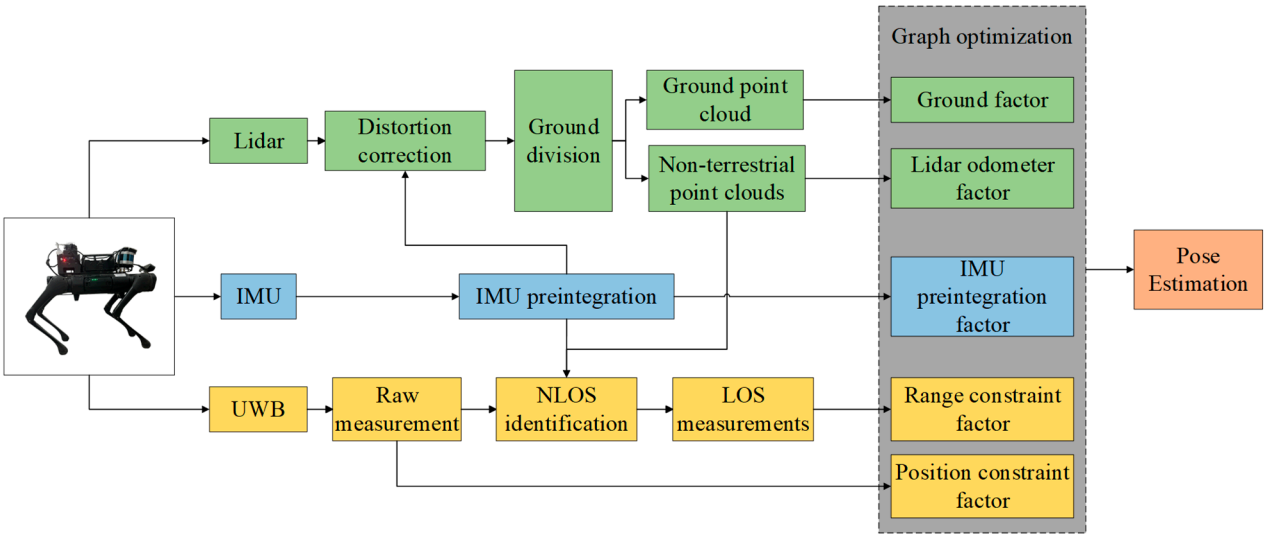


Figure 1. Overview of the proposed tightly coupled LiDAR/IMU/UWB positioning system.

2.2. Lightweight Point Cloud Segmentation

Traditional ground constraints assume the ground is horizontal, which limits their applicability to real-world scenarios. In practice, there are often slopes, stairs, and other inclined surfaces. Introducing incorrect ground information worsens vertical drift. To address uneven terrain and large-scale environments with pits and slopes, this paper adopts a two-stage ground detection method. First, possible ground point clouds are identified based on the ground normal vector. Then, a new ground segmentation algorithm is introduced to assist in accurately extracting ground point clouds.

The first stage primarily mitigates interference point clouds by analyzing the triangle formed between the LiDAR beam hitting the ground, the height of LiDAR and the ground. As shown in Figure 2, the solid line represents the LiDAR hitting the ground, while the dashed line represents the LiDAR hitting a non-ground surface. The height between the LiDAR and the ground is calculated using the LiDAR's distance to the ground and the angle of the LiDAR. The Equation (1) is as follows:

$$h = s * \cos\theta \tag{1}$$

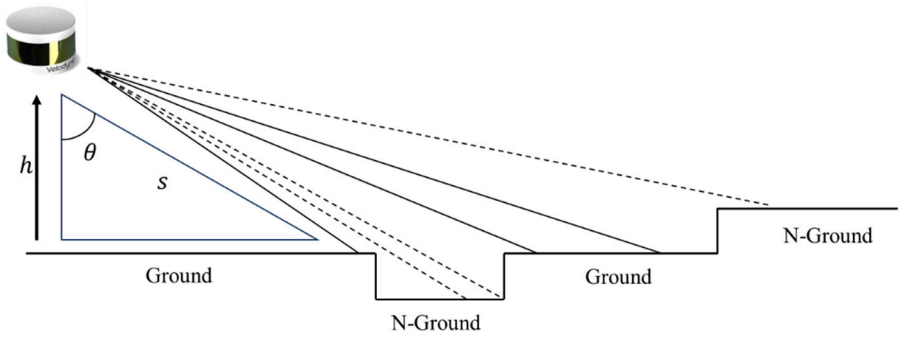


Figure 2. Triangular-based ground point cloud exclusion diagram.

For LiDAR positioning, it is crucial that its horizontal coordinate system remains parallel to the ground. During movement, the height from the LiDAR sensor to the ground typically shows only minor variations. However, if there are significant ground features—like convex structures or steep slopes—the calculated height value will vary considerably, which can impact the stability and accuracy of LiDAR-based ground detection. Since the initial height of the LiDAR from the ground is known, the difference between the calculated value and the initial height is used for judgment. If the difference is within a certain threshold, the point is considered a ground point; otherwise, it is classified as a non-ground point. The above steps allow for rough screening of ground point clouds. However, due to uneven ground surfaces and potential potholes, such point clouds cannot always be directly classified as ground. To address this, a new segmentation method called Patchwork [15] is adopted to assist in extracting ground point clouds in the second stage. Patchwork is optimized to make it lightweight and easily embedded in the LiDAR SLAM system, allowing for smoother operation. The ground segmentation algorithm consists of two parts: polar grid representation based on the concentric belt model and ground fitting.

The first part, in a nutshell, is based on a multi-plane approach. Since the observable world is not flat, it is impossible to process the entire point cloud as a whole. Instead, we assume that the point cloud of each region is flat and reduce the global problem to local considerations. Traditional methods convert sectors into uniformly sized bins, but to improve generalization—taking into account the fact that LiDAR point clouds are denser at closer distances—the point cloud can be further subdivided based on this phenomenon. By analyzing the distribution of LiDAR points, the point cloud is divided into several regions of varying size according to distance and density, as shown in Figure 3. By marking the location of the LiDAR as the center of a circle, the collected point cloud is

divided into several small fan-shaped areas according to certain rules. The most common method is to use a unified polar grid to represent the sector S, dividing it into smaller regions with regularly spaced radial and azimuth directions, i.e., rings and sectors. The Equations (2) and (3) are as follows:

$$N_a = \pi / a \tag{2}$$

$$N_l = L / l \tag{3}$$

where L is the radius of the circle, l is the length of the diameter after the circle is equally divided, a is the central angle of the equally divided sections, N_a is the segmented sector area, and N_l is the segmented inner ring number.

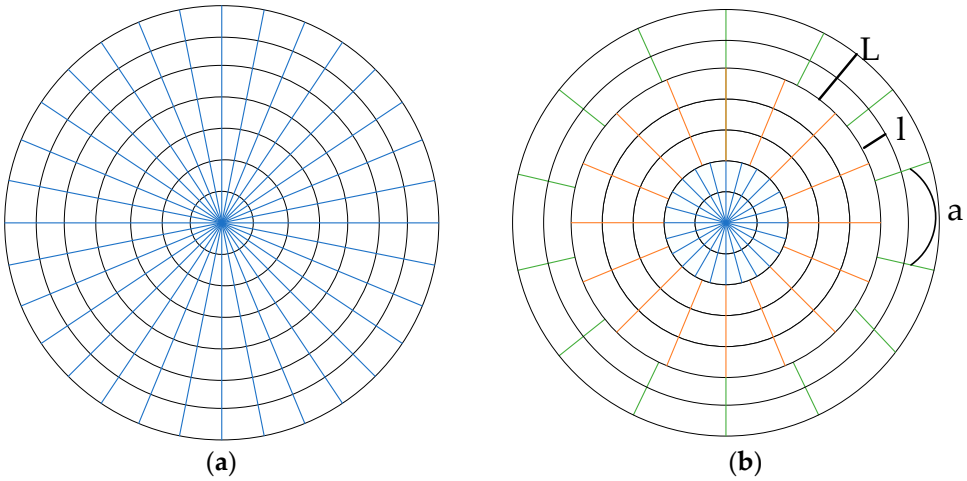


Figure 3. Concentric circle model of ground segmentation. (**a**) traditional model on the left; (**b**) improved model on the right.

Compared with traditional methods, the new concentric circle point cloud processing approach not only enhances the robustness of the point cloud but also reduces the number of regions and improves the algorithm's real-time performance. After that, plane fitting is then performed for the point clouds in each region. Since a two-stage extraction method is used in this paper, a large number of interference point clouds have already been filtered out, allowing the RANSAC method to improve the accuracy and completeness of ground extraction without compromising real-time performance. Three points are randomly selected in each point cluster $G = \{g_i | i = 1, 2, \ldots, k\}$ for plane fitting, and the fitting Equation (4) is as follows:

$$\begin{cases} n^T * g_1 + d = 0 \\ n^T * g_2 + d = 0 \\ n^T * g_3 + d = 0 \\ \quad ||n|| = 1 \end{cases} \tag{4}$$

where n is the normal vector of the plane and d is the intercept of the plane equation.

The cost function (5) for plane fitting is constructed by minimizing the distance from each point to the fitting plane, and is expressed as follows:

$$\underset{n,d}{\operatorname{argmin}} \sum_{i=1}^{k} \frac{\left| n^T * g_i + d \right|}{\|n\|} \tag{5}$$

After several iterations of the algorithm, the plane with the smallest cost function is identified as the ground plane. The detected ground information is then used to construct a ground factor for the backend to complete error suppression.

2.3. Multi-Decision Non-Line-of-Sight Judgment Based on Information Entropy

In the actual process of tracking a target using a UWB system, the anchor station or tag is easily obstructed by walls, columns, pedestrians, and other objects, leading to multipath signal propagation and NLOS ranging errors. These NLOS errors cause ranging failures, which may compromise the stability of the factor graph fusion solution, resulting in trajectory deviations or even complete localization failures. To address the instability of ranging signals in NLOS environments, a multi-decision NLOS recognition module based on information entropy is designed. This module uses information entropy to determine the weight of each decision criterion. By weighting each decision score, the module dynamically determines the reliability of ranging information through decision fusion, effectively identifying and mitigating NLOS errors.

The concept of information entropy is first proposed by Claude Shannon, the "father of information theory", to address the problem of quantitatively measuring information [38]. This paper fully leverages NLOS criteria to calculate all decisions, which are divided into decisions based on signal strength and decisions based on distance signal. Decisions based on signal strength primarily depend on the error between the first path signal strength fp and total received signal strength rx received by the tag from each anchor station during the positioning process [39].

For the i-th UWB ranging value, calculate the difference Δr_i between the received signal strength rx_i and fp_i at the current time k. The Equation (6) is as follows:

$$\Delta r_i = rx_i - fp_i \tag{6}$$

The decision based on the ranging signal mainly relies on the residual between the ranging value and the LiDAR/IMU odometer. Specifically, the LiDAR/IMU, as an autonomous system, maintains a high degree of stability over short periods of time. Therefore, in an NLOS environment, the distance measured by the UWB will typically be greater than the distance estimated by the odometer. For the i-th UWB ranging value, calculate the difference, as defined in Equation (7), between the UWB label measurement m_i and the odometer estimate o_i at the current time k:

$$\Delta d_i = m_i - o_i \tag{7}$$

Based on the above error values, the probability, as defined in Equation (8), is estimated using the normal distribution hypothesis:

$$P(\Delta_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\Delta_i-\mu)^2}{2\sigma^2}} \tag{8}$$

where Δ_i represents the difference value of the i-th anchor station under each sub-decision criterion, μ is the mean value under the current decision criterion, and σ is the standard deviation.

For each sub-decision criterion, its information entropy, as defined in Equation (9), is calculated to measure its contribution to the NLOS judgment:

$$H(\Delta_i) = -P(\Delta_i)\log(P(\Delta_i)) \tag{9}$$

Next, the current information entropy, as defined in Equation (10), is updated using the EWMA method:

$$H_k(\Delta_i) = \alpha \cdot H_k(\Delta_i) + (1-\alpha) \cdot H_{k-1}(\Delta_i) \tag{10}$$

where α is the smoothing coefficient, with a range of $(0, 1]$. A larger value indicates a greater influence from new data, while a smaller value means a greater influence from historical data.

At each time point, the updated signal strength error entropy and ranging signal residual entropy are combined to calculate the joint information entropy. The Equation (11) is as follows:

$$H_k(\Delta r_i, \Delta d_i) = H_k(\Delta r_i) + H_k(\Delta d_i) \tag{11}$$

The calculated joint information entropy is used for real-time NLOS detection. A high entropy value indicates significant uncertainty in the signal propagation process, suggesting NLOS conditions. Conversely, a lower entropy value implies a more certain signal path, indicating possible LOS conditions. Based on the comprehensive evaluation value, the NLOS judgment for each sample effectively reduces the impact of large ranging errors on position calculation.

2.4. Factor Graph Fusion Framework Based on Resilient Mechanism

The factor graph algorithm provides a concise and intuitive representation of complex probabilistic relationships between variables in a system. At its core, a factor graph represents a probability model as a bipartite graph, consisting of two types of nodes: variable nodes and factor nodes. Variable nodes correspond to the random variables in the model, while factor nodes represent functions that capture the relationships between these variables. The edges in the graph connect variable nodes to factor nodes.

In the integrated navigation problem, the position, attitude of the carrier, and ranging information are uncertain, making robot integrated navigation essentially a probability estimation problem. Specifically, the combinatorial navigation problem is a Maximum a Posteriori (MAP) estimation problem of the robot state X for a given measurement value Z. The maximum posterior probability estimation of a single state is defined in Equation (12):

$$X^{MAP} = \underset{X}{\operatorname{argmax}}\, P(X/Z) = \underset{X}{\operatorname{argmax}}\, P(Z/X)P(X) \tag{12}$$

In a Bayesian network, each observed variable is solved separately, so all conditional probabilities are expressed as products and are decomposed. In the factor graph, the decomposition of each term becomes a factor, and these factors are multiplied together to form a graph, which is used to describe the factor graph. In this paper, three types of variables are included: LiDAR raw measurements, IMU raw measurements, and UWB raw measurements. According to the central limit theorem, the noise of most sensors follows a Gaussian distribution, so each factor is defined by an exponential function, which corresponds to the error function. Therefore, the above equation is broken down as shown in Equation (13):

$$X^{MAP} = \underset{X}{\operatorname{argmax}}\, f(X) = \underset{X}{\operatorname{argmax}} \prod_k f^{Lidar}(x_k) \prod_k f^{IMU}(x_k) \prod_k f^{UWB}(x_k) \tag{13}$$

where x_k represents the set of all state variables involved in the factor node. $f^{Lidar}(x_k)$, $f^{IMU}(x_k)$ and $f^{UWB}(x_k)$ represent the conditional probabilities of LiDAR, IMU and UWB measurements in a given state, respectively.

The established fusion localization factor model of a quadruped robot is shown in Figure 4. The state vector corresponding to the current state space of the system is defined in Equation (14):

$$X_k = [P_k, V_k, Q_k, B_k]^T \tag{14}$$

where the subscript k represents the location epoch, and P, V, Q represents the position, velocity, and orientation of the carrier at the current epoch, respectively. B_k stands for the acceleration and gyroscope bias of the IMU.

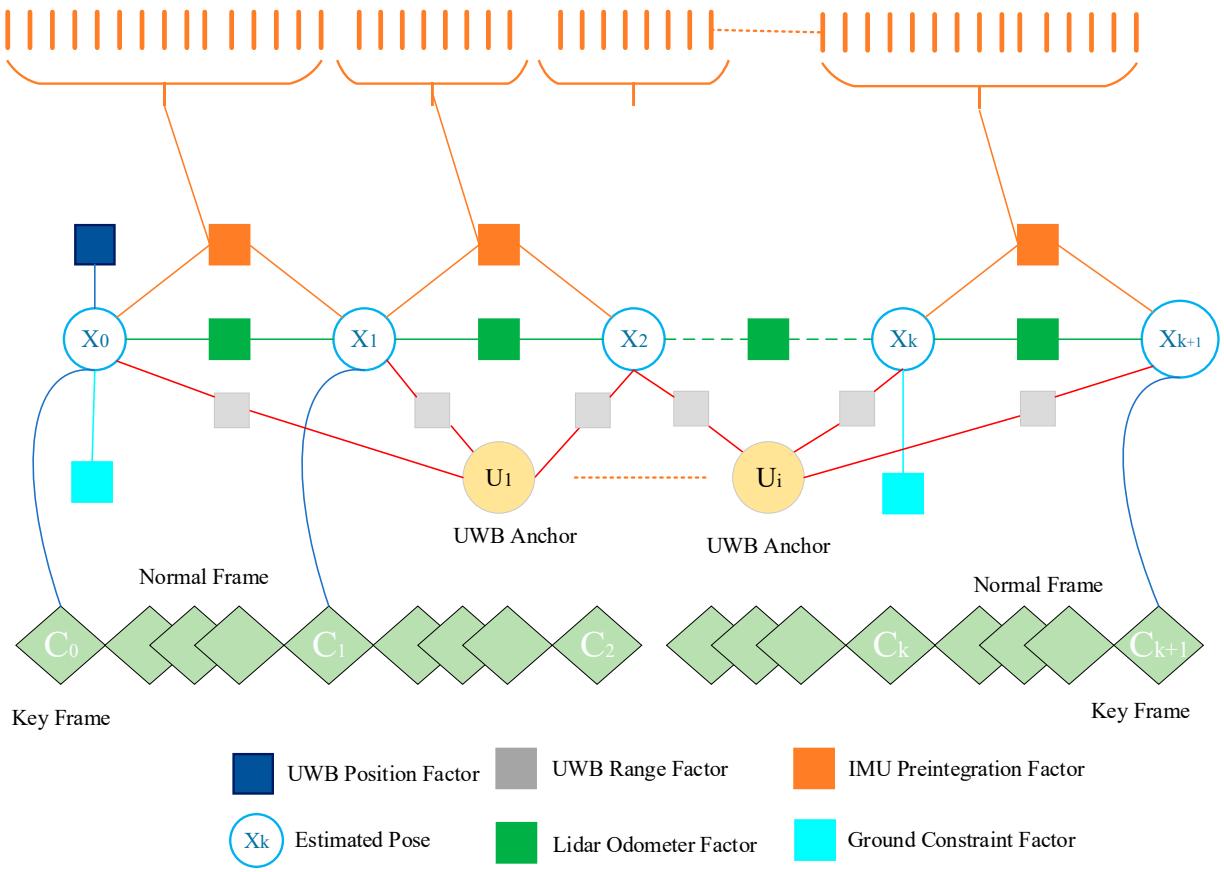


Figure 4. Schematic of the proposed LiDAR/IMU/UWB factor graph model of quadruped robot.

2.4.1. LiDAR Measurement Model

The edge feature point set F_k^e and surface feature point set F_k^p extracted from the current frame point cloud are matched with the edge feature point F_{k-1}^e and surface feature point F_{k-1}^p in the previous frame point cloud. The pose transformation from frame $k-1$ to frame k is then calculated. Edge points form lines in three-dimensional space, and matching them is achieved by constructing point-to-line constraints [40]. A point-to-line residual formula is constructed as shown in Equation (15):

$$D_{(e,k)} = \frac{\left|\left(p_{k,i}^e - p_{k-1,u}^e\right) \times \left(p_{k,i}^e - p_{k-1,v}^e\right)\right|}{\left|\left(p_{k-1,u}^e - p_{k-1,v}^e\right)\right|} \tag{15}$$

where $p_{k,i}^e$ is a feature point in F_k^e, $p_{k-1,u}^e$ and $p_{k-1,v}^e$ are feature points in F_{k-1}^e.

Similarly, the residual formula from point to surface is constructed as shown in Equation (16) [41]:

$$D_{(p,k)} = \left|\left(p_{k,j}^p - p_{k-1,u}^p\right) \bullet \frac{\left(p_{k-1,u}^p - p_{k-1,v}^p\right) \times \left(p_{k-1,u}^p - p_{k-1,w}^p\right)}{\left|\left(p_{k-1,u}^p - p_{k-1,v}^p\right) \times \left(p_{k-1,u}^p - p_{k-1,w}^p\right)\right|}\right| \tag{16}$$

where $p_{k,j}^p$ is a feature point in F_k^p, $p_{k,u}^e$, $p_{k,v}^e$ and $p_{k,w}^e$ are feature points in F_{k-1}^p.

The detected ground information is used to construct the ground factor input at the back end to complete error suppression. If the ground is successfully extracted from key frame k, the feature point set of ground points is F_k^g, and the ground equation fitted by ground feature points is denoted as $A_k X_k + B_k Y_k + C_k Z_k = D_k$. Then, the distance $D_{(g,k)}$, as defined in Equation (17), from the ground point $p_k^g(x, y, z)$ of frame k to the ground of

frame $k-1$ is calculated, as well as the distance $D_{(s,k)}$, as defined in Equation (18), from the ground plane of frame k to the ground plane $A_{k-1}X_{k-1} + B_{k-1}Y_{k-1} + C_{k-1}Z_{k-1} = D_{k-1}$ of frame $k-1$.

$$D_{(g,k)} = \frac{|A_k x + B_k y + C_k z|}{\sqrt{{A_k}^2 + {B_k}^2 + {C_k}^2}} \tag{17}$$

$$D_{(s,k)} = \frac{|D_k - D_{k-1}|}{\sqrt{{A_k}^2 + {B_k}^2 + {C_k}^2}} \tag{18}$$

After constructing the residual models for point-line, point-plane, and ground constraints, the odometer residuals, as defined in Equation (19), and ground constraint residuals, as defined in Equation (20), between laser point clouds are obtained.

$$r_{LiDAR}\left(L_{k-1}^{k}, X\right) = \sum_{p_k^e \in F_k^e} D_{(e,k)} + \sum_{p_k^p \in F_k^p} D_{(p,k)} \tag{19}$$

$$r_{Ground}\left(L_{k-1}^{k}, X\right) = \sum_{p_k^g \in F_k^g} D_{(g,k)} + D_{(s,k)} \tag{20}$$

In large indoor spaces such as underground parking lots and factories, the ground may be uneven or have small slopes. In such cases, blindly adding ground constraints may introduce incorrect information, which is counterproductive. Therefore, it is necessary to continuously fit the angle of the normal vector of the ground plane between key frames. If this angle exceeds a given threshold, the algorithm concludes that the quadruped robot is not moving on horizontal ground. In such cases, no ground constraints are added during the back-end position optimization. Conversely, if the robot is determined to be moving on horizontal ground, ground constraints are added to improve the accuracy and robustness of pose estimation. In addition, considering that the keyframe selection strategy results in the keyframes to be too far apart, the ground corresponding to adjacent keyframes may be different. To address this, a strategy is implemented in which the ground constraint is applied exclusively between keyframes that fall within a specified spatial and temporal range. By constraining the selection of keyframes in this manner, we ensure that adjacent keyframes correspond to similar ground surfaces, thereby facilitating the application of ground constraints.

2.4.2. IMU Measurement Model

Since the measurement frequency of the IMU is usually above 100 Hz, establishing factor nodes for each IMU measurement greatly increases the computational burden. To address the difficulties caused by the high overhead of a large number of factor nodes, pre-integration technology is adopted [42]. Assuming there are LiDAR measurement updates at the i time and the j time, it is necessary to pre-integrate all the IMU data between them. This allows us to obtain the position, velocity, and rotation at the j time as initial estimates, as shown in Equation (21).

$$\begin{aligned}
p_{wb_j} &= p_{wb_i} + v_i^w \Delta t - \frac{1}{2} g^w \Delta t^2 + q_{wb_i} \iint_{t \in [i,j]} \left(q_{b_i b_t} (\mathrm{a}^{b_t} - b_t^a) \right) \Delta t^2 \\
v_j^w &= v_i^w - g^w \Delta t + q_{wb_i} \int_{t \in [i,j]} \left(q_{b_i b_t} (\mathrm{a}^{b_t} - b_t^a) \right) \Delta t \\
q_{wb_j} &= q_{wb_i} \int_{t \in [i,j]} q_{b_i b_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2} w^{b_t} \Delta t \end{bmatrix}
\end{aligned} \tag{21}$$

By constructing the above equation, the pre-integrated component becomes dependent only on the IMU measurement values. Therefore, we only need to directly integrate the IMU data over a period of time to obtain the pre-integrated component. This approach

significantly reduces the number of factor nodes, thereby greatly improving the speed of optimization. Replace the pre-integration component with a variable, as shown in Equation (22).

$$\begin{aligned} \alpha_{b_i b_j} &= \iint_{t\in[i,j]} \left(q_{b_i b_t} (\mathrm{a}^{b_t} - b_t^a) \right) \Delta t^2 \\ \beta_{b_i b_j} &= \int_{t\in[i,j]} \left(q_{b_i b_t} (\mathrm{a}^{b_t} - b_t^a) \right) \Delta t \\ q_{b_i b_j} &= \int_{t\in[i,j]} q_{b_i b_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2} w^{b_t} \Delta t \end{bmatrix} \end{aligned} \tag{22}$$

The pre-integral residual is defined as shown in Equation (23).

$$r_{IMU}(I_{k-1}^k, X) = \begin{bmatrix} \mathbf{r}_p \\ \mathbf{r}_q \\ \mathbf{r}_v \\ \mathbf{r}_{ba} \\ \mathbf{r}_{bg} \end{bmatrix}_{15\times 1} = \begin{bmatrix} \mathbf{q}_{b_i w} \left(\mathbf{p}_{w b_j} - \mathbf{p}_{w b_i} - \mathbf{v}_i^w \Delta t + \frac{1}{2} \mathbf{g}^w \Delta t^2 \right) - \boldsymbol{\alpha}_{b_i b_j} \\ 2 \left[\mathbf{q}_{b_j b_i} \otimes \left(\mathbf{q}_{b_i w} \otimes \mathbf{q}_{w b_j} \right) \right]_{xyz} \\ \mathbf{q}_{b_i w} \left(\mathbf{v}_j^w - \mathbf{v}_i^w + \mathbf{g}^w \Delta t \right) - \boldsymbol{\beta}_{b_i b_j} \\ \mathbf{b}_j^a - \mathbf{b}_i^a \\ \mathbf{b}_j^g - \mathbf{b}_i^g \end{bmatrix} \tag{23}$$

In the residual, displacement, velocity, and bias are all obtained by direct subtraction. The second term represents the rotation error in the quaternion, where $[.]_{xyz}$ denotes a three-dimensional vector composed only of the imaginary part of the quaternion (x, y, z).

2.4.3. UWB Measurement Model

When the positioning system is initially activated, the global position of the system is determined through LS estimation. $\gamma_i = (x_i, y_i, z_i)$ represents the position of i-th anchor, while the initial position $p_0 = (x_0, y_0, z_0)$ of the target node remains unknown. The objective of the LS method is to minimize the sum of squared errors between the estimated distances and the measured distances from the target node to each anchor point. The corresponding cost function is formulated as shown in Equation (24).

$$\min \sum_{i=1}^{M} \left(\hat{z}_i - \sqrt{(\mathrm{x}_0 - \mathrm{x}_i)^2 + (\mathrm{y}_0 - \mathrm{y}_i)^2 + (\mathrm{z}_0 - \mathrm{z}_i)^2} \right)^2 \tag{24}$$

where M represents the number of anchor and $\hat{z}_i$ denotes the received ranging values. The LS method is applied to solve this optimization problem by iteratively adjusting p_0 to minimize the cost function, ensuring an optimal estimate of the initial position.

By solving this optimization problem, the initial position estimate of the target is obtained. This initial value is then utilized as a global position constraint to construct the residual term of the position constraint factor, as shown in Equation (25).

$$r_{Position}(Z, X) = p - p_0 = \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} \tag{25}$$

where $p = (x, y, z)$ represents the variable to be optimized for the initial position of the tag.

The UWB observation signal is subsequently utilized to construct the range constraint factor, as shown in Equations (26) and (27).

$$h(P_k, \gamma_{ki}) = \sqrt{(\mathrm{x}_k - \mathrm{x}_{k,i})^2 + (\mathrm{y}_k - \mathrm{y}_{k,i})^2 + (\mathrm{z}_k - \mathrm{z}_{k,i})^2} \tag{26}$$

$$r_{Range}(Z_k, X) = \| \, z_k^{UWB} - h(P_k, \gamma_{ki}) \, \|_{\Sigma_k^{UWB}}^2 \tag{27}$$

where (x_k, y_k, z_k) represents the position of tag at time k. $\gamma_{k,i} = (x_{k,i}, y_{k,i}, z_{k,i})$ represents the position of the i-th anchor at time k, $z_{k,i}^{UWB}$ represents the measured distance between the tag and the i-th anchor at time k. Σ_k^{UWB} represents the covariance of the UWB ranging model.

For the proposed tightly coupled LiDAR/IMU/UWB factor graph structure, the traditional approach involves solving the error function corresponding to each sensor using fixed parameters. While this strategy works in most scenarios, it has limitations in challenging indoor environments. For instance, sensors like UWB are significantly affected by NLOS conditions, leading to large ranging errors. If the solution remains fixed under these conditions, singular values may occur, negatively impacting positioning accuracy. To address this issue, this paper proposes a resilient mechanism for the adaptive solution of the objective function. For UWB observation data, the UWB signal quality is assessed using a multi-decision NLOS recognition module based on information entropy. Depending on the module's discriminant results, the UWB error function is adjusted—either increased or decreased—flexibly. After the measurements from each sensor are obtained, the corresponding factor nodes are constructed, and the estimated state value is determined by solving the nonlinear LS problem to minimize the error function of each factor. The objective function for the constructed LS problem is as shown in Equation (28).

$$e(X) = \underset{X}{\operatorname{argmin}}(\|r_{Lidar}(L,X)\|^2 + \|r_{Ground}(L,X)\|^2 + \|r_{IMU}(I,X)\|^2 + \left\|r_{Range}(Z,X)\right\|^2 + \|r_{Position}(Z,X)\|^2) \quad (28)$$

Among these, the five residuals represent the odometer constraints, ground constraints, IMU pre-integration constraints across interframe intervals, LOS observation constraints, and the initial position constraints provided by UWB.

3. Experimental Results

3.1. Experimental Platform and Environments

The experimental platform is a quadruped robot from our laboratory, and the experimental equipment is depicted in Figure 5a. This equipment includes a LinkTrack P-B UWB (Nooploop, Nanjing, China), an ADIS16497 IMU (Analog Devices, Norwood, MA, USA), and a VLP-16 LiDAR (Velodyne, San Jose, CA, USA). The UWB hardware used in this experiment is shown in Figure 5b, which follows the IEEE 802.15.4-2011 UWB standard and consists of multiple anchor nodes and a single mobile (tag) node. This experiment uses six anchor nodes and one mobile node. The distance between each anchor node and the mobile node was measured using two-way ranging based on time of flight (TOF). Since the experiment was conducted in an indoor environment without access to GNSS signals, the ATS-320M high-precision total station (Starfish, FoShan, China) was used to determine the global positions of the anchor stations and evaluation points, as shown in Figure 5c. The processing unit is an embedded controller, specifically the Jetson Xavier NX, running on the Ubuntu operating system and utilizing the robot operating system (ROS). The processing unit can obtain relevant data by installing the official ROS (Noetic version) driver package for each hardware component. Using ROS commands, it is possible to acquire and store messages from each sensor.

Since LiDAR, IMU, and UWB are three different types of positioning systems, spatial calibration and time synchronization between sensors are necessary steps for successful multi-sensor fusion. For spatial calibration, a high-precision total station was used to accurately measure the spatial positions of all sensors. For time synchronization, the ROS framework was used to assign timestamps to each sensor with the same time reference, and the three sensor types were synchronized by aligning the data timestamps. To minimize the impact of parameter estimation errors, it is crucial to estimate key parameters as accurately as possible during the initial stage of the experiment.

Figure 5. Experimental equipment: (**a**) Quadruped robot with sensors (Unitree, Hangzhou, China); (**b**) UWB hardware module (Nooploop, Nanjing, China); (**c**) Total station (Starfish, Foshan, China).

The experiment was conducted in the underground parking lot of Sipailou Campus of Southeast University. The experimental environment is depicted in Figure 6. UWB anchor points 1 to 4 are located in the initial LOS region for the quadruped robot to initialize and align the sensor coordinate system, while UWB anchor points 5 and 6 are in the initial NLOS region. Before the experiment, the total station instrument was used to accurately measure the coordinates of each anchor point. During the experiment, pedestrians and vehicles intermittently passed through the test area, which affected the UWB mobile tag signal due to both static and dynamic obstacles.

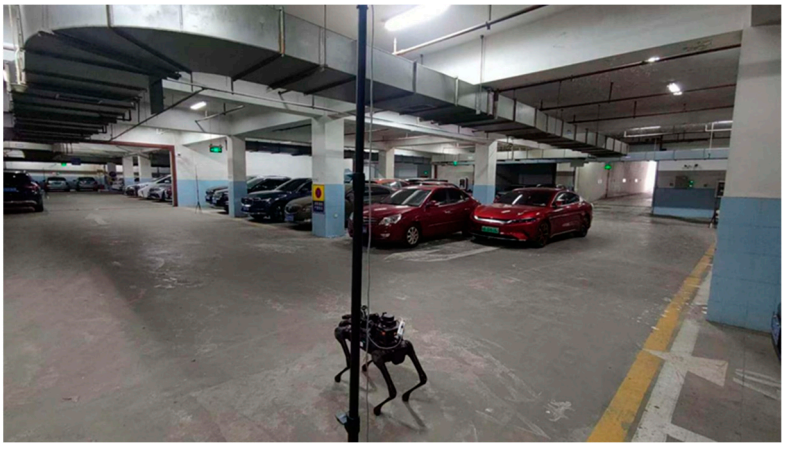


Figure 6. UWB anchor deployment and key points of the mobile experiment.

The experimental environment was set up as depicted in Figure 7. In this setup, the red triangles represent the UWB anchor points, while the blue circle indicates the position of the quadruped robot. The fusion factor graph algorithm was primarily utilized to enhance the dynamic positioning performance of the indoor environment, so the experimental path has been designed to be a curved trajectory. The quadruped robot was programmed to move along a rectangular motion path that encompassed the entire experimental area. For evaluation purposes, four specific points (purple five-pointed star) were marked on the

ground to serve as reference benchmarks. When the robot reached each evaluation point, it was kept stationary to allow for error statistics to be gathered. This approach enabled a thorough analysis of the algorithm's performance in a controlled environment.

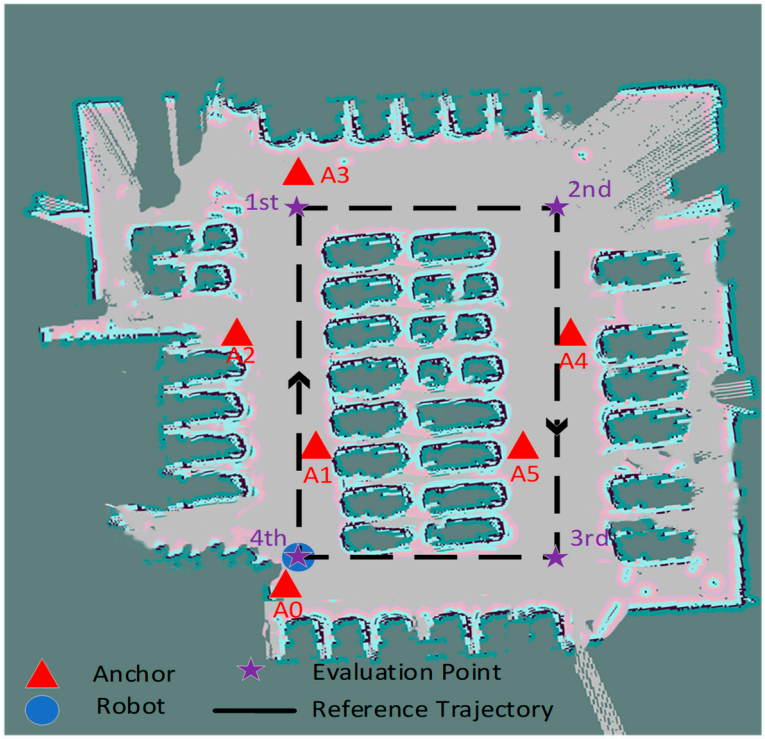


Figure 7. Experiment layout. The red triangles indicate the locations of the UWB anchor stations. The blue dot represents the location of the robot tag. The black line shows the reference track.

3.2. Analysis of Lidar Point Cloud Based on Segmentation Constraints

The point cloud module based on segmentation constraints has two main tasks: first, to improve system operation efficiency by segmenting ground point clouds, and second, to correct height errors using ground point cloud constraints. As a result, the performance evaluation was divided into two aspects. The first was to test how our ground segmentation algorithm improves system performance when processing point clouds, primarily by comparing the average running time of different algorithms when processing LiDAR frames. The second was to evaluate the effect of adding ground constraints to the odometry. The algorithm was tested on two datasets: the first is the Kitti dataset, and the second is the Parking dataset, which was collected in the parking lot using the quadruped robot experimental platform. The proposed algorithm was applied to the pure LiDAR SLAM algorithm LEGO-LOAM and compared to LOAM (which does not include ground processing) and PAGO-LOAM (which uses the state-of-the-art segmentation algorithm PATCHWORK).

Figure 8 shows the average running time of three algorithms for processing a frame of LiDAR point cloud in different datasets. Drive_0018 and Drive_0027 are Kitti datasets using a 64-line LiDAR, while Parking uses the 16-line LiDAR from the quadruped robot platform. As a result, the number of point clouds in the Parking dataset is smaller, and the processing speed of front-end feature extraction is faster. The LiDAR operates at 10 Hz, and if the system takes more than 100 ms to process, some LiDAR frames will be lost during the calculation process. The Drive_0018 and Drive_0027 datasets were chosen because they cover varied and representative conditions within the urban area scenario, including different levels of environmental complexity, lighting conditions, and typical obstacles. These datasets allowed us to comprehensively evaluate the robustness and performance of our proposed method. In addition, the Drive_0018 and Drive_0027 will return to the starting point at the end of the period, which can be very good for comparing whether there is a Z-axis drift. As shown in the figure, the running time of our algorithm is significantly

lower than that of A-LOAM, which lacks ground processing. Additionally, the single frame processing time is also 4.5 ms higher than that of PAGO-LOAM algorithm, indicating that our ground segmentation algorithm, which first performs rough ground extraction and then refines it using the RANSAC algorithm, is highly feasible and suitable for deployment in low-power systems.

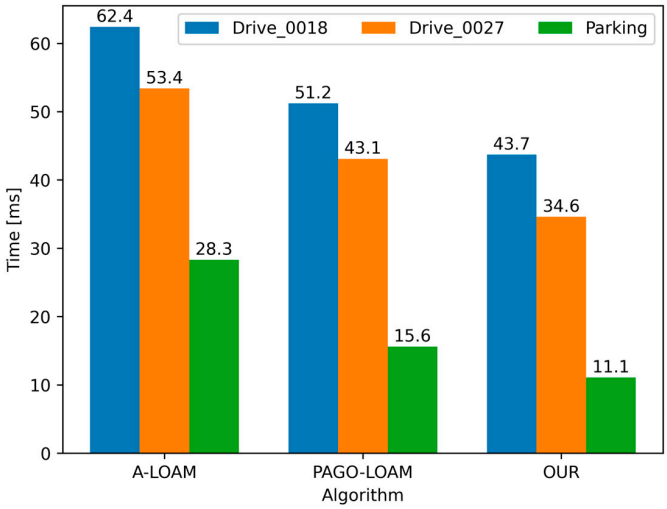


Figure 8. Comparison of running time for processing a frame of point cloud by different algorithms across various datasets.

The Kitti datasets provide reliable ground truth for trajectories, making them suitable for evaluating the proposed system in this paper. The evo tool is used to assess the absolute pose error (APE) between several algorithms. The root mean square error (RMSE) of the APE for the Kitti dataset is shown in Figure 9. Since A-LOAM does not include a loop closure mechanism, our method also does not enable loop closure detection for any of the sequences. In comparison, the trajectory produced by our algorithm shows high accuracy. In the Drive_0018 sequence, which covers a long distance of up to 2.21 km, the cumulative drift in A-LOAM cannot be corrected in time, resulting in a larger error of 18.051 m. In contrast, our method reduces the error to 4.42 m. In the Drive_0027 sequence, thanks to the addition of back-end ground constraints, our algorithm performs well on roads with elevation changes, achieving an error of just 1.086 m, which is significantly lower than that of other algorithms.

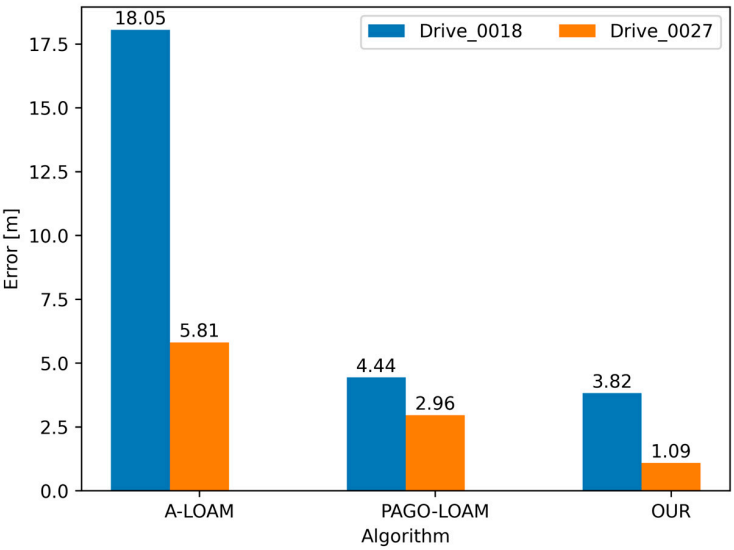


Figure 9. Error comparison of positioning results for different algorithms across various datasets.

To demonstrate the high availability of our ground constraint algorithm, we integrated it into LIO-SAM and conducted extensive testing, achieving excellent performance and stability. The test results are summarized in Table 1, showing that the accuracy of our method is 1.13 m, significantly surpassing that of traditional LIO-SAM approaches. This improvement underscores the effectiveness of incorporating ground constraints into the positioning process.

Table 1. Performance of LIO-SAM in a dataset with ground.

Dataset	RMSE Error [m]	
	LIO-SAM	Ours
Drive_0027	2.54	1.13

To further evaluate the performance of the proposed algorithm in ground segmentation tasks, particularly in complex and dynamic real-world environments, we selected a typical test scene featuring multiple steps and curbs. This environment simulates challenging terrain commonly encountered in daily life, where structures like steps and curbs are difficult for traditional ground segmentation methods, which often misclassify these features as the ground. The experimental results are illustrated in Figure 10, which visually compares the segmentation performance of various algorithms within the same test environment. As shown, the site contains complex terrain elements, such as multiple steps and curbs, that are challenging to identify accurately in real-world applications. Traditional methods often overlook critical height variations and geometric characteristics of the terrain, mistakenly classifying step areas as part of the ground and reducing segmentation accuracy. In contrast, our algorithm more effectively captures and utilizes key information, including terrain height, slope, and planarity, allowing for precise identification and segmentation of step and curb structures. The figure demonstrates that our method not only successfully differentiates between ground and non-ground regions but also exhibits superior accuracy and robustness when handling complex features like steps and curbs.

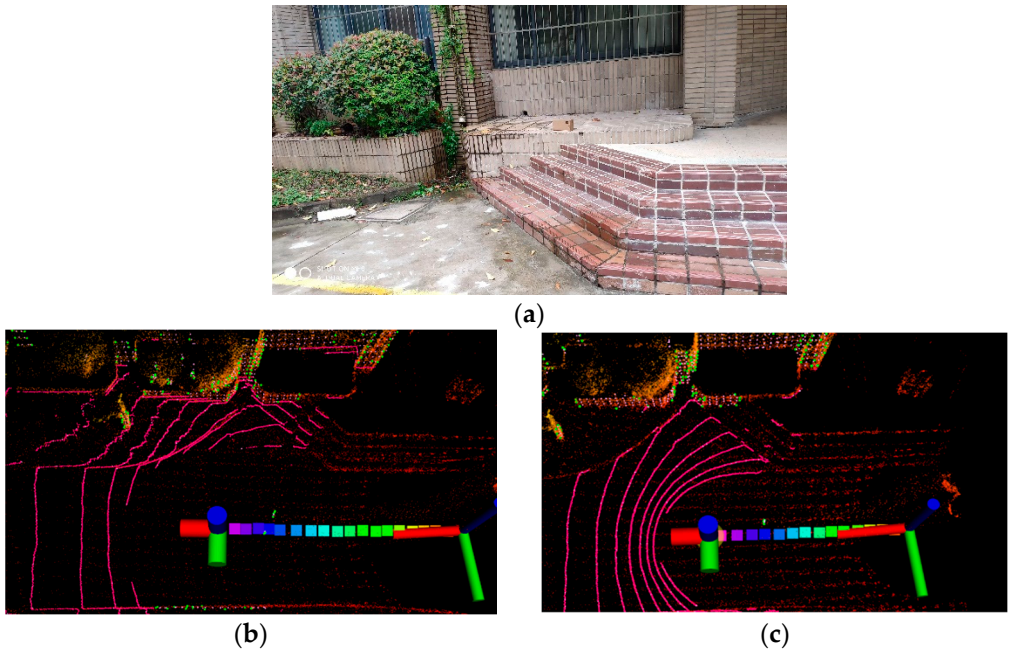
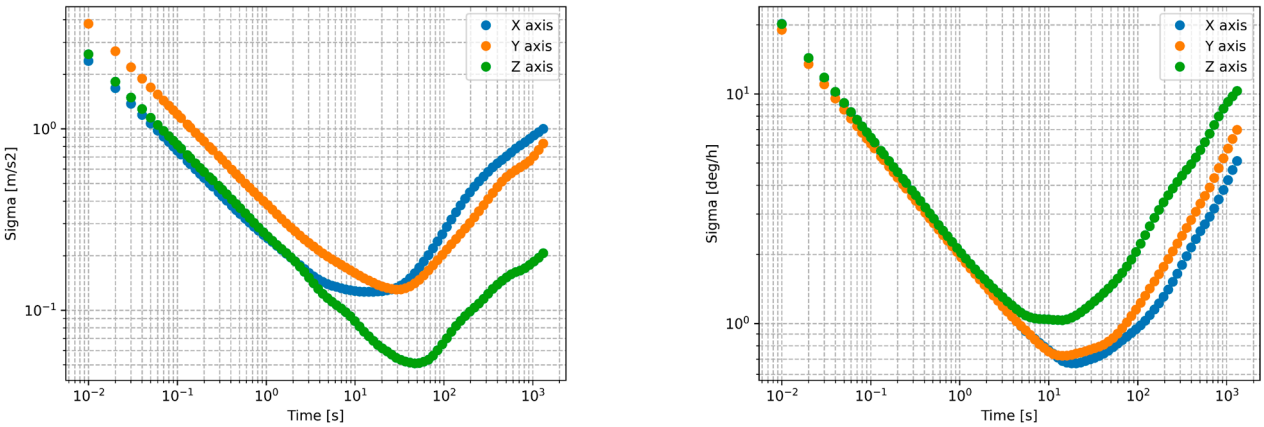


Figure 10. Comparison of point cloud segmentation results. The pink line represents the fitted ground point cloud. (**a**) Field diagram of a typical complex environment with multiple steps and curbs. (**b**) Segmentation result using a traditional point cloud processing algorithm (**c**) Segmentation result using the proposed algorithm.

3.3. ALLAN Variance Analysis of IMU

In INS positioning, the IMU integrates sensor information to estimate position through pre-integration, making IMU sensor errors critical to positioning accuracy. The measurement noise from the IMU primarily includes accelerometer noise, which introduces bias in velocity and position estimates, and gyroscope noise, which affects attitude angle estimation. This study addresses the drift and noise characteristics of the sensor by establishing an IMU error model. Allan variance, a widely used tool in the field of inertial navigation systems, quantitatively analyzes sensor performance by describing the statistics of drift and random noise in inertial sensors, such as accelerometers and gyroscopes, over time. In this paper, measurement data from the IMU were collected during a ten-hour stationary period, and the noise characteristics were analyzed using Allan variance, as illustrated in Figure 11. Utilizing the values for bias instability and random walk, we established the noise covariance matrix and noise model for the combined positioning process, aiming to minimize the impact of IMU system errors on positioning results during optimization.



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

Figure 11. IMU Allan variance analysis results. (**a**) Accelerometer. (**b**) Gyroscope.

3.4. UWB Ranging Error Mitigation

The UWB ranging error primarily consists of two components: system error and NLOS error.

3.4.1. System Error Elimination Module

The UWB system error primarily arises from the characteristics, design, manufacturing, calibration, and usage of the equipment, leading to variability in error profiles among different devices. If these errors are not addressed, they can introduce biases during the positioning solution phase. In this study, the UWB equipment was sampled within the effective working range in an indoor Line-of-Sight (LOS) environment, and an error function model was established through data fitting. By actively compensating for the inherent system errors of the sensor, the overall system error is mitigated, thereby enhancing ranging accuracy and reducing the influence of these errors on the final positioning results.

During the sampling process of equidistant ranging values, a complete LOS scenario was maintained to ensure that there are no obstructions and no personnel moving between the anchor station and the tag. Both the anchor station and the tag were fixed on 1 m tripods. For the procedure, anchor station 1 (A0) was taken as an example: A0 remained stationary while the tag was moved progressively from the nearest distance of 1 m. A total of 60 sampling points were collected. The device acquisition frequency was set to 20 Hz, and static sampling was performed at each position for 60 s, resulting in approximately 1200 ranging values per sampling point. The true distances were measured using a high-precision total station. For modeling the system error, anchor station A1 was used as an example to fit the ranging error. To balance fitting accuracy and computational complexity, a fourth-order

polynomial was selected as the system error description function (29). This choice helped to avoid the risks of underfitting or overfitting, ensuring a robust representation of the system error while maintaining manageable computational requirements.

$$d_0 = -0.00014{x_0}^4 + 0.00286{x_0}^3 - 0.02089{x_0}^2 + 1.05664x_0 + 0.09052 \quad (29)$$

where d_0 represents the fitted ranging value of anchor station A0, and x_0 represents the measured value of anchor station A0. The same applies to the other anchor stations.

The error optimization effect of the UWB system is illustrated in Figure 12. In the figure, the blue bar chart represents the original ranging error of the anchor station, while the green bar chart shows the ranging error after correction using the error model. As the figure demonstrates, the UWB ranging error is significantly reduced after applying the correction. The error model established through data fitting effectively reduces the ranging error, thereby improving the overall ranging accuracy. This optimization is crucial for ensuring the precision of UWB positioning in LOS scenarios.

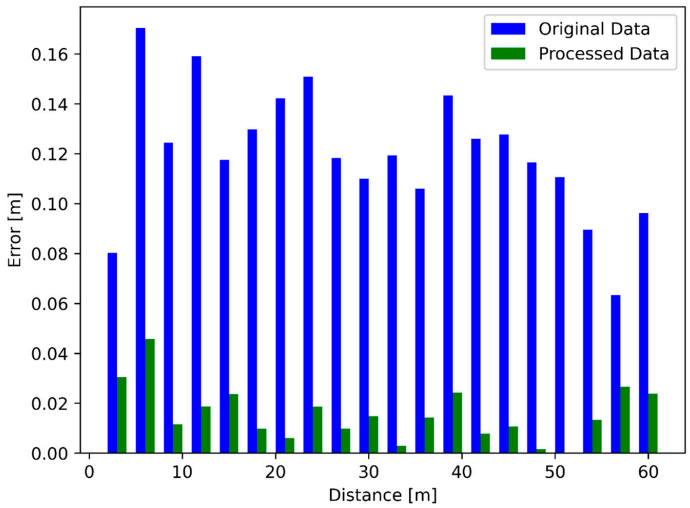


Figure 12. The result of distance measurement error optimization.

3.4.2. Non-Line-of-Sight Recognition Module Based on Information Entropy

For NLOS recognition, experiments were conducted in the previously mentioned parking lot environment. Figure 13 shows the difference curve between the first path signal strength and the total received signal strength for each anchor station during the experiment. It can be observed that, as the robot moves, each anchor station is affected by varying degrees of NLOS interference.

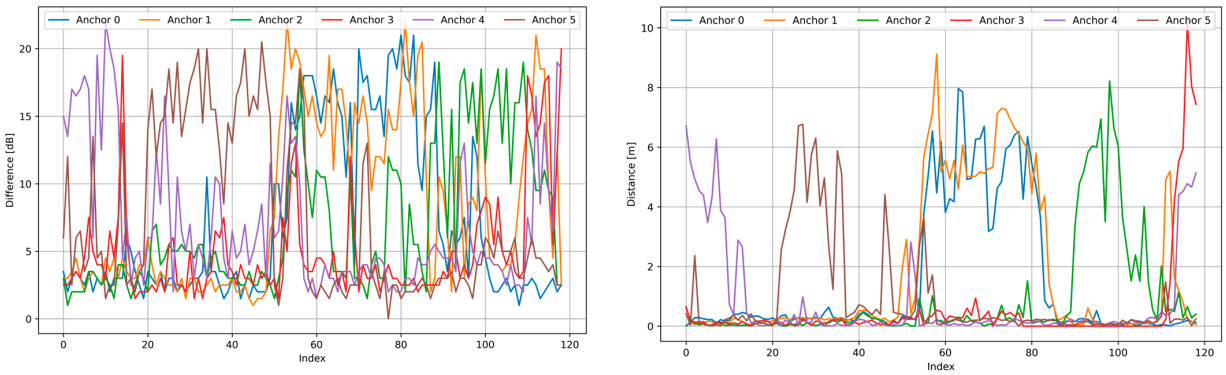


Figure 13. Changes in the data curve for each decision criterion. (**a**) Signal strength difference curve for each anchor station; (**b**) Distance signal difference curve for each anchor station.

Figure 14 shows a comparison between the evaluation results of each anchor station based on information entropy and the real situation. In the figure, the horizontal axis represents each key epoch of the quadruped robot in the parking lot environment, and the vertical axis indicates the LOS/NLOS status identified by each anchor station. A value of 0 indicates LOS, and 1 indicates NLOS. The actual LOS/NLOS status, determined based on the robot's movement trajectory and the environment, is represented by a solid black line. As seen in the figure, the NLOS recognition algorithm based on information entropy was used to accurately identify the LOS/NLOS status of the anchor station, closely matching the actual trend. When a ranging error occurs, the ranging information from the anchor station is classified as NLOS. While there are a few errors at individual decision points, the number of LOS selections is less than in the actual situation, effectively ensuring the robustness of the positioning system.

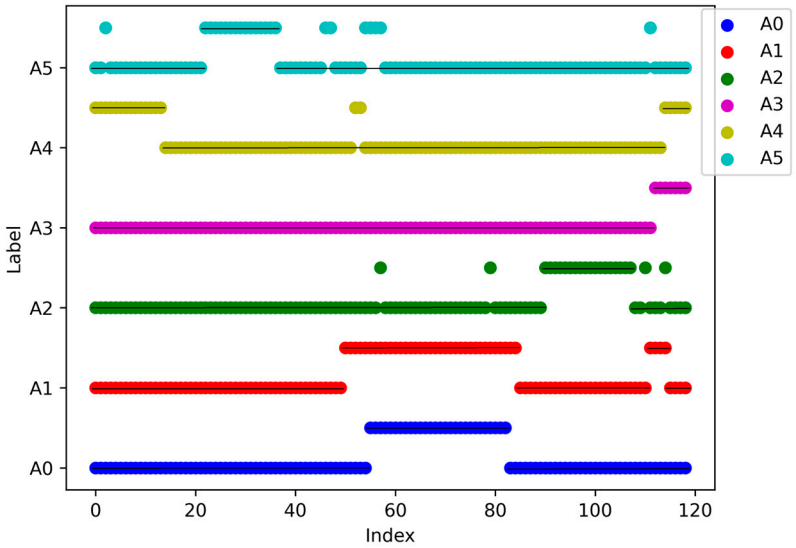


Figure 14. LOS/NLOS discrimination based on information entropy.

3.5. Analysis of Collaborative Optimization

To verify the validity of the factor graph fusion framework based on the resilient mechanism proposed in this paper, we compared and analyzed it with four other methods. The first method is the least squares positioning method (LS) based solely on UWB. The second method is the UWB/INS fusion method (EKF) based on the extended Kalman filter. The third method is a LiDAR/INS fusion method (LIU), where UWB information is only used for initialization and is not integrated into the system afterward. The fourth method is based on the traditional LiDAR/INS/UWB tightly coupled approach, where UWB information is integrated into the positioning system (LIUT).

Figure 15 shows a comparison of the positioning results from different methods. It should be noted that, due to the spatial distribution of UWB anchor stations and the influence of obstacles in the parking lot, when the movement range of the quadruped robot extends from the 1st to the 4th, the quality of the UWB ranging signal degrades, introducing more NLOS errors and causing significant jumps in the UWB positioning results. As a result, the LS and EKF results are only shown for part of the area. However, the subsequent positioning accuracy analysis will be based on the positioning results corresponding to the time stamps. In the fourth to first stage, where the number and distribution of anchor stations are favorable, LS and EKF produced good positioning results. While LS shows some jumps due to noise interference from a single UWB solution, both methods maintained accurate positioning within a small range of the first evaluation point. The LIU algorithm exhibits a relatively smooth overall trajectory; however, it suffers from gradual drift due to the lack of absolute information injection and the accumulation of inherent errors from the laser and IMU sensors. In contrast, the LIUT method successfully integrates UWB

ranging information, receiving high-quality data during the transition from the fourth to the first stage, which helps suppress cumulative errors and maintain high positioning accuracy. Nevertheless, as the robot becomes increasingly affected by NLOS conditions, its positioning results begin to diverge from the true motion trajectory. In contrast, our tightly coupled system demonstrates significant advantages by integrating LiDAR, IMU, and UWB data, thereby compensating for the limitations inherent in any single sensor and ensuring smooth positioning even in challenging environments. The introduction of UWB signals allows our method to obtain absolute position information during the initialization phase, facilitating the generation of an accurate absolute trajectory. Furthermore, by flexibly incorporating anchor ranges within the LOS, our approach effectively corrects cumulative errors and enhances system robustness, mitigating the impact of low-quality ranging information on positioning results.

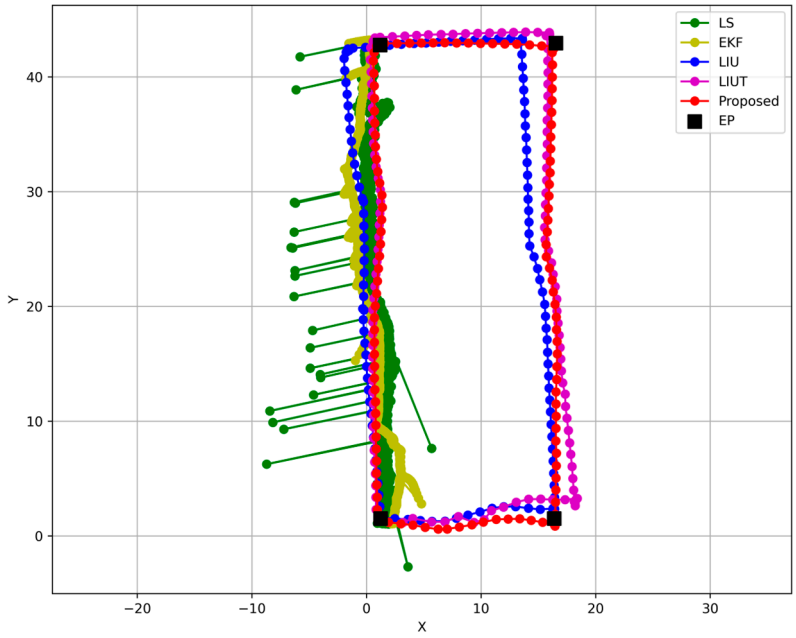


Figure 15. Two-dimensional trajectory comparisons in a parking lot scenario using LS, EKF, LIU, LIUT, and our method.

As seen from the quantitative comparison in Figure 16, the positioning method proposed in this paper significantly improves positioning accuracy. The average error of the LS at the evaluation point is 14.67 m, the EKF is 3.56 m, the LIU is 1.19 m and the LIUT is 1.15 m. In comparison, the average error of our method is the smallest, at 0.36 m. This represents a reduction in positioning errors of 97.53%, 89.85%, 79.25%, and 76.49%, respectively, compared to the aforementioned methods. It is important to note that, due to the challenges in obtaining real-time ground truth in indoor environments, the positioning results are analyzed using static evaluation points. Consequently, when the positioning error at certain evaluation points is substantial, the overall positioning accuracy is significantly impacted. For instance, the LIUT positioning method can maintain satisfactory accuracy in areas where the distribution of anchor points is reasonable and NLOS errors are minimal. However, as cumulative errors gradually increase and the quality of UWB positioning deteriorates, the positioning results tend to drift, leading to substantial errors and a notable increase in overall inaccuracies. In contrast, our method effectively mitigates the influence of NLOS on the system while also eliminating cumulative errors. This approach enhances the robustness of the positioning system, resulting in improved positioning outcomes across multiple evaluation points and ultimately significantly reducing overall positioning errors.

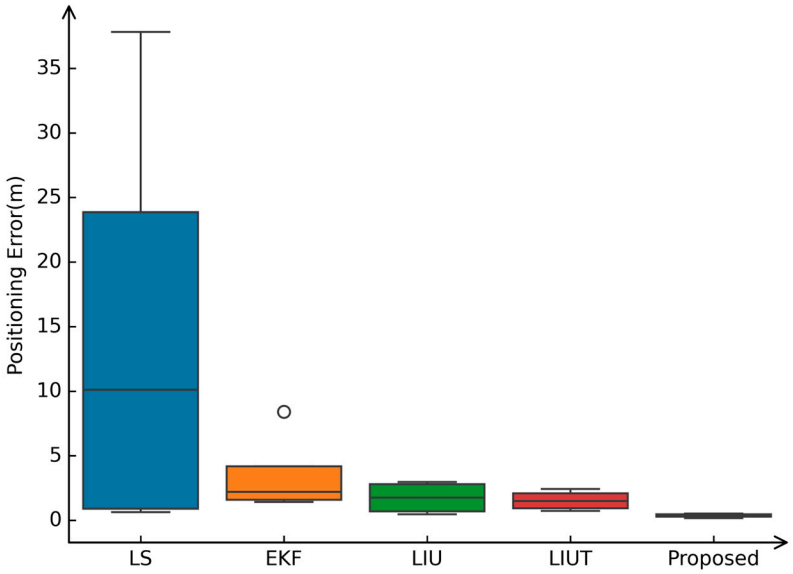


Figure 16. Comparison of error results of different positioning algorithms.

Since the UWB anchor stations in the experiment are at the same height, we cannot analyze the changes in LS and EKF on the Z-axis. As shown in Figures 17 and 18, when the quadruped robot system began to move, the height error of LIU gradually increased, with a maximum drift of 5.43 m. The height error only decreased back to the initial 0.6 m when the robot moved back and point cloud registration occurred. Therefore, in a long-distance environment without loops, the error will continue to accumulate. UWB information was introduced into LIUT, which helped to some extent in suppressing height drift. However, the system robustness deteriorated when NLOS were received, resulting in a maximum drift of 5.846 m. In contrast, the proposed method demonstrates good robustness, significantly constraining height drift, with a maximum drift of only 1.05 m.

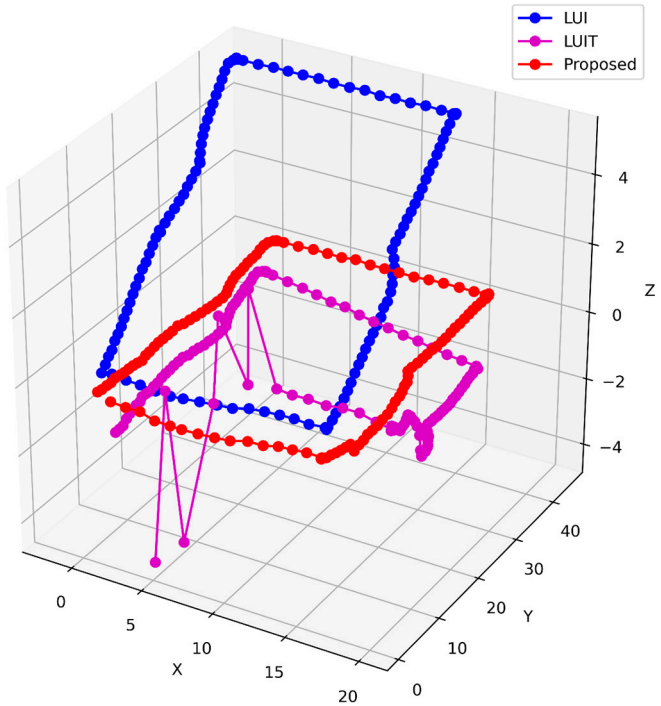


Figure 17. Three-dimensional trajectory coparions in parking lot scenario using LIU, LIUT and our method.

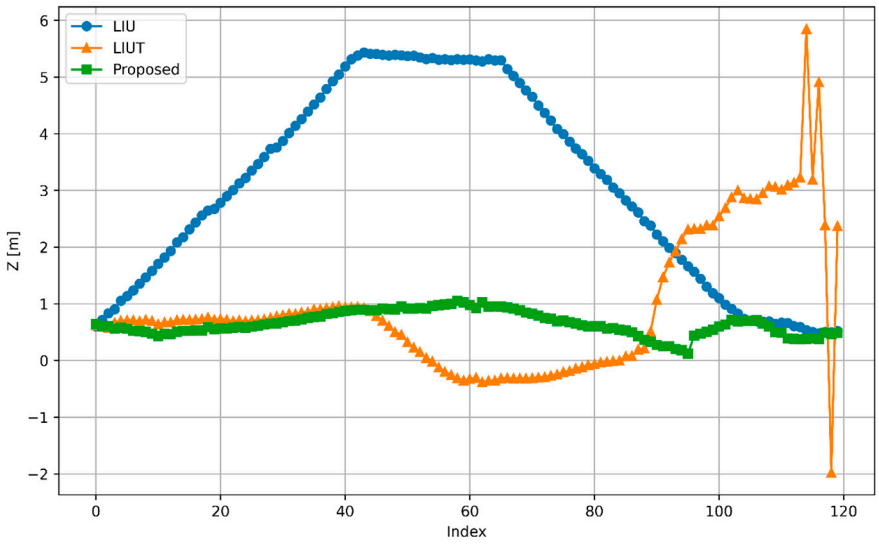


Figure 18. Comparison of z-axis results of different positioning algorithms.

Additionally, an ablation analysis was conducted to further assess the effects of each module individually. The results show that introducing the point cloud segmentation module significantly improves point cloud processing speed, with single-frame processing times reduced by 17.2 ms when using 16 threads, and the positioning error reduced by 4.72 m (see Figures 8 and 9). In the UWB-based positioning system, degraded signal quality can lead to jumps or interruptions in positioning (see Figure 16). Furthermore, for the LIUT algorithm, the indiscriminate fusion of UWB data resulted in an inability to correct LiDAR/IMU accumulated errors, even introducing new errors, which led to a divergence trend in the positioning results (see Figure 15). In contrast, the proposed method demonstrates optimal performance, achieving a positioning accuracy of 0.36 m, which is exceptional for maintaining both positioning accuracy and stability.

3.6. Laboratory Experiment Results

To further validate the effectiveness of the proposed algorithm in complex indoor environments, experiments were conducted on the third floor of the comprehensive experimental building at Sipai Lou Campus, Southeast University. The robot navigated through a diverse experimental area that included interconnected scenes such as long corridors, halls, and balconies. Notably, the long corridors feature parallel walls and consistent geometry, leading to LiDAR-extracted features that are often monotonous and repetitive, thereby lacking sufficient distinctive characteristics for accurate matching. The narrowest corridor, measuring only one meter wide, presents severely degraded walls that pose significant challenges for LiDAR performance. Additionally, NLOS interference due to wall and room occlusions is notably pronounced.

The experimental environment and positioning results are illustrated in Figure 19, where the red triangles represent UWB anchor points, the blue circles denote the quadruped robot, and the red curve indicates the trajectory result. The findings demonstrate that existing UWB localization algorithms struggle in this environment due to a lack of sufficient and effective UWB ranging information. Furthermore, the LIU algorithm fails to extract significant features for point cloud registration, resulting in trajectory jumps in the long corridor and an inability to return to the original region. Although the LUIT algorithm initially received a sufficient number of UWB signals, which helped suppress odometer errors and maintain a consistent trajectory, the increasing influence of low-quality ranging signals led to gradual cumulative errors and significant drift in the final trajectory. In contrast, while our method exhibits slight offsets in the narrow corridor, the incorporation of high-quality multi-sensor information enables the positioning system to effectively

navigate these challenges, successfully completing the global positioning task in this demanding indoor environment and ensuring smooth positioning results.

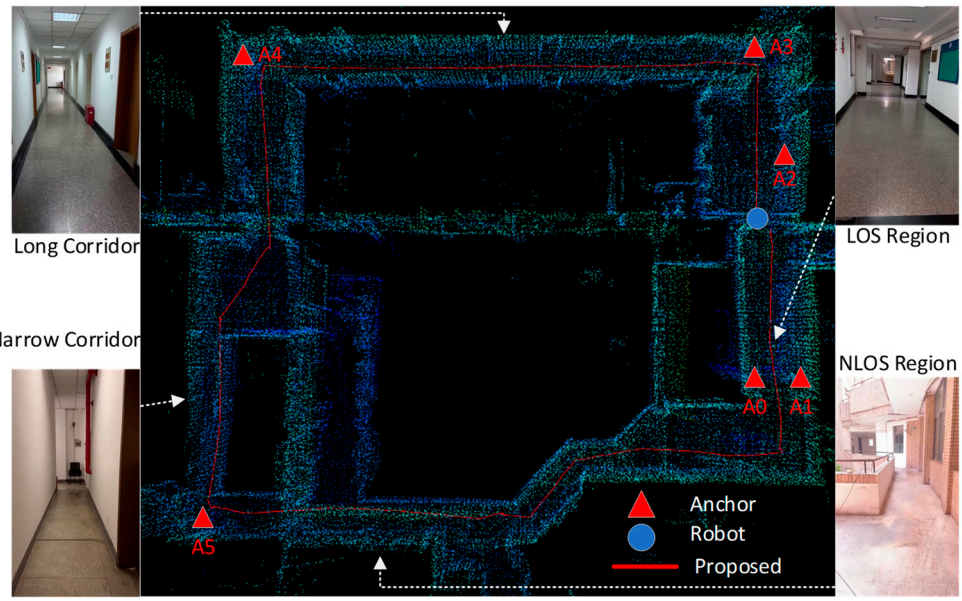


Figure 19. Indoor complex environment and corresponding positioning results.

3.7. Real-Time Performance

In real-world environments, real-time positioning is essential for quadruped robots to perform autonomous tasks. The system's efficiency is evaluated on the robot's industrial computer, and the time consumption of each module is shown in Table 2. The lidar point cloud processing module removes a large number of invalid point clouds, effectively reducing the computational load for subsequent point cloud frame matching, with a total processing time of 16.2 ms. The UWB ranging value processing module, which primarily involves NLOS judgment and outlier removal, takes only 0.01 ms due to the small amount of computation required for the entropy-based evaluation. The factor graph optimization module, which incorporates a resilient mechanism, does not add all information to the factor graph fusion model, thus reducing the time required for optimization to 1.22 ms. Additionally, multi-threading technology is employed to process these modules' computing tasks in parallel, ensuring higher computational efficiency. In summary, our positioning system is well-suited for real-time positioning applications of quadruped robots.

Table 2. Average Time Consumption of Each Module.

Module	Time Consuming (ms)
Lidar point process	16.2
UWB range process	0.01
Factor graph optimization	1.22

4. Conclusions

For accurate and reliable positioning of quadruped robots in GNSS-denied environments, this paper proposes a tightly coupled LiDAR/UWB/INS navigation system based on a resilient factor graph. Compared with existing methods, our approach demonstrates superior robustness and precision. By constructing a point cloud segmentation and constraint module, we minimized elevation errors and reduced the computational load. By incorporating information entropy into NLOS detection, the quality of UWB data in multi-obstacle environments is ensured. Additionally, the resilient mechanism and tightly coupled factor graph model guarantee both accuracy and real-time location updates. To verify the effectiveness of the proposed algorithm on the quadruped robot platform, tests

were conducted in a real-world scenario within an indoor underground parking lot. The experimental results validate the superiority of our method and offer a viable solution for robot positioning in GNSS-denied environments.

Informative semantic data are crucial for the continuous operation of a positioning system. One of our future research goals is to focus on indoor semantic information to explore the impact on mapping, positioning, and navigation. We will also focus on multi-robot navigation and will further optimize positioning results by shared information between multiple robots, aiming to better meet the continuous positioning needs of quadruped robots.

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

Funding: This work is supported in part by the foundation of National Natural Science Foundation of China, grant number 62073078; Postgraduate Research & Practice Innovation Program of Jiangsu Province of China, grant number SJCX24_0068.

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. Meng, X.; Wang, S.; Cao, Z.; Zhang, L. A Review of Quadruped Robots and Environment Perception. In Proceedings of the 2016 35th Chinese Control Conference (CCC), Chengdu, China, 27–29 July 2016; pp. 6350–6356.
2. Afsari, K.; Halder, S.; King, R.; Thabet, W.; Serdakowski, J.; DeVito, S.; Ensafi, M.; Lopez, J. Identification of Indicators for Effectiveness Evaluation of Four-Legged Robots in Automated Construction Progress Monitoring. In Proceedings of the Construction Research Congress 2022, Arlington, VA, USA, 9–12 March 2022; pp. 610–620.
3. Halder, S.; Afsari, K.; Serdakowski, J.; DeVito, S. A Methodology for BIM-Enabled Automated Reality Capture in Construction Inspection with Quadruped Robots. In Proceedings of the ISARC International Symposium on Automation and Robotics in Construction, Dubai, United Arab Emirates, 2–4 November 2021; Volume 38, pp. 17–24.
4. Biswal, P.; Mohanty, P.K. Development of Quadruped Walking Robots: A Review. *Ain Shams Eng. J.* **2021**, *12*, 2017–2031. [CrossRef]
5. Mantha, B.R.K.; de Soto, B. Designing a Reliable Fiducial Marker Network for Autonomous Indoor Robot Navigation. In Proceedings of the 36th international symposium on automation and robotics in construction (ISARC), Banff, AB, Canada, 21–24 May 2019; pp. 74–81.
6. Kunhoth, J.; Karkar, A.; Al-Maadeed, S.; Al-Ali, A. Indoor Positioning and Wayfinding Systems: A Survey. *Hum.-Centric Comput. Inf. Sci.* **2020**, *10*, 18. [CrossRef]
7. Liang, S.; Cao, Z.; Guan, P.; Wang, C.; Yu, J.; Wang, S. A Novel Sparse Geometric 3-d Lidar Odometry Approach. *IEEE Syst. J.* **2020**, *15*, 1390–1400. [CrossRef]
8. Wang, Q.; Tan, Y.; Mei, Z. Computational Methods of Acquisition and Processing of 3D Point Cloud Data for Construction Applications. *Arch. Comput. Methods Eng.* **2020**, *27*, 479–499. [CrossRef]
9. Li, K.; Li, M.; Hanebeck, U.D. Towards High-Performance Solid-State-Lidar-Inertial Odometry and Mapping. *IEEE Robot. Autom. Lett.* **2021**, *6*, 5167–5174. [CrossRef]
10. Rummelhard, L.; Paigwar, A.; Nègre, A.; Laugier, C. Ground Estimation and Point Cloud Segmentation Using Spatiotemporal Conditional Random Field. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017; pp. 1105–1110.
11. Zhang, M.; Morris, D.D.; Fu, R. Ground Segmentation Based on Loopy Belief Propagation for Sparse 3D Point Clouds. In Proceedings of the 2015 International Conference on 3D Vision, Lyon, France, 19–22 October 2015; pp. 615–622.
12. Mongus, D.; Lukač, N.; Žalik, B. Ground and Building Extraction from LiDAR Data Based on Differential Morphological Profiles and Locally Fitted Surfaces. *ISPRS J. Photogramm. Remote Sens.* **2014**, *93*, 145–156. [CrossRef]
13. 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.

14. Liu, X.; Zhang, L.; Qin, S.; Tian, D.; Ouyang, S.; Chen, C. Optimized LOAM Using Ground Plane Constraints and SegMatch-Based Loop Detection. *Sensors* **2019**, *19*, 5419. [CrossRef]
15. Lim, H.; Oh, M.; Myung, H. Patchwork: Concentric Zone-Based Region-Wise Ground Segmentation with Ground Likelihood Estimation Using a 3D LiDAR Sensor. *IEEE Robot. Autom. Lett.* **2021**, *6*, 6458–6465. [CrossRef]
16. Seo, D.-U.; Lim, H.; Lee, S.; Myung, H. PaGO-LOAM: Robust Ground-Optimized LiDAR Odometry. In Proceedings of the 2022 19th International Conference on Ubiquitous Robots (UR), Jeju, Republic of Korea, 4–6 July 2022; pp. 1–7.
17. Koide, K.; Miura, J.; Menegatti, E. A Portable Three-Dimensional LIDAR-Based System for Long-Term and Wide-Area People Behavior Measurement. *Int. J. Adv. Robot. Syst.* **2019**, *16*, 1729881419841532. [CrossRef]
18. 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.
19. Tinchev, G.; Penate-Sanchez, A.; Fallon, M. Learning to See the Wood for the Trees: Deep Laser Localization in Urban and Natural Environments on a CPU. *IEEE Robot. Autom. Lett.* **2019**, *4*, 1327–1334. [CrossRef]
20. Liu, T.; Liu, J.; Wang, J.; Zhang, H.; Zhang, B.; Ma, Y.; Sun, M.; Lv, Z.; Xu, G. Pseudolites to Support Location Services in Smart Cities: Review and Prospects. *Smart Cities* **2023**, *6*, 2081–2105. [CrossRef]
21. Zhang, X.; Kuang, Y.; Yang, H.; Lu, H.; Yang, Y. UWB Indoor Localization Algorithm Using Firefly of Multistage Optimization on Particle Filter. *J. Sens.* **2021**, *2021*, 1383767. [CrossRef]
22. Khodjaev, J.; Park, Y.; Saeed Malik, A. Survey of NLOS Identification and Error Mitigation Problems in UWB-Based Positioning Algorithms for Dense Environments. *Ann. Telecommun. Des. Télécommun.* **2010**, *65*, 301–311. [CrossRef]
23. Güvenç, I.; Chong, C.-C.; Watanabe, F.; Inamura, H. NLOS Identification and Weighted Least-Squares Localization for UWB Systems Using Multipath Channel Statistics. *EURASIP J. Adv. Signal Process.* **2007**, *2008*, 271984. [CrossRef]
24. Wen, K.; Yu, K.; Li, Y. NLOS Identification and Compensation for UWB Ranging Based on Obstruction Classification. In Proceedings of the 2017 25th European signal processing conference (EUSIPCO), Kos, Greece, 28 August–2 September 2017; pp. 2704–2708.
25. Zhu, X.; Yi, J.; Cheng, J.; He, L. Adapted Error Map Based Mobile Robot UWB Indoor Positioning. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 6336–6350. [CrossRef]
26. Wang, C.; Xu, A.; Kuang, J.; Sui, X.; Hao, Y.; Niu, X. A High-Accuracy Indoor Localization System and Applications Based on Tightly Coupled UWB/INS/Floor Map Integration. *IEEE Sens. J.* **2021**, *21*, 18166–18177. [CrossRef]
27. Wang, C.; Xu, A.; Sui, X.; Hao, Y.; Shi, Z.; Chen, Z. A Seamless Navigation System and Applications for Autonomous Vehicles Using a Tightly Coupled GNSS/UWB/INS/Map Integration Scheme. *Remote Sens.* **2021**, *14*, 27. [CrossRef]
28. Zhang, H.; Wang, Q.; Yan, C.; Xu, J.; Zhang, B. Research on UWB Indoor Positioning Algorithm under the Influence of Human Occlusion and Spatial NLOS. *Remote Sens.* **2022**, *14*, 6338. [CrossRef]
29. Ng, H.-F.; Zhang, G.; Hsu, L.-T. A Computation Effective Range-Based 3D Mapping Aided GNSS with NLOS Correction Method. *J. Navig.* **2020**, *73*, 1202–1222. [CrossRef]
30. Zhao, S.; Huang, B. Trial-and-Error or Avoiding a Guess? Initialization of the Kalman Filter. *Automatica* **2020**, *121*, 109184. [CrossRef]
31. Zhao, S.; Shmaliy, Y.S.; Liu, F. Fast Kalman-like Optimal Unbiased FIR Filtering with Applications. *IEEE Trans. Signal Process.* **2016**, *64*, 2284–2297. [CrossRef]
32. He, G.; Yuan, X.; Zhuang, Y.; Hu, H. An Integrated GNSS/LiDAR-SLAM Pose Estimation Framework for Large-Scale Map Building in Partially GNSS-Denied Environments. *IEEE Trans. Instrum. Meas.* **2020**, *70*, 1–9. [CrossRef]
33. Marković, L.; Kovač, M.; Milijas, R.; Car, M.; Bogdan, S. Error State Extended Kalman Filter Multi-Sensor Fusion for Unmanned Aerial Vehicle Localization in Gps and Magnetometer Denied Indoor Environments. In Proceedings of the 2022 International Conference on Unmanned Aircraft Systems (ICUAS), Dubrovnik, Croatia, 21–24 June 2022; pp. 184–190.
34. Wang, X.; Gao, F.; Huang, J.; Xue, Y. UWB/LiDAR Tightly Coupled Positioning Algorithm Based on ISSA Optimized Particle Filter. *IEEE Sens. J.* **2024**, *24*, 11217–11228. [CrossRef]
35. Zhuang, Y.; Sun, X.; Li, Y.; Huai, J.; Hua, L.; Yang, X.; Cao, X.; Zhang, P.; Cao, Y.; Qi, L.; et al. Multi-Sensor Integrated Navigation/Positioning Systems Using Data Fusion: From Analytics-Based to Learning-Based Approaches. *Inf. Fusion.* **2023**, *95*, 62–90. [CrossRef]
36. Zhou, H.; Yao, Z.; Lu, M. Lidar/UWB Fusion Based SLAM with Anti-Degeneration Capability. *IEEE Trans. Veh. Technol.* **2020**, *70*, 820–830. [CrossRef]
37. Nguyen, T.-M.; Cao, M.; Yuan, S.; Lyu, Y.; Nguyen, T.H.; Xie, L. LIRO: Tightly Coupled Lidar-Inertia-Ranging Odometry. In Proceedings of the 2021 IEEE international conference on robotics and automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 14484–14490.
38. Shannon, C.E. A Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [CrossRef]
39. Gururaj, K.; Rajendra, A.K.; Song, Y.; Law, C.L.; Cai, G. Real-Time Identification of NLOS Range Measurements for Enhanced UWB Localization. In Proceedings of the 2017 international conference on indoor positioning and indoor navigation (IPIN), Sapporo, Japan, 18–21 September 2017; pp. 1–7.
40. Chiang, K.-W.; Tsai, G.-J.; Li, Y.-H.; Li, Y.; El-Sheimy, N. Navigation Engine Design for Automated Driving Using INS/GNSS/3D LiDAR-SLAM and Integrity Assessment. *Remote Sens.* **2020**, *12*, 1564. [CrossRef]

41. He, X.; Pan, S.; Gao, W.; Lu, X. LiDAR-Inertial-GNSS Fusion Positioning System in Urban Environment: Local Accurate Registration and Global Drift-Free. *Remote Sens.* **2022**, *14*, 2104. [CrossRef]
42. Forster, C.; Carlone, L.; Dellaert, F.; Scaramuzza, D. IMU Preintegration on Manifold for Efficient Visual-Inertial Maximum-a-Posteriori Estimation. In Proceedings of the Robotics: Science and Systems XI, Rome, Italy, 13–17 July 2015.