Robust Lidar-Inertial Odometry with Ground Condition Perception and Optimization Algorithm for UGV

Zixu Zhao,1,2*, Yucheng Zhang1, Jinglin Shi1, Long Long1 and Zaiwang Lu1,2

1 Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
2 University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: zhaozixu18z@ict.ac.cn

Abstract: Unmanned ground vehicles (UGVs) are making more and more progress in many application scenarios in recent years, such as exploring unknown wild terrain, working in precision agriculture and serving in emergency rescue. Due to the complex ground conditions and changeable surroundings of these unstructured environments, it is challenging for these UGVs to obtain robust and accurate state estimations by using sensor fusion odometry without prior perception and optimization for specific scenarios. In this paper, based on an error-state Kalman filter (ESKF) fusion model, we propose a robust lidar-inertial odometry with a novel ground condition perception and optimization algorithm specifically designed for UGVs. The probability distribution gained from the raw inertial measurement unit (IMU) measurements during a certain time period and the state estimation of ESKF were both utilized to evaluate the flatness of ground conditions in real-time; then, by analyzing the relationship between the current ground condition and the accuracy of the state estimation, the tightly coupled lidar-inertial odometry was dynamically optimized further by adjusting the related parameters of the processing algorithm of the lidar points to obtain robust and accurate ego-motion state estimations of UGVs. The method was validated in various types of environments with changeable ground conditions, and the robustness and accuracy are shown through the consistent accurate state estimation in different ground conditions compared with the state-of-art lidar-inertial odometry systems.

Keywords: lidar-inertial odometry; ground perception; state estimation; sensor fusion; UGV

1. Introduction

With the rapid development of computer software technology and sensor hardware technology, the state estimation of UGVs utilizing various types of sensors, such as lidar, inertial measurement units (IMUs), stereo cameras and ultra-wide bands (UWBs), has been studied by many researchers during the past decade, and multi-sensor suites and different sensor fusion methods have been proposed that aim to improve the accuracy, efficiency and robustness of UGVs’ ego-motion state estimation [1]. Due to the physical characteristics of lidar-inertial odometry, it has a lower cost of computing resources and less degeneration in tough environments [2] compared with the other types of odometry, such as visual-inertial odometry. In addition, the fusion of lidar and IMU has been noticed and studied further for usage in UGVs in recent years. Furthermore, new types of lidar with a low cost and smaller size have also increasingly arisen in the market, such as solid-state lidar [3,4]. Based on the above analysis, our work mainly revolves around the improvements and optimizations of lidar-inertial odometry used in UGVs.

It is known that lidar odometry may degenerate in situations where large planar areas and few geometry features exist, so inertial information is always used to maintain robustness and boost the frequency of the output for the odometry system. Inertial information can provide valid pose transformation constraints and can be leveraged to compensate for the spinning motion of lidar. Furthermore, it can also be utilized for the
coarse initialization of the lidar point cloud registration algorithm, such as iterative closest point (ICP) [5]. However, the perception of environmental elements of high-frequency inertial information gained from IMU is not taken seriously in many lidar-inertial odometry systems, and the information from lidar and IMU can only be fused through a manner that has fixed parameters and models as the environmental elements change. As a consequence, the adaptability to environments is limited and the accuracy of the state estimation of UGVs cannot stay consistent in changing environments. In this paper, we pay more attention to the potential capacity for the environmental perception of inertial information to advance the performance of lidar-inertial odometry. Since the ground surface is the only environmental element directly and physically contacted by UGVs and a large percentage of lidar points stay on the ground surface, the ground condition plays a key role in affecting the quality of a six degrees of freedom (DoF) ego-motion state estimation of lidar-inertial odometry. The fluctuation of the ground surface can cause changes in UGVs’ inertial information, which can be measured by IMU. As UGVs travel on the ground, with a perception of the ground condition in real-time, the parameters used in the lidar points processing algorithm can be adjusted specifically to adapt to the current environment and obtain more accurate state estimations of UGVs.

The error-state Kalman filter [6] is a common method that formulates the error state between the true state and nominal state as a probability distribution to fuse the state estimation evaluated by different sensors. Since ESKF is able to explain the process of sensor fusion from the perspective of probability, it is well-matched conceptually with the ground perception method proposed in the paper, which is also based on the probability distribution of IMU measurements, and the data flow from the ESKF framework can be reused in the process of the evaluation of the ground condition.

In this paper, a novel ground condition perception and optimization algorithm is presented that aims to enhance the performance of lidar-inertial odometry used in UGVs mainly in terms of robustness and accuracy. Several works were carried out based on the ESKF sensor fusion framework, and a new relationship between the IMU measurements and the accuracy of lidar odometry is founded and leveraged in order to bring the extra improvement in performance. The ground condition was firstly evaluated by the raw inertial information and then modified further according to the correction information from the reused data in ESKF. In addition, a specially designed vector was used to describe the ground conditions. Finally, the lidar points processing module responded to the current ground condition vector and the related parameters were optimized according to a formula. The formula was designed by analyzing the latent relationship between the current ground condition and the accuracy of lidar odometry. To the best of our knowledge, this is the first attempt to take the ground condition sensed by IMU into consideration to optimize the lidar-inertial odometry. The main contributions are introduced as follows.

• The novel ground condition perception and optimization algorithm is presented, and is composed of two methods. The first one is the method of ground condition perception using high-frequency inertial information and the data from the output of ESKF, and the second one is the method of the optimization of lidar points processing according to the output of the ground condition perception module.
• A framework of lidar-inertial odometry based on ESKF is proposed to estimate the state of UGVs’ poses, and the ground condition perception and optimization algorithm is well-embedded in this framework.
• Sufficient experiments were performed to validate the performance in terms of the accuracy and robustness of the system. Besides the open-source dataset, the environment with different ground conditions gathered by ourselves was also considered as part of the experimental dataset.

The rest of this paper is organized as follows. In Section 2, the related studies of lidar-inertial odometry are presented and analyzed. Section 3 talks about the framework and the main modules and algorithms of the odometry system. Section 4 introduces the
experiments conducted in the open dataset and self-gathered dataset. Section 5 gives the conclusion and future work.

2. Related Work

As lidar-inertial odometry becomes increasingly irreplaceable in UGVs, there have been many theoretical studies about the fusion models of lidar and IMU and applied studies about the optimization methods designed for specific scenarios. In this part, based on the general architecture, we mainly focus on the works that aim toward the optimization of the accuracy, robustness and efficiency of lidar-inertial odometry, and the advantages and disadvantages of these methods are analyzed and stated.

The errors of sensor fusion odometry systems are generally thought to come from three sources: sensor noise, extrinsic calibration [7] and degenerated state estimation [8,9]. The sensor noise is always modeled as Gaussian noise or a similar variant, it is handled by the Kalman filter or pose graph in many optimization methods [10,11]. Regarding the extrinsic calibration of sensors, the parameters of the calibration are usually seen as parts of the robot state vector, and these vectors are optimized by various methods to satisfy the objective functions dynamically [12–15]. The degenerated phenomenon of lidar-inertial odometry in changing environments is hard to eliminate, and has attracted many researchers’ attention in recent years. An evaluation methodology of degeneracy was proposed by Ji Zhang [16]; it is used for an optimization-based state estimator. The degeneracy factor used to represent the degenerated directions is derived from the constraints brought by sensors’ measurements, and the final solution of the state estimation is optimized by trading-off the constraints in well-conditioned scenes and degenerated scenes. However, the threshold determining degeneration needs to be set manually, which is not suitable for changing environments. LINS [17] is the first work attempting to build a lidar-inertial odometry leveraging the effect of an iterated error-state Kalman filter, as the uncertainty of the robot state estimation grows with time [18]. A robocentric formulation is proposed to average the noisy measurements of different sensors; then, the degeneracy and divergency can be avoided. There is also a series of advanced lidar-inertial odometry [19–22] developed from the methodology of LINS. These methods show the advantage of the accuracy of pose estimation, but, as the idea of an averaging error is not suitable for severe degeneration, failure solutions may occur and the direction of degeneration cannot be determined in these methods. LIO-SAM [23] is the representative work that utilizes graph optimization methods to avoid the degeneration of lidar-inertial odometry; the state constraints generated in degenerated environments are seen as edges and vertices of the factor graph [24] with lower weights. The graph-based methods rely on the information matrix to measure the quality of state constraints. The drawback is that the information matrix may have singular values and cause a local optimum. These methods mentioned above tend to cope with degeneration by analyzing the uncertainty of the state estimation of lidar-inertial odometry (LIO) systems, but the capacity of the odometry to perceive and adapt to the environment in real-time is not paid enough attention; therefore, the performances in different environments may differ a lot.

As the ground occupies a large part of the lidar scan of the surrounding environment, it is an non-negligible environmental element for LIO systems used in UGV. Despite the general methods mentioned above, several works embedded with ground-optimized methods are proposed to preserve the qualities of state estimation, especially for UGVs [25–29]. LeGO-LOAM [25] is a loosely coupled lidar-inertial odometry that utilizes the technique of ground optimization in various terrain. The sensor noise from the ground is notified, and the solution to eliminate the noise is to refer to the ground separation of lidar point clouds and the two-step optimization method for state estimation: at each step, suitable features of the point cloud are chosen to calculate the corresponding components of the six-degrees-of-freedom poses of the UGV. This shows advantages in terms of efficiency and accuracy in the scenarios where stable ground features exist, but its accuracy may degrade in bumpy road conditions as the ground is not clear enough to be separated.
HDL-Graph-SLAM [26] is a tightly coupled sensor fusion odometry system based on normal-distributions transform (NDT) [30] lidar scan matching and graph optimization. Lidar odometry, IMU measurements, GPS and loop-closing are regarded as the constraints of the poses of moving robots; furthermore, the ground-plane constraint is also added to correct the accumulated rotational error, which is hard to compensated for by loop closure on a plane. It assumes the presence of a unified plane in large indoor public environments, so the application scenarios are limited. These ground optimization methods are mostly designed for environments with a flat ground, and are not suitable for environments with changeable terrains.

Based on the ESKF, which handles the fusion of the measurements of lidar and IMU as the fusion of two probabilistic models, we decided to explore more relationships between lidar and IMU to further increase the accuracy and robustness of lidar-inertial odometry. In some popular and efficient pure lidar odometry systems that are based on the normal distributions of lidar points in voxel grids, such as SuMa [31] and LiTAMIN [32,33], the appropriate voxel size in different environments is pointed out to be a key factor influencing the accuracy of lidar odometry. In this paper, motivated by this mechanism and the idea of ground optimization, ground condition sensing was carried out by analyzing the inertial information, and the optimizing process was performed by adjusting the related parameters of the lidar points processing algorithm. As a consequence, the overall performance of the lidar-inertial odometry is improved.

3. Methodology

The following contents mainly introduce the architecture of the lidar-inertial odometry system and the key methods, including the ground condition perception algorithm, the lidar points processing optimization algorithm and the method used to embed these algorithms into the ESKF framework.

3.1. System Overview

The overall architecture of our system and the pipeline of the dataflow are shown in Figure 1. The system is composed of four functional modules, including the ground condition perception module (Section 3.2), lidar points processing module (Section 3.3), mapping module (Section 3.4) and ESKF fusion module (Section 3.5). The pipeline of the dataflow is shown as the following procedure.

1. The raw measurements from IMU are fed into the ESKF fusion module to propagate the error state of UGV’s poses. At the same time, the inertial measurements are utilized in the ground condition perception module to calculate the ground condition vector. Furthermore, the inertial information is cached in the lidar points processing module for lidar points de-skewing.

2. The lidar scan points from the lidar are sent to the lidar points processing module for downsampling and de-skewing, and the parameters used in the points downsampling are determined by the ground condition vectors from the ground condition perception module. The optimized lidar scan points are produced during this procedure.

3. The optimized lidar points, the local map maintained by the mapping module and the lidar points transformation from the ESKF optimization are used when performing the point-to-plane error computation, and the parameters used in this procedure are also dynamically adjusted according to the ground condition vectors from the ground condition perception module.

4. ESKF optimization utilizes the point-to-plane error and the state propagated by IMU measurements to update the error state iteratively until the convergence is achieved. The final output of the odometry is gained from the state vector maintained by ESKF. At the same time, the state vector is used to transform the lidar scan points serving the mapping module; it can also be used in the ground condition perception for correction and calibration.
3.2. Ground Condition Perception

The raw measurements of IMU were utilized to evaluate the condition of the ground, which the moving UGV is currently in contact with. The raw inertial information $u$ is composed of the angular velocity $\omega$ around three axes and the linear acceleration $a$ on the three axes.

$$u = [\omega \ a], \omega = [\omega_x \ \omega_y \ \omega_z], \ a = [a_x \ a_y \ a_z]$$

(1)

Suppose that the UGV moves straightforward, since the rise and fall of the ground may mainly cause the variance in angular velocity. Only the angular velocity $\omega$ is considered to measure the ground condition. In order to evaluate the flatness of the ground, the probability distribution of inertial rotational information from IMU measurements during a certain period of time was computed as the following procedure.

For a certain time period $\Delta t$ between two consecutive lidar scans, suppose that there are $n$ measurements of angular velocity. The average angular velocity $\bar{\omega}$ is computed. In order to eliminate the noise brought by IMU measurements and the effect of the random walk of IMU’s bias, the Gaussian white noise $n_\omega$ and the bias of angular velocity $b_\omega$ evaluated by the ESKF fusion module (Section 3.5) at the current time are utilized.

$$\bar{\omega} = \sum_{i=1}^{n} \omega_i - b_\omega - n_\omega = [\bar{\omega}_x, \ \bar{\omega}_y, \ \bar{\omega}_z]$$

(2)

The covariance matrix $P_\omega$ of these measurements is also computed.

$$P_\omega = \begin{bmatrix}
\text{cov}(\omega_x, \omega_x) & \text{cov}(\omega_x, \omega_y) & \text{cov}(\omega_x, \omega_z) \\
\text{cov}(\omega_y, \omega_x) & \text{cov}(\omega_y, \omega_y) & \text{cov}(\omega_y, \omega_z) \\
\text{cov}(\omega_z, \omega_x) & \text{cov}(\omega_z, \omega_y) & \text{cov}(\omega_z, \omega_z)
\end{bmatrix}$$

$$\text{cov}(X, Y) = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{n}$$

(3)

Then, the probability distribution of the measurements during the time period $\Delta t$ can be obtained as $N(\bar{\omega}, P_\omega)$. However, since UGV's do not always move straightforwardly, the uncertainty of $N$ is not caused entirely by the variance in the ground surface. In order to eliminate the effect of the active steering of UGVs, principle component analysis (PCA) was performed on the covariance matrix $P_\omega$. 

Figure 1. The overall architecture of the lidar-inertial odometry system proposed in this paper.
\[ P_\omega = Q \sum Q^T Q = \begin{bmatrix} v1 & v2 & v3 \end{bmatrix}, \sum = \begin{bmatrix} \lambda1 & 0 & 0 \\ 0 & \lambda2 & 0 \\ 0 & 0 & \lambda3 \end{bmatrix} \] (4)

\[ \sum \] is the matrix composed of the eigen values, where \( \lambda1 > \lambda2 > \lambda3 \), and \( Q \) is the matrix composed of the eigen vectors of the corresponding eigen values. According to the theory of PCA, the eigen vector that has the largest eigen value is the main component of the covariance matrix. As for \( P_\omega \), \( v1 \) represents the main direction where the orientation of IMU’s measurements varies. If the UGV begins to steer, \( \lambda1 \) will be much larger than \( \lambda2 \) and \( \lambda3 \); in order to remove the influence of steering and ensure that the uncertainty of probability distribution \( N \) is purely caused by the variance in the ground condition, \( P_\omega \) and \( N \) were modified according to the following strategy.

\[ P'_\omega = Q' \sum Q', N'' = N'(\bar{\omega}, P'_\omega) \]

\[ \sum' = \begin{bmatrix} \lambda1' & 0 & 0 \\ 0 & \lambda2 & 0 \\ 0 & 0 & \lambda3 \end{bmatrix} \]

\[ \lambda1' = \begin{cases} (\lambda2 + \lambda3)/2, & \text{if } \lambda1 >> \lambda2, \\ \lambda1, & \text{else} \end{cases} \] (5)

Inspired by the work carried out by LiTAMIN2 [33], which uses symmetric KL divergence to evaluate the difference in two distributions of point clouds, the KL-divergence was also utilized to compare two different distributions of IMU measurements in this paper. We formulated the difference between two series of IMU measurements gained in two consecutive time periods as the symmetric KL divergence of the two probability distributions \( N_1'(\omega_1, P_{\omega_1}) \) and \( N_2'(\omega_2, P_{\omega_2}) \) transformed by Equation (5).

\[ \text{Div}(N_1', N_2') = (\omega_1 - \omega_2)^T (P_{\omega_1} + P_{\omega_2})^{-1} (\omega_1 - \omega_2) + \text{tr}(P_{\omega_1}^{-1} P_{\omega_2}) + \text{tr}(P_{\omega_2}^{-1} P_{\omega_1}) - 2d \] (6)

\( \text{tr}(A) \) represents the trace of matrix \( A \) and \( d \) denotes the dimension of \( \omega \). Since the \( \text{Div}(N_1', N_2') \) indicates the degree of the relative changes in the ground where the UGV travels during two consecutive time periods, it can be used as the signal to activate the optimization of parameters in the lidar points processing module, which will be introduced in detail in Sections 3.3.1 and 3.3.3.

Since the KL divergence only describes the relative changes in the ground condition, the absolute value was addressed by computing the trace of the covariance matrix. The trace is an indicator of the level of the linear transformations contained in the matrix, and is a suitable factor to measure the flatness of the ground. In addition, the linear velocity of UGV may vary all of the time during the process of the evaluation of the ground condition. The results may be inconsistent under changeable linear velocities. To address this problem and obtain consistent results, the most recent linear velocity \( v_{k-1} \) was taken from the output of the EKSF module; it can serve as the denominator of the trace of the covariance matrix \( P_{\omega_k} \) to eliminate the effect of linear velocity. The ground condition vector \( GC_k \) is finally given as follows, including the absolute value and the relative value. \( k \) represents the index of the lidar scan.

\[ GC_k = \left( \frac{\text{tr}(P_{\omega_k})}{v_{k-1}}, \text{Div}(N_{k-1}', N_k') \right) \] (7)
3.3. Lidar Points Processing Module
3.3.1. Lidar Points Downsampling

In order to maintain the real-time performance of the odometry system, the lidar points were downsampled to improve the efficiency. The basic rate of downsampling was 1/4, which means that only one point was extracted for every four lidar points, which are temporally consecutive in the raw lidar scan. In this paper, due to the change in the ground condition, the downsampling rate of the lidar points lying on the ground should be dynamic; this is relevant to the ground condition vector computed in Equation (7). Firstly, the ground lidar points were segmented from the other points in a single lidar scan. Since the ground condition vector was computed from the inertial information of IMU, it can only represent the condition of the ground where the UGV directly contacts. Therefore, the segmented ground points are restricted to the region where UGV may have travelled on. The range of the ground lidar points is shown as the green rectangle region in Figure 2, and the height of points in the ground region should also be restricted to ±1 m. Then, the lidar points of the ground region were downsampled according to the ground condition vector $GC_k(tr(P_{ωk}), Div(N_{k-1}, N_k))$, and the downsampling rate $a_k$ of the $k$th lidar scan was optimized according to the following strategy.

$$a_k = \begin{cases} \frac{a_{k-1}}{2}, & \text{if } GC_k(1) < T_{activate} \\ \frac{1}{2} - \frac{3}{2} \min GC(0)+2, & \text{else} \end{cases}$$

Figure 2. The range of the ground lidar points. $v$ is the current velocity from the state vector evaluated by ESKF, $Δt$ is the time period utilized to compute the distribution of IMU measurements and the width of the ground region is the width of the UGV.

Regarding the description in Section 3.2, $GC_k(1) = Div(N_{k-1}, N_k)$ is regarded as the activating signal for the optimization of the downsampling rate and $T_{activate}$ is the threshold of the activating signal. When $GC_k(1)$ is relatively small, the downsampling rate is unchanged as the ground condition does not change too much; otherwise, it is adjusted according to $GC_k(0) = \frac{tr(P_{ωk})}{N_k-1}$. Suppose that the range of $GC_k(0)$ is $[0, G_{max}]$. The $G_{max}$ is an empirical value that is based on the engineering practice in real world environments. When $GC_k(0) = 0$, which implies that the ground is a plane through the evaluation of ground perception, $a_k$ takes the minimum value of 1/4, which is the same as the other region of the lidar scan, because the lidar points in the planar ground will not contribute too much to the state estimation of odometry. If $GC_k(0)$ is close to the upper bound, which means that the ground condition is rough enough, $a_k$ takes the maximum value of 1, which represents that none of the points taken from the ground region are ignored. The details about the ground are saved so that more useful information from the ground points can be utilized.
for state estimation. We utilized an exponential function to represent the relationship between \( GC_k(0) \) and \( a_k \). The changing rate of \( a_k \) is small when \( GC_k(0) \) is close to zero, as the downsampling rate is not taken seriously when the ground is relatively flat.

### 3.3.2. Lidar Points De-Skewing

A lidar scan consists of an accumulated set of points received during a short time period, but the lidar is generally considered to be static during the time period. In order to compensate for the small error led by the lidar moving, IMU measurements can be used to finely tune the positions of lidar points in the current lidar frame. Suppose that a lidar scan begins at \( t_0 \), the timestamp of the \( j \)th lidar point in the lidar scan is represented as \( t_j \) and the original position of the point in the current lidar frame is \( p_j \). The relative transformation per lidar point is composed of the rotation part \( R_{j,t_0} \) and the translation part \( P_{j,t_0} \). To simplify the representation, we used the form of the continuous function to express the computation of \( R_{j,t_0} \) and \( P_{j,t_0} \); in fact, they are accumulated by the discrete IMU measurements. \( \hat{b}_{\omega t_j}, n_{\omega t_j}, b_{at}, n_{at}, v_t \) are from the state vector estimated by ESKF most recently in Section 3.5.

\[
R_{j,t_0} = \text{EulerToRotationMatrix}(\int_{t_0}^{t_j} (\omega_t - b_{\omega t} - n_{\omega t})dt)
\]

\[
P_{j,t_0} = \int_{t_0}^{t_j} v_t dt + \int_{t_0}^{t_j} (a_t - b_{at} - n_{at}) dt
\]

Then, the tuned position \( p'_j \) per lidar point is computed.

\[
p'_j = R_{j,t_0}p_j + P_{j,t_0}
\]

### 3.3.3. Point-to-Plane Error Computation

In order to compute the point-to-plane errors used in the iterated ESKF optimization (Section 3.5.3), a small plane \( s_i \) was fitted by \( n \) map points; it consists of the center position \( Wq_i \) in the world frame and normal vector \( n_i \) representing the direction of the plane. These map points were founded by searching for the \( n \) points in the local map that are closest to the projection of the related point \( p_j \) in the current lidar scan. The map construction is introduced in Section 3.4, and the point-to-plane error \( E(p_j, s_i) \) for each lidar point \( p_j \) in the lidar frame and its corresponding plane \( s_i \) in the map was computed as below.

\[
E(p_j, s_i) = n_i(\text{^W}R_{j}^{-1}\text{^L}R_{L}p_j + \text{^L}P_L) + \text{^W}P_L - \text{^W}q_i
\]

\( \text{^L}R_L \) and \( \text{^L}P_L \) are the transformation matrices from the lidar frame (denoted as \( L \)) to the IMU frame (denoted as \( I \)) that were calibrated previously. \( \text{^W}R_{j}^{-1}\text{^L}R_{L} \) and \( \text{^W}P_L \) represent the transformation matrices from the IMU frame to the world frame (denoted as \( W \)) that were taken from the state vector of IMU in the iterated ESKF optimization module and \( \text{^L}P_L \) is the position of the \( j \)th lidar point in the current lidar frame.

However, the number of considered map points used to fit the plane may have an influence on the result of the error computation. Figure 3 gives an example of the point-to-plane error computation in different ground conditions. In the flat area, the local plane \( s_i \) is fitted correctly to reflect the local direction of the ground, and the normal vector of the fitted plane stays consistent as the number of considered map points changes (\( s_i \approx s_i' \)). However, in the bumpy area where the ground surface is uneven, excessive points used to fit the local plane will result in different point-to-plane errors as the direction of the local plane varies too much with the number of considered map points (\( s_i \neq s_i' \)). The excessive points cannot reflect the details of the local plane on an uneven ground surface, especially the direction represented by the normal vector. Consequently, the accuracy of the state estimation may be influenced. In order to cope with this problem, in the environments with changeable ground conditions, the points of the \( k \)th lidar scan used to compute point-to-plane errors need to be optimized using the following procedure. (i) The downsampling rate \( \alpha_k \) of the points in the ground region is increased according to Equation (8) to keep the details of the ground. (ii) The
The number of map points $N_k$ used to fit the plane is decreased to obtain a more accurate local plane according to Equation (12). The basic number of considered map points in the flat area is 8, and it will decrease linearly as the flatness of the ground condition represented by $GC_k(0)$ grows, where the minimal number is restricted to 5. $\text{Round}(x)$ is the rounding function.

\[
N_k = \begin{cases} 
N_{k-1}, & \text{if } GC_k(1) < T_{\text{activate}} \\
\text{Round}(8 - \frac{3GC_k(0)}{G_{\text{max}}}), & \text{else}
\end{cases}
\]

(12)

---

**Figure 3.** Point-to-plane error computation in different ground conditions. (a) The error computed in the ground of flat area. (b) The error computed in the ground of bumpy area. The blue line is the lidar points in the map, the red line is the lidar points in current lidar scan. $s_i$ is the local plane fitted by the nearest 5 map points from the point $p_i$, and $s'_i$ is fitted by 3 points. $E_i$ is the associated point-to-plane error represented as the length of the related black line. The enlarged drawing at the top highlights the local plane fitted by the map points of different numbers. (c) The picture of real-world ground condition containing the planar area and bumpy area. (d) The related current lidar points and the surrounding local map points of picture (c), where the pink points and the blue points correspond to the map points in the flat area and bumpy area, respectively, and the white points are the lidar points in the current scan.

### 3.4. Mapping

The map maintained by the odometry system proposed in this paper is the collection of selected lidar points in the world frame during the whole process of the UGV’s state estimation. The map representation and construction methods were adopted from the work carried out by FAST-LIO2 [21]. Based on the incremental data structure of ikd-Tree used to manage the global map, the speed of the neighbor search of the nearest points is faster than the traditional kd-Tree, and the map updating (points insertion and deletion) also has a low time complexity of $O(\text{inlog } n)$. These characteristics of ikd-Tree can guarantee the efficiency of the odometry system.
3.5. ESKF Fusion Module

The measurements from IMU and lidar were fused through a manner maintained by ESKF. Since ESKF has been well-studied in recent works of lidar-inertial odometry, only a brief process of ESKF is introduced in this section.

3.5.1. Error State Representation

ESKF maintains the error state vector $\tilde{x}$, which is centric on the pose of IMU, and the error state was computed as below. $\boxplus/\boxminus$ are the operator of addition and subtraction on the state vector.

$$\tilde{x} = x \boxminus \hat{x}$$  \hspace{1cm} (13)

$x$ is the true state vector of IMU and composed of the elements shown below, and $\hat{x}$ is the predicted state vector propagated by IMU measurements.

$$x = \begin{bmatrix} W_R I & W_P I & W_v I & W_a b \omega & b_a \omega & b_a \end{bmatrix}$$  \hspace{1cm} (14)

$W_R I$ and $W_P I$ represent the rotation and translation from the IMU frame to the world frame, $W_v I$ denotes the linear velocity of IMU and $b_a \omega$ and $b_a$ represent the bias of IMU, which observes the distributions of random walk noise.

3.5.2. State Propagation by IMU Measurements

Suppose that $k$ denotes the index of lidar scans and $i$ denotes the index of IMU measurements between two consecutive lidar scans, given the last state vector of IMU $x_{k-1}$; then, the predicted state vector at the time of current lidar $\hat{x}_k$ can be computed recursively by Equation (15) utilizing the inertial information between two consecutive lidar scans. $\hat{x}_i$ is the predicted state vector at the time of the $i$th IMU measurement, and the recursive formula $f(\hat{x}_i, u_i, w_i)$ of the predicted state vector is described below. $u_i$ is the IMU input and $w_i$ is the noise vector composed of IMU white noise and IMU bias noise.

$$\hat{x}_{i+1} = \hat{x}_i \boxplus (\Delta t f(\hat{x}_i, u_i, w_i))$$
$$u_i = \begin{bmatrix} \omega_i & a_i \end{bmatrix}, w_i = \begin{bmatrix} n_{\omega_i} & n_{a_i} & n_{b_{\omega_i}} & n_{b_a} \end{bmatrix}$$
$$f(\hat{x}_i, u_i, w_i) = \begin{bmatrix} \omega_i - \hat{b}_{\omega_i} - \hat{n}_{\omega_i} \\ W_v I (u_i - \hat{b}_a - \hat{n}_a) \\ n_{b_{\omega_i}} \\ n_{b_a} \end{bmatrix}$$  \hspace{1cm} (15)

Let $\hat{P}_k$ represent the covariance matrix measuring the uncertainty of the predicted state $\hat{x}_k$, starting from $\hat{P}_{k-1}$. $\hat{P}_k$ can be also propagated recursively using the following formula.

$$\hat{P}_{i+1} = F_{x_i} \hat{P}_i F_{x_i}^T + F_{w_i} Q_i F_{w_i}^T$$  \hspace{1cm} (16)

The details on $F_{x_i}$, $F_{w_i}$ and $Q_i$ are introduced in the supplementary material of R²LIVE [20]. $F_{x_i}$ and $F_{w_i}$ are the linearized propagation matrix describing the relationship of the state covariance between two consecutive IMU measurements, and $Q_i$ is the noise covariance matrix created during the process of IMU measurements.

Using the covariance matrix $\hat{P}_k$ of the predicted state, the prior probability distribution of $\tilde{x}_k$ can be represented as follows.

$$\tilde{x}_k \sim \mathcal{N}(0, \hat{P}_k)$$  \hspace{1cm} (17)
3.5.3. Iterated ESKF Optimization by Lidar Measurements

Based on the point-to-plane error computed by Equation (11) in Section 3.3.3, the lidar measurements can provide the posterior probability distribution. Let \( h_i(x_k) \) denote the measurement model of the \( i \)th lidar point for the given state \( x_k \). It is linearized at \( \hat{x}_k \) using the method of first-order Taylor expansion. \( H_i \) is the Jacobian matrix with respect to the error state \( \tilde{x}_k \) and \( v_i \) is created by the raw noise of lidar measurements. It can be represented as the posterior probability distribution at 0 with the covariance \( R_i \).

\[
\begin{align*}
    h_i(x_k) &= h_i(\hat{x}_k \boxplus \tilde{x}_k) = h_i(\hat{x}_k) + H_i \tilde{x}_k + v_i = E(p_i, s_i) + H_i \tilde{x}_k + v_i = 0 \\
    v_i &\sim \mathcal{N}(0, R_i) = -E(p_i, s_i) - H_i \tilde{x}_k
\end{align*}
\]

(18)

The ESKF update was performed iteratively to minimize the cost function represented as the maximum a posteriori estimation of the error state \( \tilde{x}_k \); it is composed of the prior probability distribution of \( \tilde{x}_k \) computed by IMU propagation and the posterior probability distribution of \( v_i \) led by lidar measurements. The optimizing problem can be represented as below. \( n \) is the number of lidar points in the \( k \)th lidar scan and \( \mathcal{H} \) is the Jacobian matrix of \( (\hat{x}_k \boxplus \tilde{x}_k) \boxplus \hat{x}_{k_{\text{initial}}} \) with respect to \( \tilde{x}_k \). \( \hat{x}_{k_{\text{initial}}} \) is the initial value of \( \hat{x}_k \) taken from IMU propagation.

\[
\begin{align*}
    \min_{\hat{x}_k} \left( \|\tilde{x}_k\|_H^2 + \sum_{i=1}^{n} \|v_i\|_R_i^2 \right) = \\
    \min_{\hat{x}_k} \left( \|\hat{x}_k \boxplus \hat{x}_{k_{\text{initial}}} \boxplus \mathcal{H} \tilde{x}_k\|_{\mathcal{H}^T H^{-1}}^2 + \sum_{i=1}^{n} \|E(p_i, s_i) + H_i \tilde{x}_k\|_R_i^2 \right)
\end{align*}
\]

(19)

The problem is solved by computing the Kalman gain \( K \) and \( \hat{x}_k \) iteratively. \( H, R \) and \( E_k \) are composed of \( H_i, R_i \) and \( E(p_i, s_i) \), respectively.

\[
K = \left( H^T R^{-1} H + (H^{-1} \mathcal{H}_k H^{-1})^{-1} \right)^{-1} H^T R^{-1} \\
\hat{x}_k = \hat{x}_k \boxplus (-K E_k - (I - K \mathcal{H}) H^{-1} (\hat{x}_k \boxplus \hat{x}_{k_{\text{initial}}}))
\]

(20)

The optimized state vector \( \hat{x}_{k_{\text{final}}} \) is given as the final output of the odometry system until \( \hat{x}_k \) converges to a certain small range or the number of iterations reaches a maximal threshold.

4. Experiments

The performance of the lidar-inertial odometry system proposed in this paper was tested mainly in terms of accuracy and robustness through a series of sufficient experiments. The effect on the system that the ground condition perception and optimization algorithm causes was analyzed in these experiments. We implemented the odometry in ROS kinetic 16.04 with a normal laptop with 8 GB RAM and Intel i5-9300 CPU. As the paper is focused on UGVs’ state estimation, the datasets used to evaluate the performance were gathered by the sensor-suites equipped on UGV. There are two sources of these datasets, including the open-source dataset used in the experiments of LIO-SAM [23] and LVI-SAM [34], and the dataset gathered by ourselves. The state-of-art lidar-inertial odometry systems used as comparisons were LINS [17], LIO-SAM, Lego-LOAM [23] and FAST-LIO2 [21]; these LIO systems were chosen to cover different types of lidar-inertial fusion methods. LINS is the representative work of tightly coupled lidar-inertial fusion methods using Kalman filter and feature-based method of lidar points registration, LIO-SAM is the representative work of tightly coupled methods using factor graph, Lego-LOAM is the representative work of loosely coupled methods and FAST-LIO2 is the representative work leveraging Kalman filter and direct method of lidar points registration.
4.1. System Performance in Open Dataset

In order to validate the effect of our ground condition perception and optimization algorithm on the precision of lidar-inertial odometry, a series of experiments were performed based on the two park datasets used in LIO-SAM and LVI-SAM. These datasets were gathered by driving a Clearpath UGV equipped with a sensor-suite composed of a Velodyne 16-line lidar, a MicroStrain 3DM-GX5-25 IMU and a Reach MGPS in unstructured areas. The dataset in LIO-SAM is small scale, covering approximately 0.7 km forested hiking trail, whereas the dataset in LVI-SAM is large scale, covering a journey of 4.6 km; there are various road conditions, including asphalt, grassland and dirt-covered trails in this dataset. Both datasets are challenging to LIO systems as the environments comprise changeable environmental elements and road conditions without prior knowledge.

4.1.1. Visualization of the Ground Perception and Optimization Algorithm

In Figure 4, in order to visualize the process of ground perception and optimization, the ground condition vectors of the two datasets are presented according to the perception method in Section 3.2; furthermore, the changing process of the downsampling rate of lidar points and the number of map points used to fit plane, which were introduced in Section 3.3, are also shown in this figure. The flatness of and changes in the ground can be described quantitatively by the \( GC_k(0) \) and \( GC_k(1) \) in the ground condition vector, respectively, the downsampling rate and the number of map points used to fit plane were adjusted as \( GC_k(0) \) varies and \( GC_k(1) \) was utilized as the activating signal of the optimization. We can see from (b) in Figure 4 that the grassland corresponds to bigger value of \( GC_k(0) \) than that of the asphalt pavement, due to the grassland being more rough. As a consequence, the downsampling rate in the grassland was adjusted to be bigger than that in the asphalt pavement to save more details of the ground, and the number of map points used to fit plane was adjusted to be lower than that in the asphalt pavement to avoid incorrectly fitted plane and inconsistent point-to-plane error computation.

4.1.2. Ablation and Comparison Study of the Ground Perception and Optimization Algorithm

Figure 5 shows the trajectories of the datasets evaluated by different methods. We disabled the ground condition optimization algorithm and used consistent downsampling rate of lidar points and unchanged number of map points used to fit plane to see the extra increase in accuracy that the algorithm brings to the LIO system. In addition, the trajectories evaluated by Lego-LOAM, LIO-SAM, LINS and FAST-LIO2 are also included as comparative result. In order to cope with the influence of loop closure and obtain consistent experimental results, we disabled the mechanism of loop closure in these methods. Table 1 shows the overall translational errors represented as the translational root means square error (RMSE) compared with GPS ground truth, and the regions with poor GPS signals were not included during the computation of RMSE. Due to the large-scale dataset starting and ending at the same position, the end-to-end translational and rotational errors were also considered as indices to evaluate the accuracy. It can be noticed that our method without ground optimization shows similar accuracy with LINS and FAST-LIO2 in both datasets; this is due to the fact that all three methods were adopted from iterated Kalman filter. The ground perception and optimization algorithm can significantly improve the accuracy of our LIO system according to the experimental results. In the small-scale dataset, where the ground condition does not change a lot, our system with the ground optimization could achieve similar accuracy of state estimation with LIO-SAM, and the translational RMSE with optimization was approximately 19% lower than the index achieved by the method without optimization, whereas, in the large-scale dataset, where the ground condition changes frequently, the extra enhancement in the ground optimization is more obvious: the RMSE with optimization is 26% lower than the method without optimization, the end-to-end translational error is reduced by 20% and the rotational error is reduced by 40% compared with the method without optimization.
Furthermore, our complete LIO system could achieve the best accuracy compared with the other 4 state-of-art LIO systems in the large-scale dataset.

Figure 4. The changing process of the ground condition vectors, the downsampling rate of lidar points and the number of map points used to fit plane. (a) The situation in small-scale dataset of 500 s, 0.7 km. (b) The situation in large-scale dataset of 2200 s, 4.6 km. The pictures of ground are shown on the upper left of (b) as the visual information is included in the large-scale dataset, and the corresponding lidar point cloud maps are also provided below these pictures to reflect the ground conditions in detail. The picture on the upper right of (b) shows the locations of the ground condition in Google satellite map.
Table 1. Accuracy evaluation of the trajectories estimated by different methods and LIO systems in open dataset.

|                                | Small-Scale Dataset                                      | Large-Scale Dataset                                      |
|--------------------------------|---------------------------------------------------------|----------------------------------------------------------|
|                                | Our Method without Optimization | Our Method with Optimization | LINS | Lego-LOAM | LIO-SAM | FAST-LIO2 |
| Estimated length (m)           | 661.76 | 661.32 | 661.08 | 661.25 | 661.11 | 661.53     |
| Translational RMSE (%)         | 1.49   | 1.21   | 2.34   | 1.57   | 1.12   | 1.52       |

|                                | Our Method without Optimization | Our Method with Optimization | LINS | Lego-LOAM | LIO-SAM | FAST-LIO2 |
| Estimated length (m)           | 4668.22 | 4670.55 | 4668.96 | 4677.23 | 4670.73 | 4667.81     |
| Translational RMSE (%)         | 3.78   | 2.80   | 3.95   | 8.96   | 3.54   | 3.87       |
| End-to-end translational error (m) | 5.25 | 4.22 | 7.37 | 10.44 | 5.48 | 6.54 |
| End-to-end rotational error (m) | 3.43   | 2.07   | 4.80   | 5.17   | 2.18   | 3.66       |

The red text represents the optimal value of the related item.

Figure 5. Trajectories of the datasets estimated by different LIO systems. (a) The trajectories evaluated in small-scale dataset. (b) The trajectories evaluated in large-scale dataset. In the small-scale dataset, GPS ground truth is available everywhere, whereas, in the large-scale dataset, GPS signal is partial available because of the blocking of forests. The ground truth will drift considerably in GPS-denied area.
4.2. System Performance in Self-Gathered Dataset with Changing Ground Conditions

The accuracy and robustness of state estimation of the proposed LIO system were also tested in our self-gathered dataset. The sensor suite has a low cost, consisting of a RoboSense-16 lidar and a LMPS IMU, and the UGV carrying the sensors is a wheeled robot with Ackerman steering structure. The dataset was collected in an agricultural demonstration park, where we controlled the UGV to cross various ground conditions and scenes in the park, including the sidewalk composed of bricks, the bumpy grassland and the road made of asphalt. The travelling distance in each ground condition was approximately similar (approximately 500 m), which makes the dataset suitable for the usage of comparative experiments.

Figure 6 shows the trajectories and the point cloud map produced during the process of UGV traveling in the park aligned with the satellite map in Google Earth using our complete LIO system. First, the pose of the starting point (shown as the yellow star) during the trajectory was recorded using real-time kinematic global positioning system (RTK-GPS), where the starting pose consists of the position (longitude, latitude, altitude) and the angle (roll, pitch, yaw) in WGS-84 (World Geodetic System 1984). Second, the position and the angle of the starting pose were transformed into universal transverse mercator (UTM) grid system) to obtain the transformation matrix. The trajectories and the point cloud map produced by our LIO system can be transformed into UTM using the transformation matrix. Finally, the trajectories and the point cloud map could obtain their positions (longitude, latitude, altitude) in WGS-84 using the inherent transformation between UTM and WGS-84; then, they could be aligned with Google satellite map. The flatness of the ground is also shown through the color of the trajectory in the figure. As the condition of ground varies from flat to rough, the color of the trajectory varies from white to red, representing $GC_k(0)$, which varies from 0 to 0.5. We can see from the figure that the ground condition of the grassland is the roughest, which implies that the larger downsampling rate and the smaller number of map points used to fit plane are needed to advance the accuracy of state estimation. This situation is opposite in asphalt, where the ground is flat. As a result, the large-scale point cloud map is well-matched with the satellite map, which means that the robustness and accuracy of state estimation in long term can be guaranteed; furthermore, the capacity of our system to adapt to the changing ground conditions is also reflected.

The accuracy of state estimation of different methods in the self-gathered dataset is shown in Figure 7. It is represented as the absolute trajectory error (ATE)) of the trajectory sequence with respect to GPS ground truth. In order to show the robustness and consistent accuracy of the proposed method, the dataset was divided into three parts according to three different ground conditions (sidewalk, grassland and asphalt), and we measured the ATE separately in different ground conditions. The trajectory sequences produced by the LIO systems were firstly compared with the GPS ground truth, and then the sequences of the ATEs could be computed and obtained. Finally, the sequences of ATEs represented as arrays were imported into the box plot function of MATLAB to plot Figure 7. The figure produced by the box plot can display the distribution trend of the sample data, and can better visualize the ATE on the whole dataset and measure the robustness of LIO systems. It can be seen that the accuracy of the UGV’s states estimated by our method is similar in different ground conditions. Most of the ATEs fall into the interval from 1.5 cm to 2.5 cm, which means that the robustness under different ground conditions is guaranteed, whereas the other LIO systems cannot maintain the consistent accuracy of state estimation in changing ground conditions, and the ATEs in different ground conditions differ considerably for the comparative methods. The overall accuracy in the self-gathered dataset is shown in Table 2. Benefiting from the ground perception and optimization algorithm, our method shows a better overall accuracy than LINS and FAST-LIO2, which are also based on iterated ESKF. A similar overall accuracy with a little advantage can be achieved by our method compared with LIO-SAM, which is based on geometric feature-matching and factor-graph optimization, and a better accuracy is achieved by our system in
rough ground conditions, such as a sidewalk and grassland. In the areas with flat ground conditions, Lego-LOAM performs the best accuracy as the plane hypothesis is made to improve the system performance on flat ground, but the overall accuracy in changing ground conditions cannot be guaranteed.

Figure 6. Trajectories and Point cloud map aligned with Google Earth satellite map of the park dataset. The dataset starts and ends at the same position which is indicated by the yellow star in the figure. The enlarged drawing is the real picture of the related ground condition. The detailed distributions of different types of ground conditions can be seen on the lower right corner.

Figure 7. Absolute trajectory errors of the trajectories estimated by different LIO systems compared with GPS ground truth in different ground conditions.
Table 2. Accuracy evaluation of the trajectories estimated by different methods and LIO systems in self-gathered dataset.

|                          | Our Method | LINS     | Lego-LOAM | LIO-SAM    | FAST-LIO2   |
|--------------------------|------------|----------|-----------|------------|-------------|
| Estimated length (m)     | 1517.68    | 1518.24  | 1518.33   | 1517.14    | 1517.96     |
| Translational RMSE (%)   | 2.60       | 3.61     | 3.93      | 2.85       | 3.22        |
| End-to-end translational error (m) | 3.21     | 3.94     | 4.73      | 4.28       | 4.01        |
| End-to-end rotational error (m) | 2.19     | 3.87     | 3.32      | 2.27       | 2.56        |

The red text represents the optimal value of the related item.

5. Conclusions

In this paper, based on the ESKF fusion model, the relationship between the flatness of the ground that the UGV is currently in contact with and the accuracy of the state estimation of the odometry was analyzed; then, in order to improve the accuracy and robustness of lidar-inertial odometry used by UGV traveling in various ground conditions, a ground condition perception and optimization algorithm along with the method to be embedded in the ESKF framework are proposed. The ground condition perception was realized by transforming the raw inertial information from IMU into a ground condition vector in real-time, and the optimization was realized by changing the parameters used in the lidar points processing module to adapt to the changing ground conditions. Through the results gained from the experiments in open datasets and self-gathered datasets with different ground conditions, the ground perception and optimization algorithm can enhance the performance of the LIO system by reducing approximately 20% of the translational RMSE, and the better accuracy and robustness of our system compared with several state-of-art LIO systems boosted by different algorithms are shown. The translational RMSE can achieve around 2%, which is enough to guarantee the accurate localization of a UGV in unstructured environments with various ground conditions. Furthermore, the capacity to maintain the consistency of the state estimation in changing ground conditions is also a characteristic of our method, and most of the ATEs of state estimation can stay at approximately 2 cm regardless of the ground conditions changing. However, our method is designed for ground vehicles and cannot be migrated to aerial vehicles, as the aerial vehicles do not contact the ground directly. In future work, visual information may be added in the ground perception and optimization algorithm to evaluate the flatness of the ground quantitatively and to broaden the application scenarios.

Author Contributions: Conceptualization, Z.Z.; methodology: Z.Z., software, Z.Z. and Z.L.; data duration, Z.Z., Z.L., L.L. and Y.Z.; validation, Z.Z., Z.L. and L.L.; writing—original draft preparation, Z.Z.; writing—review and editing, L.L., Y.Z. and J.S.; supervision, Y.Z. and J.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant No. XDA28040500.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

The following abbreviations are used in this manuscript:

- **UGV**: Unmanned Ground Vehicle
- **ESKF**: Error-State Kalman Filter
- **IMU**: Inertial Measurement Unit
- **LIO**: Lidar-Inertial Odometry
- **NDT**: Normal Distributions Transform
- **LINS**: Lidar-Inertial State Estimator
- **Lego-LOAM**: Lightweight and Ground-Optimized Lidar Odometry and Mapping
- **LIO-SAM**: Lidar-Inertial Odometry via Smoothing and Mapping
- **RMSE**: Root Means Square Error
- **ATE**: Absolute Trajectory Error

References

1. Mohamed, S.A.; Haghbayan, M.H.; Westerlund, T.; Heikkonen, J.; Tenhunen, H.; Plosila, J. A survey on odometry for autonomous navigation systems. *IEEE Access* 2019, 7, 97466–97486. [CrossRef]
2. Xu, X.; Zhang, L.; Yang, J.; Cao, C.; Wang, W.; Ran, Y.; Tan, Z.; Luo, M. A Review of Multi-Sensor Fusion SLAM Systems Based on 3D LIDAR. *Remote Sens.* 2022, 14, 2835. [CrossRef]
3. Li, Y.; Ibanez-Guzman, J. Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems. *IEEE Signal Process. Mag.* 2020, 37, 50–61. [CrossRef]
4. 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]
5. 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; SPIE: Bellingham, WA USA, 1992; Volume 1611, pp. 586–606.
6. Madyastha, V.; Ravindra, V.; Mallikarjunan, S.; Goyal, A. Extended Kalman filter vs. error state Kalman filter for aircraft attitude estimation. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, Portland, OR, USA, 8–11 August 2011; p. 6615.
7. Jiao, J.; Ye, H.; Zhu, Y.; Liu, M. Robust odometry and mapping for multi-lidar systems with online extrinsic calibration. *IEEE Trans. Robot.* 2021, 38, 351–371. [CrossRef]
8. Yang, Y.; Huang, G. Observability analysis of aided ins with heterogeneous features of points, lines, and planes. *IEEE Trans. Robot.* 2019, 35, 1399–1418. [CrossRef]
9. Yang, Y.; Geneva, P.; Eckenhoff, K.; Huang, G. Degenerate motion analysis for aided ins with online spatial and temporal sensor calibration. *IEEE Robot. Autom. Lett.* 2019, 4, 2070–2077. [CrossRef]
10. Luo, X.; Genoves, P.; Lee, W.; Liu, Y.; Huang, G. Lic-fusion: Lidar-inertial-camera odometry. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 5848–5854.
11. 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]
12. Mourikis, A.I.; Roumeliotis, S.I. A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation. In Proceedings of the ICRA, Roma, Italy, 10–14 April 2007; Volume 2, p. 6.
13. Lv, J.; Xu, J.; Hu, K.; Liu, Y.; Zuo, X. Targetless calibration of lidar-imu system based on continuous-time batch estimation. 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. 9968–9975.
14. Guanbei, W.; Guirong, Z. LIDAR/IMU calibration based on ego-motion estimation. In Proceedings of the 2004 4th CAA International Conference on Vehicular Control and Intelligence (CVCI), Hangzhou, China, 18–20 December 2004; pp. 109–112.
15. Luo, X.; Yang, Y.; Genoves, P.; Lv, J.; Liu, Y.; Huang, G.; Pollefeys, M. Lic-fusion 2.0: Lidar-inertial-camera odometry with sliding-window plane-feature tracking. 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. 5112–5119.
16. Zhang, J.; Kaess, M.; Singh, S. On degeneracy of optimization-based state estimation problems. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 809–816.
17. 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.
18. Kim, Y.; Kim, A. On the uncertainty propagation: Why uncertainty on lie groups preserves monotonicity? In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 3425–3432.
19. 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]
20. Lin, J.; Zheng, C.; Xu, W.; Zhang, F. R^2 LIVE: A Robust, Real-Time, LiDAR-Inertial-Visual Tightly-Coupled State Estimator and Mapping. IEEE Robot. Autom. Lett. 2021, 6, 7469–7476. [CrossRef]
21. 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]
22. Bai, C.; Xiao, T.; Chen, Y.; Wang, H.; Zhang, F.; Gao, X. Faster-LIO: Lightweight Tightly Coupled Lidar-Inertial Odometry Using Parallel Sparse Incremental Voxels. IEEE Robot. Autom. Lett. 2022, 7, 4861–4868. [CrossRef]
23. Shan, T.; Englert, 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.
24. Dellaert, F.; Kaess, M. Factor graphs for robot perception. Found. Trends Robot. 2017, 6, 1–139. [CrossRef]
25. Shan, T.; Englert, 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.
26. 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]
27. Su, Y.; Wang, T.; Shao, S.; Yao, C.; Wang, Z. GR-LOAM: LiDAR-based sensor fusion SLAM for ground robots on complex terrain. Robot. Auton. Syst. 2021, 140, 103759. [CrossRef]
28. Wei, X.; Lv, J.; Sun, J.; Pu, S. Ground-SLAM: Ground Constrained LiDAR SLAM for Structured Multi-Floor Environments. arXiv 2021, arXiv:2103.03713
29. Seo, D.U.; Lim, H.; Lee, S.; Myung, H. PaGO-LOAM: Robust Ground-Optimized LiDAR Odometry. arXiv 2022, arXiv:2206.00266
30. Magnusson, M. The Three-Dimensional Normal-Distributions Transform: An Efficient Representation for Registration, Surface Analysis, and Loop Detection. Ph.D. Thesis, Örebro Universitet, Örebro, Sweden, 2009.
31. Behley, J.; Stachniss, C. Efficient Surfel-Based SLAM using 3D Laser Range Data in Urban Environments. In Proceedings of the Robotics: Science and Systems, Pittsburgh, PA, USA, 26–30 June 2018; Volume 2018, p. 59.
32. Yokozuka, M.; Koide, K.; Oishi, S.; Banno, A. LiTAMIN: LiDAR-based tracking and mapping by stabilized ICP for geometry approximation with normal distributions. 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. 5143–5150.
33. Yokozuka, M.; Koide, K.; Oishi, S.; Banno, A. LiTAMIN2: Ultra light lidar-based slam using geometric approximation applied with KL-divergence. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi’an, China, 30 May–5 June 2021; pp. 11619–11625.
34. Shan, T.; Englert, B.; Ratti, C.; Rus, D. Lvi-sam: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi’an, China, 30 May–5 June 2021; pp. 5692–5698.