



Article

InfoLa-SLAM: Efficient Lidar-Based Lightweight Simultaneous Localization and Mapping with Information-Based Keyframe Selection and Landmarks Assisted Relocalization

Yuan Lin ¹ , Haiqing Dong ², Wentao Ye ¹, Xue Dong ^{1,*} and Shuogui Xu ³

¹ China-UK Low Carbon College, Shanghai Jiao Tong University, Shanghai 201100, China; yuan.lin@sjtu.edu.cn (Y.L.); yewentao2000@sjtu.edu.cn (W.Y.)

² Xingji Meizu Group, Wuhan 430056, China; haiqing.dong@xjsd.com

³ Changhai Hospital Affiliated to the Second Military Medical University, Road No. 168, Yangpu District, Shanghai 200433, China; shuogui_xu@smmu.edu.cn

* Correspondence: xue.dong@sjtu.edu.cn

Abstract: This work reports an information-based landmarks assisted simultaneous localization and mapping (InfoLa-SLAM) in large-scale scenes using single-line lidar. The solution employed two novel designs. The first design was a keyframe selection method based on Fisher information, which reduced the computational cost of the nonlinear optimization for the back-end of SLAM by selecting a relatively small number of keyframes while ensuring the accuracy of mapping. The Fisher information was acquired from the point cloud registration between the current frame and the previous keyframe. The second design was an efficient global descriptor for place recognition, which was achieved by designing a unique graphical feature ID to effectively match the local map with the global one. The results showed that compared with traditional keyframe selection strategies (e.g., based on time, angle, or distance), the proposed method allowed for a 35.16% reduction in the number of keyframes in a warehouse with an area of about 10,000 m². The relocalization module demonstrated a high probability (96%) of correction even under high levels of measurement noise (0.05 m), while the time consumption for relocalization was below 28 ms. The proposed InfoLa-SLAM was also compared with Cartographer under the same dataset. The results showed that InfoLa-SLAM achieved very similar mapping accuracy to Cartographer but excelled in lightweight performance, achieving a 9.11% reduction in the CPU load and a significant 56.67% decrease in the memory consumption.

Keywords: SLAM; keyframe selection; relocalization; lidar; accurate indoor localization



Citation: Lin, Y.; Dong, H.; Ye, W.; Dong, X.; Xu, S. InfoLa-SLAM: Efficient Lidar-Based Lightweight Simultaneous Localization and Mapping with Information-Based Keyframe Selection and Landmarks Assisted Relocalization. *Remote Sens.* **2023**, *15*, 4627. <https://doi.org/10.3390/rs15184627>

Academic Editor: Eliseo Clementini

Received: 22 August 2023

Revised: 19 September 2023

Accepted: 19 September 2023

Published: 20 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The widespread utilization of mobile robots in factories and public spaces necessitates the implementation of simultaneous localization and mapping (SLAM) algorithms in applications characterized by large areas and substantial numbers of laser frames [1]. Two-dimensional laser SLAM is increasingly favored due to the lower costs of the single-line lidar and reduced computational requirements in comparison with 3D laser SLAM. However, the current mainstream 2D laser SLAM algorithm (represented by Cartographer [2]) is severely challenged in large area applications [3]. The main issues can be summarized as follows:

(a). As the scale of working space expands, the computational cost for back-end optimization increases substantially, bringing more challenges to the traditional keyframe selection method. Hence, a novel keyframe selection method should be designed to deal with a large number of laser frames.

(b). With the increase in running time, the drift of pose will be an inevitable problem, leading to the failure of the pose estimation. Accurate relocalization is therefore necessary

for quickly recovering from pose estimation failures and acquiring accurate global pose estimation with the prior map. Therefore, a robust back-end is needed to correct the robot's pose, reduce the localization covariance, and update the global trajectory.

(c). The dynamic environment in the working space adversely affects the accuracy and robustness of SLAM algorithms, making it crucial to prevent accidents such as collisions. Consequently, there is a need to mitigate the impact of environmental changes on localization and enhance the system robustness.

To overcome the above challenges, this work proposes the following strategies:

(1) To increase the localization accuracy and robustness and reduce map update errors caused by highly dynamic scenes in industrial applications, landmarks are fused into the point cloud map, and then the map is updated based on the match of landmarks.

(2) An efficient keyframe selection method, as is shown in Figure 1a, is proposed according to the Fisher information between the current frame and the previous keyframe. The traditional keyframe selection methods, such as time, angle, or distance threshold, are no longer utilized in the proposed approach.

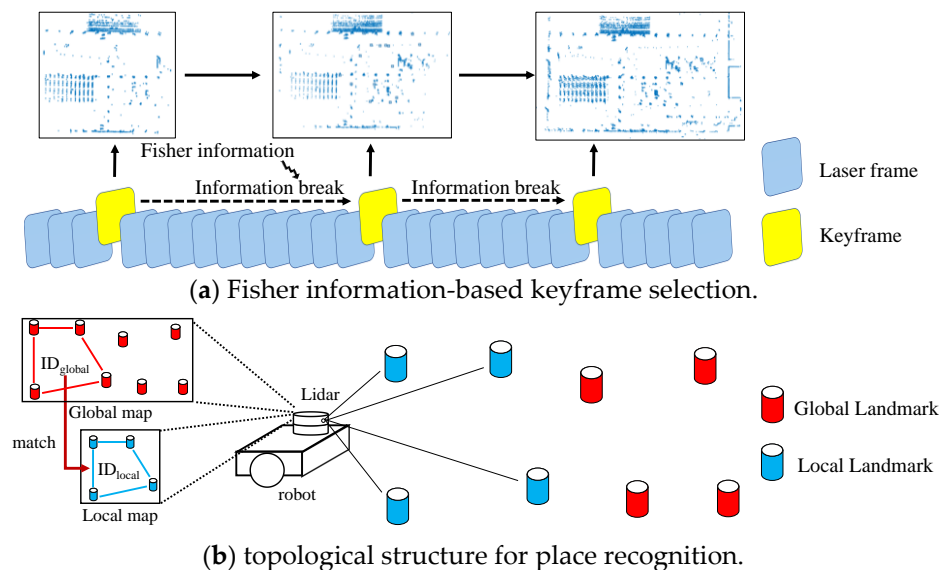


Figure 1. (a) The process of keyframe selection: Initially, the Fisher information is computed between the current frame and the latest keyframe. As the Fisher information gradually drops and the count of low-information frames reaches a predefined threshold (referred to as the “information break”), a new keyframe is chosen. (b) The overview of relocalization: A unique ID is assigned to each local landmark map, which is then used to match with the global landmark map. This facilitates the discovery of closed loops and enhances opportunities for relocalization.

(3) A unique graphical feature ID is designed as a global descriptor for place recognition. This ID is mainly used for matching the local map with the global one to acquire the global pose, as is shown in Figure 1b. The proposed method enables the InfoLa-SLAM to recover the pose rapidly when the robot gets lost.

The rest of the paper is organized as follows. Section 2 provides a comprehensive review of keyframe selection and relocalization methods in SLAM. In Section 3, the structure of the proposed InfoLa-SLAM is introduced, followed by the description of the novel Fisher information-based keyframe selection method and the adaptive relocalization method. Finally, InfoLa-SLAM is validated using laser scan data collected from two scenarios: one with a wide corridor of about 600 m² and the other a warehouse of about 10,000 m². The accuracy and lightweight performance of InfoLa-SLAM were also compared with Cartographer, as is presented in Section 4.

2. Related Work

2.1. Keyframe Selection

Keyframe selection methods allow the system to choose a set of representative frames as keyframes to perform pose estimation and map construction between them. In general, the keyframe selection criteria can be categorized into heuristic-based methods, information-theoretic methods, and learning methods [4].

One of the traditional methods for keyframe selection, based on heuristics, involves determining keyframes by setting a certain threshold to the translation and rotation of the robot, e.g., in the work of Lin et al. [5] and Lio-Sam [6], when the change in the robot's pose exceeds the established threshold, the current laser frame is designated as a keyframe. Similarly, SVO [7] depends on the distance between frames and time threshold to select a keyframe. DSO [8] also selects a keyframe according to the changes of field of view, occlusions, and exposure time of the camera. In addition, some other classical method like ORB-SLAM [9] uses a fixed time interval as a criterion. Another heuristic-based approach to keyframe selection utilizes the number of detected features as the criterion, i.e., when the number or proportion of detected features in the current frame falls within a predetermined threshold, a new keyframe is selected. For example, Qin et al. employ this method to select keyframes in VINS-MONO [10]; a similar approach is also applied by Harmat et al. [11]. Xie [12] further proposes a repetitive motion detection method to reduce redundant keyframes due to the back-and-forth movement of the camera.

The most widely used Cartographer [2] creates a new submap when the mapped location or size of the area has exceeded a certain threshold. Additionally, Cartographer attempts to use distance and time threshold to select new keyframes. However, if Cartographer runs for an extended period of time, the number of submaps to be optimized will increase, leading to a shortage of computational resources and a reduction in the localization efficiency. Our system follows a similar approach, in which a keyframe is selected to represent a portion of the overall map. However, when dealing with large-scale scenes, such approaches can lead to an excessive number of keyframes, resulting in wasted computational resources and reduced system performance.

Another approach to select keyframes is the information-theoretic approach, which is based on more rigorous metrics. For example, Das et al. [13] propose a keyframe selection method for multi-camera SLAM that uses entropy as a criterion. This information-theoretic method uses the entropy of the distribution of feature points from each camera to determine the keyframe. The selected keyframe represents the distribution of feature points in the environment. Similarly, Davide Scaramuzza et al. [4] propose an entropy-based keyframe selection method, which selects keyframes based on the uncertainty of the pose of the keyframe bundle with respect to the current map. The method employs the concept of differential entropy to estimate the uncertainty of the current pose and uses this information to determine whether a new keyframe should be inserted. M-LOAM by Liu [14] uses the information matrix to select a keyframe, i.e., when a new frame is acquired, M-LOAM first extracts the feature points and edge points from the lidar data like LOAM [15] does, then it aligns the feature points and edge points with the previous keyframe to estimate motion. The information matrix is utilized to assess the quality of the pose estimation. If the information matrix of the current frame falls below a predetermined threshold, the current frame is selected as a keyframe. Following their work, the current InfoLa-SLAM also adopts the information matrix for keyframe selection; however, our approach distinguishes itself by considering not only the information derived from the current frame but also the information acquired from all frames between the current frame and the previous keyframe. By incorporating this additional information, we make more comprehensive and informed decisions when selecting keyframes, leading to improved performance in various scenarios.

Furthermore, there are also research on how to use the learning method to select keyframes. For example, Piao [16] proposes a keyframe selection method that utilizes the IMU preintegration between two image frames to estimate the current pose, and the

model weight is automatically updated by continuously learning whether the value of the preintegrated IMU data and the result of the input frame is keyframe or not.

2.2. Relocalization

In addition to keyframe selection, relocalization (i.e., place recognition) is another one of the most important problems in SLAM, which significantly affects accuracy and stability. This module is necessary when a robot enters a previously mapped environment, as well as when the robot gets lost in the course of moving. Current studies on relocalization can be categorized into image-based [17–21] and lidar-based methods [22–28].

For place recognition, the descriptors are usually used to represent the geometric profile of the point clouds or features in an image. BRIEF [21] is a feature descriptor that is commonly used in visual SLAM, which generates binary codes by comparing pairs of pixels to describe local features of the image. Similarly, 3D-SIFT [29] is another type of descriptor for geometric features. The descriptor is constructed using a set of histograms of oriented gradients (HOG), which are calculated by using a set of gradient orientations and magnitudes computed from the points in the neighborhood. In addition, 3D-Harris [30] is a feature detection algorithm and is used to identify distinctive points in an image. It is based on the 2D Harris corner detection algorithm, which is used to identify corners in the point cloud. The algorithm calculates a measure of the cornerness of each point based on the curvature of the surface in the neighborhood and selects points with high cornerness as interest points.

In addition, various descriptors can be utilized in lidar-based SLAM. Shutong Jin [22] proposes a lidar-based place recognition descriptor called SectionKey, which incorporates both semantic and geometric information to address the challenge of place recognition. The concept of Sectionkey is composed of three distinct components, i.e., the number of semantic clusters, the number of transitions, and the relative topological distance. These components work together to provide a comprehensive representation of the Sectionkey, which is then utilized to select a new keyframe. Similarly, Scan Context [27] is a non-histogram-based global descriptor for 3D lidar scans. It starts by extracting semantic information from the segmented image or point cloud to identify static main objects based on their average coordinates. Rings are drawn around each main object, further divided into sectors to form a grid. The 3D point cloud is projected onto a plane, and the average height within each grid cell generates a matrix representation. The RingKey is derived from row averages, while the SectorKey is obtained from column averages. Utilizing the RingKey provides rotational invariance for extracting candidate frames. Subsequently, the SectorKey calculates similarity scores between objects in the candidate frames and the current frame, based on their deviation (yaw angle), to select a new keyframe.

Additionally, recent advancements in computer vision have led to the development of feature descriptors based on deep learning techniques. These descriptors are generated with neural network architectures to learn how to extract features from large datasets of images or point clouds. One example of a deep learning-based descriptor is the SIFT-CNN [31], which is developed from SIFT. Another example is the PointNet [32] descriptor, which is a deep learning-based descriptor for 3D point clouds. PointNet has also been used in tasks such as object recognition and scene segmentation.

3. Methodology

3.1. System Overview

As is shown in Figure 2, InfoLa-SLAM processes real-time lidar point cloud data through a structured architecture. This mechanism initially identifies a keyframe in order to construct a submap from the input. Subsequently, areas of high echo intensities, arising from reflectors, are detected and conserved as landmarks within the submap. The system then executes point cloud registration to estimate the relative pose, serving as the prediction of an extended Kalman filter (EKF). Correspondingly, landmarks in each frame are identified and matched with those of the preceding keyframe, thereby updating the EKF. This is

followed by the execution of loop closure detection and global optimization, which leverage the poses of the keyframes. In circumstances involving tracking loss or when prior map knowledge is available, relocalization is performed. The system receives real-time lidar point cloud and outputs real-time robot pose and map, enabling accurate localization and mapping in dynamic environments. The novel approach to keyframe selection and relocalization will be elaborated in the subsequent parts.

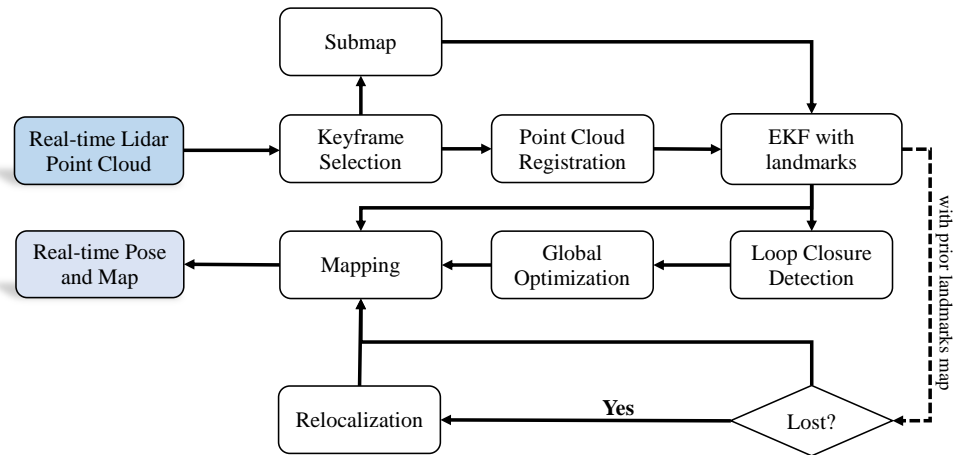


Figure 2. The structure of the lidar-based InfoLa-SLAM.

3.2. Keyframe Selection Strategy

In this section, the novel keyframe selection strategy proposed in this work is introduced. To begin with, it is necessary to provide a clear and concise explanation of the related statistical background as follows.

Assume that X is an independent, identically distributed and discrete random variable. X_1, \dots, X_n are a set of samples from X , and x_1, \dots, x_n are the observed values. They obey a probability distribution $f(x; \theta)$, where θ is a parameter vector. The way to estimate θ is to maximize the likelihood $L(x; \theta)$ with respect to θ , i.e., a maximum likelihood estimation (MLE). The likelihood function is constructed as follows:

$$L(\theta; \mathbf{x}) = \prod_{i=1}^n f(x_i; \theta). \quad (1)$$

Take the logarithm of the likelihood function to acquire the log-likelihood function. Next, take the derivative of the log-likelihood function with respect to θ , which is defined as the score function as follows:

$$s(\theta) = \nabla \log L(\theta; \mathbf{x}). \quad (2)$$

According to [33], Fisher information is defined as the covariance of the score function. If θ is represented as a vector, the Fisher information takes the form of a matrix as follows:

$$\begin{aligned} \mathbf{I}(\theta) &= E \left[(s - E(s))(s - E(s))^T \right] \\ &= E \left[\nabla \log L(\theta; \mathbf{x}) \nabla \log L(\theta; \mathbf{x})^T \right]. \end{aligned} \quad (3)$$

This matrix represents the reliability of the estimation of θ . The expected value of the score function, $E(s)$, can be shown to be zero under certain regularity conditions [34]; see details in Appendix A Equation (A2). Additionally, it can be rigorously established that the Fisher information matrix, I , is equal to the negative expectation of the Hessian matrix for the log-likelihood; see details in Appendix A Equation (A5).

$$I = -E\left[\nabla^2 \log L(\theta; x)\right] = -E\left[H_{\log L(\theta; x)}\right]. \quad (4)$$

The second derivative of the log-likelihood function, i.e., the Fisher information matrix, is always multiplied by -1 to ensure that the Fisher information is a positive quantity because the Fisher information matrix provides the curvature of the log-likelihood function near its maximum. In addition, the Hessian matrix provides a local approximation of the Fisher information at the MLE. Thus, in MLE, the information matrix is often treated as the Hessian matrix as follows:

$$I(\theta) \approx H(\theta). \quad (5)$$

Additionally, it has been proved that based on the assumptions of asymptotic normality, consistency, regularity conditions, and independence of the data, the covariance matrix of the estimator is approximately equal to the inverse of the Fisher information matrix according to [33,35].

$$\Sigma_{\theta} = I(\theta)^{-1} \approx H(\theta)^{-1}. \quad (6)$$

For the keyframe selection strategy, the registration between two laser frames is modeled as an independent and identically distributed discrete random variable, Y . This variable is assumed to follow a specified probability distribution

$$Y \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad (7)$$

where $\boldsymbol{\mu} = (\delta x, \delta y, \delta \theta)$ is a mean value vector of pose estimation, and $\boldsymbol{\Sigma}$ is the covariance matrix. The calculation of the lidar pose can be formulated as an MLE. In order to estimate the parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, we formulate a nonlinear optimization problem and iteratively converge the formula to the target values through linearization. In this work, the nonlinear optimization problem is formulated with a residual function $f(\boldsymbol{\mu})$ according to the distance between corresponding points d_i in the registration.

$$f(\boldsymbol{\mu}) = d_i, \quad (8)$$

where d_i represents the distance of each corresponding point pair. The optimization problem can be described as

$$\boldsymbol{\mu}^* = \arg \min F(\boldsymbol{\mu}) = \arg \min \sum_{i=1}^n f(\boldsymbol{\mu}). \quad (9)$$

In this study, the Gauss–Newton method is employed to solve this nonlinear optimization problem, and through this method, the Hessian matrix with respect to $\boldsymbol{\mu}$ can be obtained. Meanwhile, as was discussed before, the Hessian matrix can be considered as the Fisher information matrix. Thus, the Fisher information matrix contains the result of this optimization, which is useful in keyframe selection because it takes the correlation of consecutive scans into account. Therefore, Fisher information can be a powerful tool in keyframe selection.

To measure the Fisher information from the Fisher information matrix, we consider it as a linear transformation of the standard normal distribution. The Fisher information matrix is a real symmetric matrix, with eigenvalues reflecting the scale and eigenvectors reflecting the rotation. We define the *information value (IV)* as the sum of the Fisher information of the sample, using the eigenvalue vector λ of the Fisher information matrix.

$$IV = \text{tr}(I) = \sum_{i=1}^n \lambda_i, \quad (10)$$

where I is the Fisher information matrix, and the *average information value (AIV)* is the average Fisher information of the sample

$$AIV = \frac{IV}{n}, \quad (11)$$

where n is the number of the corresponding point pairs.

The average information value is a measure to quantify the Fisher information in the registration between the current laser frame and the previous keyframe. A larger average information value indicates a smaller registration error. On the other hand, if the value is low, or even close to zero, it means that the error of the alignment is large. In our keyframe selection strategy, the following definitions are provided, as shown in Figure 3.

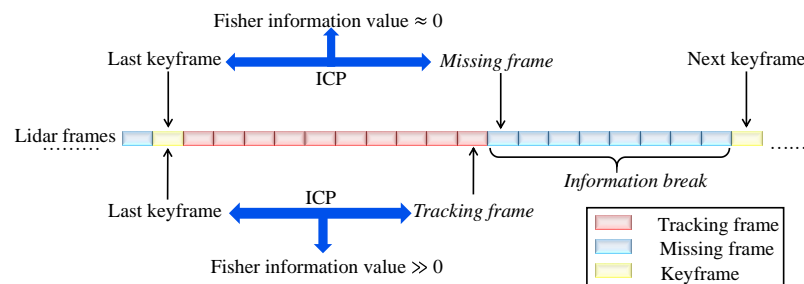


Figure 3. An illustration of keyframes (yellow), tracking frames (red), missing frames (blue), and information breaks. The classification of the frames is based on the Fisher information value obtained through the registration (ICP). The selection of the keyframes, represented by the yellow boxes, is contingent upon the occurrence of information breaks, caused by a significant number of missing frames.

Tracking frame: The average information value of the registration between the current frame and the previous keyframe is significantly greater than zero. The tracking frame indicates that there is a strong correspondence between the current frame and the previous keyframe.

Missing frame: The average information value is close to zero. The missing frame indicates that there is little correspondence between the current frame and the previous keyframe, resulting in little Fisher information being obtained from the registration.

Information break: When the number of missing frames exceeds a predetermined threshold, this is defined as an information break. This situation indicates that the Fisher information between the current frame and the previous keyframe cannot be acquired anymore.

At last, as demonstrated in Figure 3, our strategy checks whether there is an information break. If the information break occurs, the current laser frame is selected as a keyframe. This method has been demonstrated to be both efficient and lightweight.

3.3. Adaptive Relocalization

As is shown in Figure 4, place recognition is necessary when the robot initializes in an unknown environment or gets lost while moving. Inspired by SectionKey and Scan Context for feature description, a unique graphical feature identifier (ID) is designed for each landmark detected in both the local and global maps, as illustrated in Figure 5. This identifier serves as the basis for the match between two maps. The graphical feature ID is constructed according to specific rules, which will be described in detail as follows.

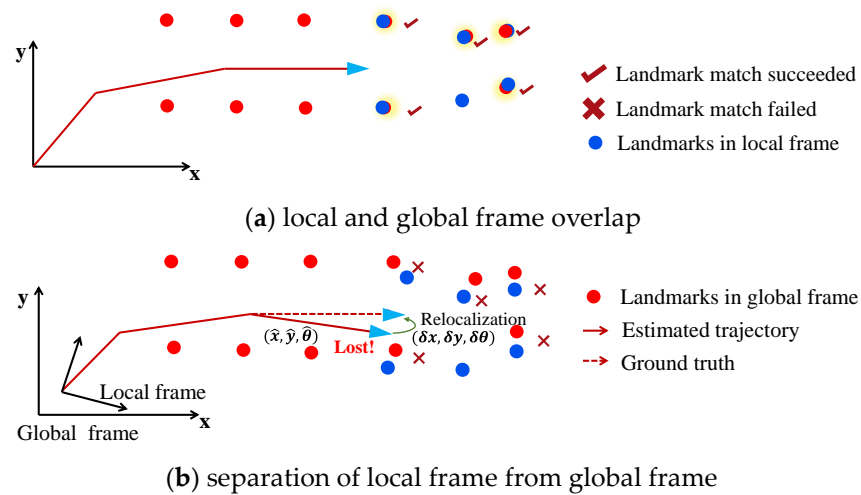


Figure 4. Illustration of matching of the landmarks between local map and global map. (a) Local and global frame overlap. (b) The local frame deviates from the global frame because of some factors like noise $(\delta x, \delta y, \delta \theta)$, and a failure in EKF update occurs, which requires the relocalization module to calculate the transformation between the two frames to acquire the global pose.

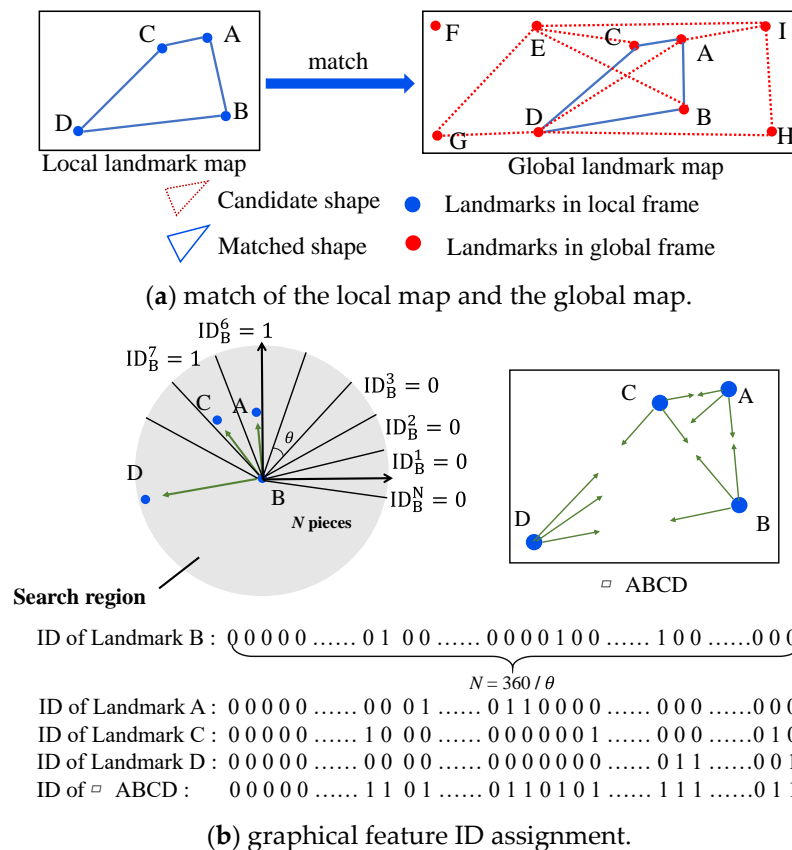


Figure 5. Local landmark map is matched with the global one, as shown in (a). Each landmark on both the local and global maps is assigned a unique graphical feature ID, shown in (b), which is constructed by considering the presence or absence of direction vectors within a set of predefined intervals surrounding the landmark. A, B, C, D represent the detected landmarks.

(1) Given a set of landmarks z_1, \dots, z_n in the local map, we calculate all the direction vectors of each pair of landmarks. This is performed within the 2D local coordinate system.

$$\overrightarrow{z_i z_j} = (x_i - x_j, y_i - y_j)^T, i \neq j; i, j = 1, \dots, n. \tag{12}$$

Then the angle of direction vector can be calculated as

$$\alpha_j = \arctan \frac{y_i - y_j}{x_i - x_j}. \tag{13}$$

(2) In polar coordinates we divide the surrounding area of the landmark z_i into $N = \frac{360}{\theta}$ pieces, as is shown in Figure 5, where θ is the angle resolution. Assume $S_{z_i}^k$ means the k -th piece of landmark z_i , and each piece is represented by

$$S_{z_i}^k = (\theta \cdot (k - 1), \theta \cdot k], k = 1, \dots, N, \tag{14}$$

where k is the index of the piece.

(3) Let ID_{z_i} represent the complete graphical feature ID of landmark z_i . Assume each interval can only contain at most one direction vector. If the angle of direction vector α_j falls within the interval $S_{z_i}^k$, $ID_{z_i}^k$ is then marked as 1; otherwise, it is set to 0. By recording the presence or absence of the direction vector in each interval, the graphical feature ID of the landmark z_i is obtained, which is only composed of 0 and 1.

$$ID_{z_i}^k = \begin{cases} 1, & \alpha_j \in S_{z_i}^k, \\ 0, & else. \end{cases} \tag{15}$$

For example, if the direction vector $\overrightarrow{z_1 z_3}$ falls within the interval $S_{z_1}^4$, $ID_{z_1}^4$ is then marked as 1. In this way the graphical feature ID of landmark z_i is as follows:

$$ID_{z_i} = \underbrace{00010001\dots\dots00010}_N. \tag{16}$$

(4) The graphical feature ID of each landmark ID_{z_i} can be calculated with (15). An OR operation is taken to form the complete graphical feature ID of the graph ID_z , where z represents a graph composed of the landmarks. This complete graphical feature ID_z is a representation of the landmarks graph on the map.

$$ID_z = ID_{z_1} \vee ID_{z_2} \vee \dots \vee ID_{z_n}. \tag{17}$$

The complete graphical feature ID of a map ID_z can be calculated by four steps:

- (1) Determine the direction vectors for current landmark with respect to the surrounding landmarks;
- (2) Divide the surrounding area into a number of intervals following (14) at the center of the current landmark;
- (3) Mark the corresponding interval according to the direction vector with 0 or 1 to acquire the graphical feature ID of one single landmark;
- (4) Combine the graphical feature ID of each landmark as (17) to form the complete graphical feature ID of the graph.

However, it is not practical to match with all landmarks on the global map, because a large number of possible combinations would exhaust the computational resources. To address this, an adaptive design with two filters is proposed. In the global map, the selection of landmarks is limited to a specific search region. And the number of selected landmarks decreases as the distance from the center landmark increases because the detected landmarks on the local map must be within the range of lidar. To filter out the mismatch in advance, two filters are proposed:

(1) The first filter calculates the area of the graph made up of the landmarks in the local map, ensuring that the corresponding area of the graph in the global map does not deviate significantly;

(2) The second filter calculates the distance from the centroid of the graph in the local map to each landmark point, ensuring that the corresponding distances in the global map do not deviate significantly.

These improvements accelerate the match by reducing the number of candidates that need to be considered. In addition, this approach is highly robust to noise because of the design of (14). The following experiments have shown that it can tolerate noise from sensors or errors in the pose estimation.

Once a match is found, an optimization process is used to solve the rough transformation between the local and the global frame. This rough transformation is then used as the initial guess of the iterative closest point (ICP) registration between the global and local point clouds to refine the current pose estimation.

4. Experiments

A series of experiments were performed to qualitatively and quantitatively analyze the proposed methods and system. For this work, laser scan data were collected from two scenarios using a low-cost and short-range single-line lidar: one with a wide corridor of about 600 m² and the other a warehouse of about 10,000 m². The lidar was horizontally mounted on the top of the robot to obtain a wide field of view. The landmarks were cylinders with highly reflective surfaces, with a diameter of 10 cm and a height of 30 cm. To mitigate potential topological ambiguities, landmarks should be prearranged irregularly. This is because our proposed relocalization method was based on the topological structure of landmarks. A regular arrangement could reduce the distinctiveness between landmarks, thus affecting the accuracy of the localization. In addition, if landmarks are viewed from different directions, the method remains effective, but the performance may be different according to the shape of landmarks. The cylindrical shape for landmarks is recommended to ensure consistent visibility of their centers from multiple viewpoints. The implementation was carried out in C++ programming and executed on a laptop with an Intel i7-11800H CPU, running Ubuntu Linux. No parallel computing was utilized, and computation was solely performed using the CPU.

4.1. Keyframe Selection

To validate the effectiveness of the proposed keyframe selection method, Figure 6 illustrates the variations in the average Fisher information between the current frame and the previous keyframe. Notably, this figure showcases two information breaks, which trigger the selection of new keyframes. When the average Fisher information approaches zero, it indicates a missing frame. Once the number of missing frames exceeds a predefined threshold, an information break is detected, leading to the selection of a new keyframe. In the experiments conducted in this paper, the optimal threshold for triggering an information break was set at a fixed value of 30, meaning an information break was detected when the count of missing frames accumulated to 30. This threshold can be adjusted based on specific environments; for instance, in compact scenarios, a higher threshold can be set. Furthermore, the figure also illustrates that as the pose of the current frame deviated away from the previous keyframe, the average Fisher information of the registration progressively approached zero, indicating a decreasing correlation between the two frames.

A comparative study of the proposed keyframe selection method and traditional keyframe selection method was performed in a 600 m² corridor that contained 2750 laser scans without loop closure. The traditional keyframe selection method studied here was based on the distance, angle, or time threshold. As is shown in Figure 7, under similar localization and mapping performances, the proposed method selected 12 keyframes, while the traditional method chose 23 keyframes. Therefore, the proposed method reduced the number of redundant keyframes by 47.83%. Figure 7b demonstrates the ground truth

map of the corridor, where the landmarks and some obstacles are marked. Compared with the ground truth, the RMSE values of the maps with the proposed method and the traditional method were 0.31 and 0.51, respectively, as is shown in Table 1. During the robot’s movement, vibrations were inevitable. Since the single-line lidar was rigidly connected to the robot and scanned only a single plane, there would inevitably be some abnormal frames, such as those caused by unstable lidar leading to tilted and discontinuous scans. In traditional methods that use fixed motion changes or time intervals to select keyframes, many keyframes are often extracted because it is unknown whether the current frame is important. This can result in abnormal frames being chosen as keyframes, and all subsequent frames will be registered to this faulty keyframe, leading to the accumulation of errors. This is why the accuracy appears worse in traditional methods. The proposed method evaluated the correspondence between the current frame and the last keyframe and did not easily select a new keyframe unless there was an information break. This avoided the error accumulation caused by choosing many abnormal frames as keyframes, leading to more accurate pose estimations.

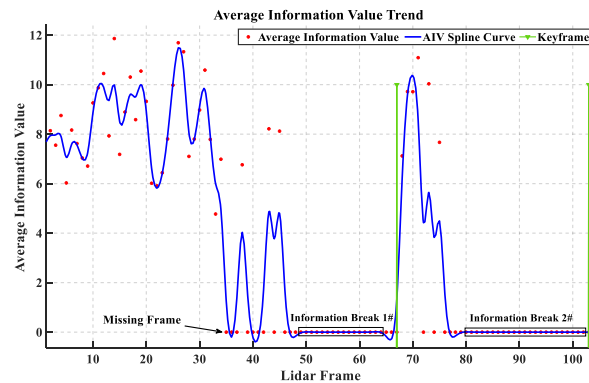
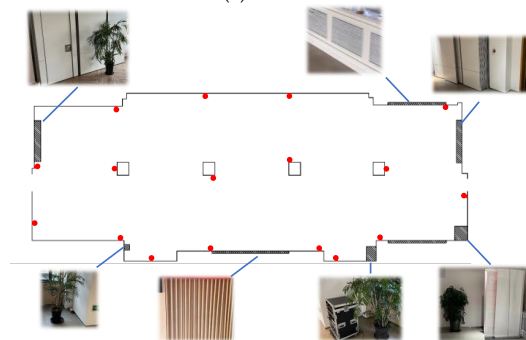


Figure 6. Curve of average Fisher information values acquired by registration of the current frame with the previous keyframe. The curve demonstrates a decrease in average Fisher information value as the distance between the current frame and previous keyframe increases, until it approaches zero. A large number of missing frames, indicated by low average Fisher information values, result in information breaks and prompt the selection of a new keyframe.



(a) corridor.



(b) ground truth (landmarks marked as red points).

Figure 7. Cont.

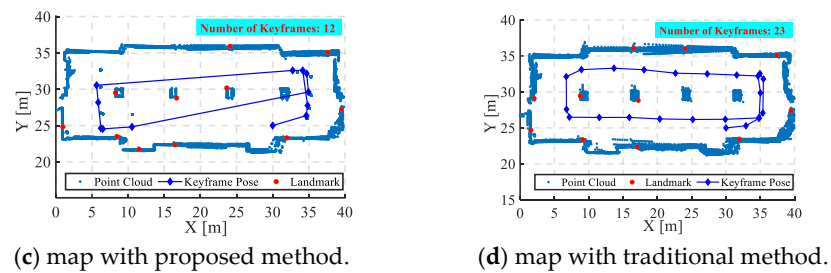


Figure 7. (a) An about 600 m² corridor where the laser frames were collected, with the landmarks shown in the yellow dashed square. (b) The ground truth of the map. The red dots represent the landmarks. The shaded areas represent some obstacles, which are marked with pictures. (c,d) A comparison of the map with the proposed Fisher information-based keyframe selection method and the traditional method (based on the distance, angle, and time threshold). The number of selected keyframes in (c) is much less than that in (d).

Table 1. Relative error of maps reconstructed with the proposed and traditional keyframe selection methods.

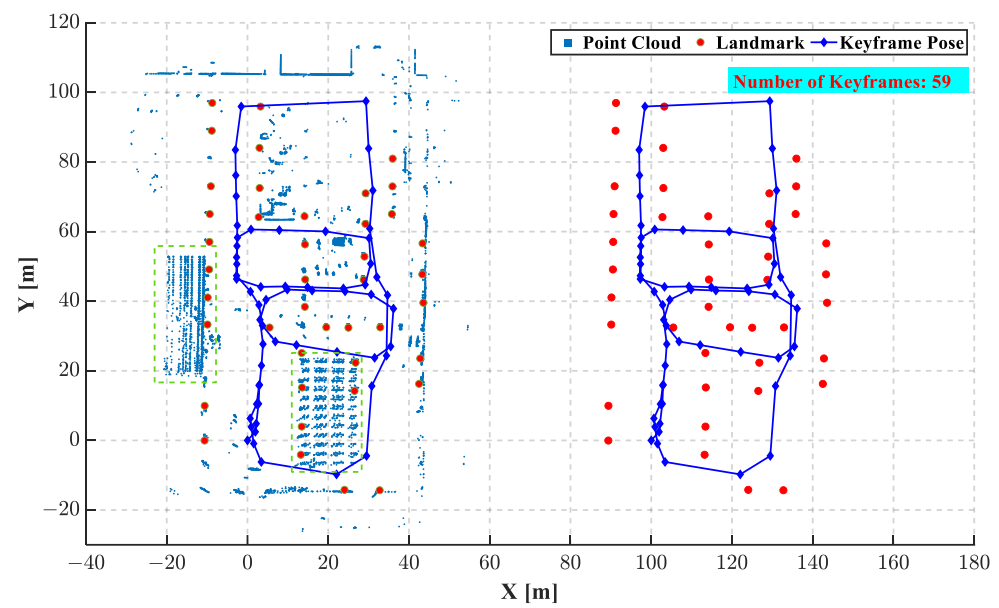
Method	Area Error (Relative)	Point Error (Relative)	RMSE
InfoLa-SLAM	6.44%	1.97%	0.31
Traditional method	9.97%	6.34%	0.51

To further evaluate the performance of the keyframe selection strategy in large areas, we tested its performance in a 10,000 m² warehouse, as is shown in Figure 8a. Since ground truth was not available in the warehouse, only a qualitative comparison was made between the two reconstructed maps. As is shown in Figure 8b,c, under very similar mapping performances, the proposed approach generated a map using 59 selected keyframes, while the traditional method utilized 91 keyframes, achieving a significant reduction of 35.16% in the number of keyframes. These findings underscored the effectiveness of the proposed approach in minimizing keyframe redundancy. Thus, by achieving the same mapping result with fewer keyframes, the proposed method can reduce the computational cost for SLAM.

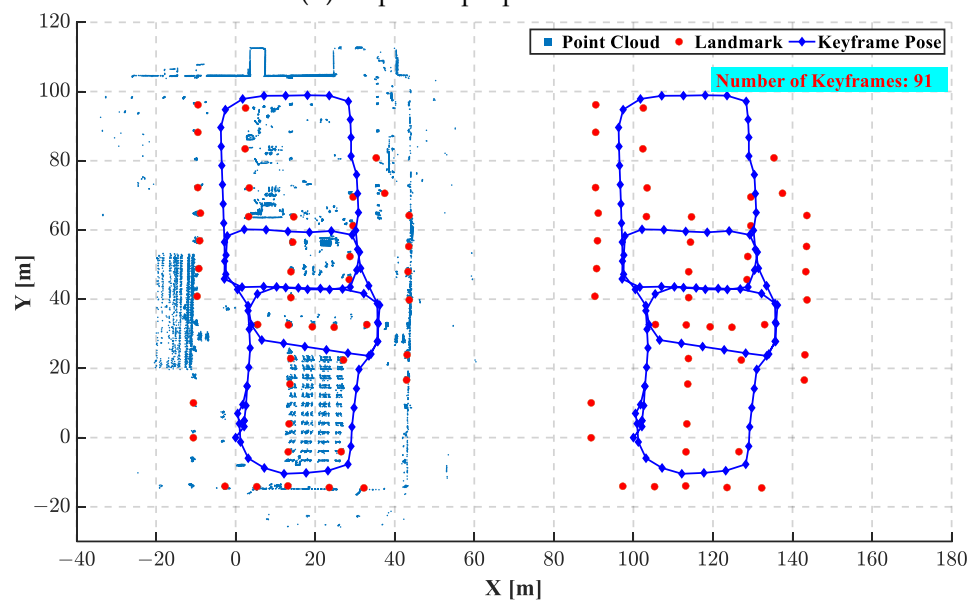


(a) logistic warehouse

Figure 8. Cont.



(b) map with proposed method



(c) map with traditional method

Figure 8. A 10,000 m² logistics warehouse where the dataset was collected (a–c) is a comparison of the proposed keyframe selection method and a traditional method. Landmarks fused with the point cloud and trajectory map are shown on the left, and those without the point cloud are presented on the right. The green dashed rectangle in (b) represents the shelves of warehouse.

4.2. Relocalization

To evaluate the performance of the relocalization module, the warehouse dataset was used. In order to simulate the pose estimation failures, noise was deliberately introduced to three positions, depicted by the dashed blue circles in Figure 9a,b.

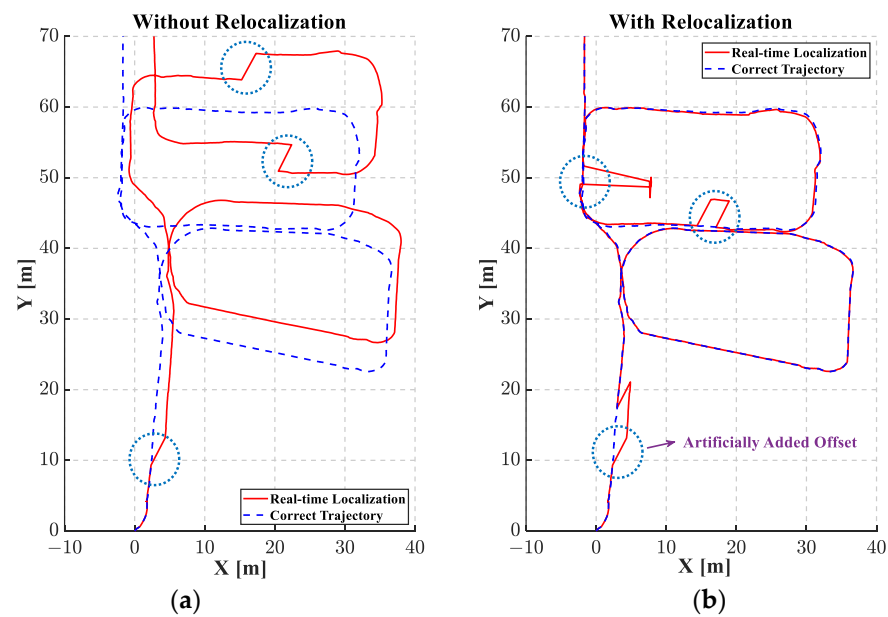


Figure 9. Result of the proposed relocalization method. Without relocalization (a) and with relocalization (b), the red line represents the real-time localization, the dashed blue line represents the correct trajectory, and the dashed blue circles represent three artificial added offsets.

It can be seen in Figure 9a that online localization drifted dramatically without relocalization. Conversely, the proposed relocalization was able to acquire an accurate global pose in a very short period, resulting in a track that mostly aligned with the correct trajectory (Figure 9b). It demonstrated the effectiveness and accuracy of the relocalization module in estimating the global pose, even when subjected to artificial perturbations.

To evaluate the sensitivity of the proposed relocalization module to the measurement noise from sensor, we conducted experiments to assess its successful match probability and time consumption under different levels of measurement noise, as is shown in Figure 10. When the level of measurement noise was lower than 0.05 m, nearly all of the local map could be matched with the global one with a success rate of 96%. The time consumption for the relocalization module was within 28 ms. This indicated that the proposed relocalization method was highly reliable and efficient even under high levels of measurement noise.

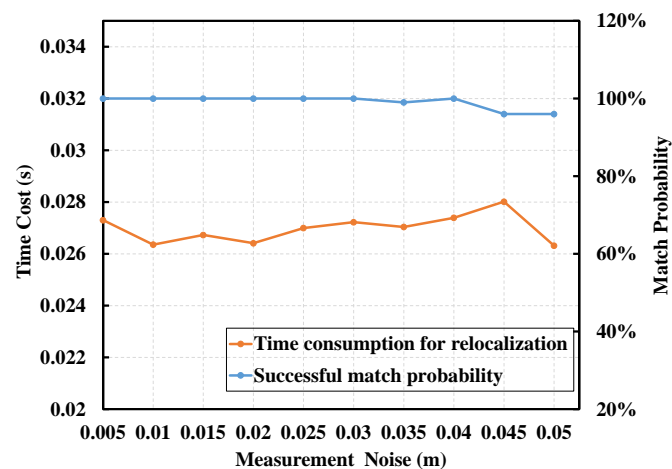


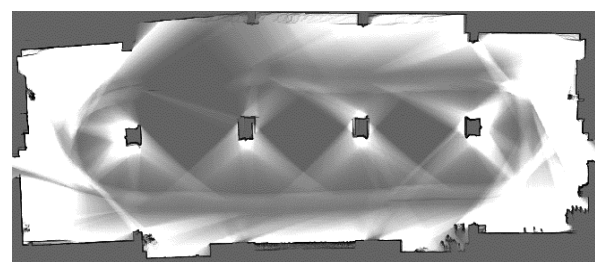
Figure 10. The impact of measurement noise on the relocalization module. When the measurement noise is less than 0.05 m, almost all local map is correctly matched with the global one. The time consumption is within 28 ms.

4.3. Lightweight Performance

In order to assess the lightweight performance of InfoLa-SLAM, extensive testing was conducted using the corridor dataset, and a comparative analysis was performed against the widely used Cartographer. In Cartographer, the default configuration was employed, with the “use_landmarks” and “use_pose_extrapolator” options set to true. Both systems were subjected to the same dataset and demonstrated similar mapping results. When compared with the ground truth, as is shown in Table 2, the proposed system exhibited a modest area error of 2.9% relative to the ground truth, whereas the Cartographer presented an area error of 5.5%. Furthermore, the root mean square error (RMSE) of the proposed system was 0.39, as opposed to 0.49 for Cartographer. The mapping of both systems is shown in Figure 11.

Table 2. Comparison of relative error of maps reconstructed by the proposed InfoLa-SLAM and Cartographer.

Metric	InfoLa-SLAM	Cartographer
Area error (relative to GT)	2.9%	5.5%
RMSE	0.39	0.49



(a) mapping of Cartographer.



(b) mapping of the proposed system.

Figure 11. Mapping of Cartographer and the proposed system.

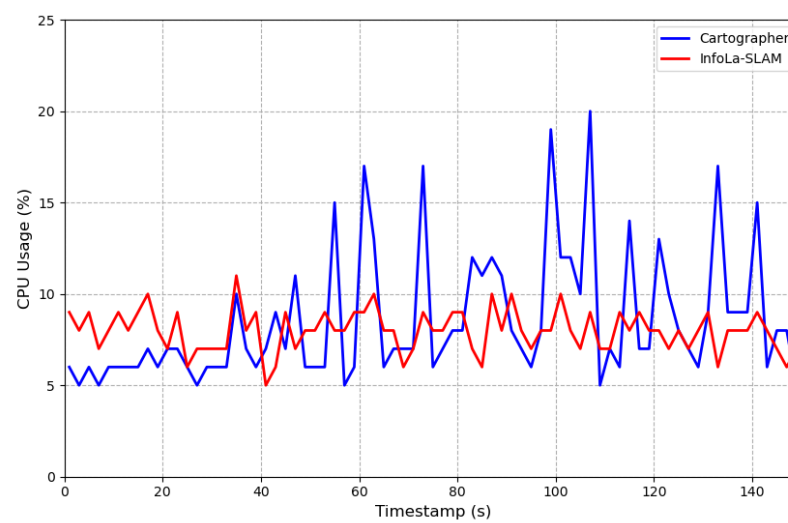
InfoLa-SLAM, which incorporates an extended Kalman filter (EKF) process supplemented with easily detectable landmarks, guaranteed efficient and precise pose estimation. Additionally, the novel approach to keyframe selection effectively reduced redundancy, alleviating the computational burden of global optimization predicated on keyframes. Coupled with the innovative global descriptor, efficient and precise loop closure detection was achieved.

As a collective result of these improvements, InfoLa-SLAM outperformed Cartographer in both CPU load and memory usage, as detailed in Table 3. InfoLa-SLAM showcased a lower average CPU load of just 7.98%, in comparison with the Cartographer’s 8.52%. Moreover, the reduced variance and peak CPU loads of our system indicate its stability and efficient resource allocation, suggesting its superiority in handling real-time computational tasks. Similarly, InfoLa-SLAM exhibited superior memory efficiency with an average memory usage of 0.55%, while Cartographer’s memory usage averaged at 1.23%. The variance and peak memory usage were also notably reduced in InfoLa-SLAM, showcasing its ability to optimize memory allocation. The CPU load and memory usage during the

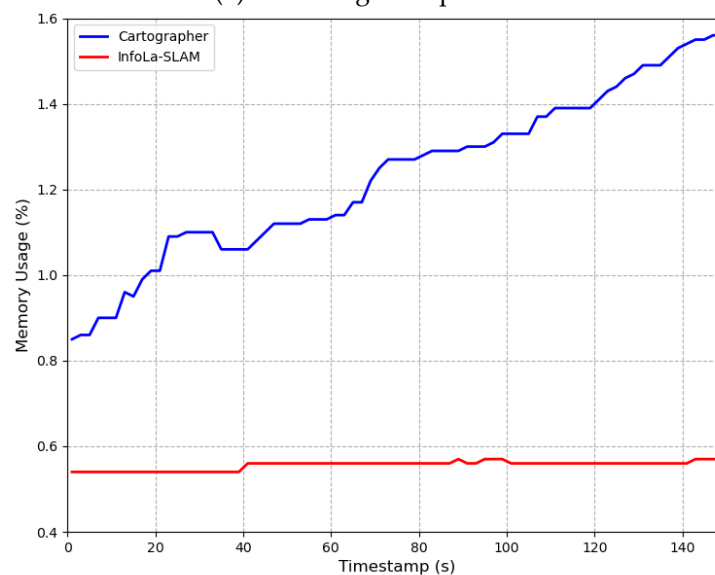
runtime of both systems are shown in Figure 12. Based on these results, it can be seen that InfoLa-SLAM's lightweight character ensures efficient utilization of resources, making it well-suited for applications in resource-constrained environments or devices with low processing capabilities.

Table 3. Comparison of CPU Load and Memory Usage: InfoLa-SLAM vs. Cartographer.

Metric	InfoLa-SLAM	Cartographer
CPU load (mean)	7.98%	8.52%
CPU load (variance)	1.29	12.5
CPU load (peak)	11.0%	20.0%
Memory usage (mean)	0.55%	1.23%
Memory usage (variance)	0.0001	0.037
Memory usage (peak)	0.57%	1.56%



(a) CPU usage comparison.



(b) Memory usage comparison.

Figure 12. Comparison of CPU load and memory usage between InfoLa-SLAM and Cartographer.

5. Conclusions

This work reported an information-based landmarks assisted SLAM, named InfoLa-SLAM, with a novel keyframe selection method based on Fisher information and an efficient adaptive relocalization method based on the unique graphical feature ID of the landmarks map. The experimental results showed that InfoLa-SLAM with these two novel designs exhibited enhanced stability and lightweight performance.

The keyframe selection method, leveraging Fisher information in point cloud registration, enabled the precise and comprehensive assessment of correspondence between the current frame and the previous keyframe. The experimental results demonstrated its effectiveness, achieving a 35.16% reduction in redundant keyframes in a warehouse of 10,000 m² and a 47.83% reduction in a corridor of 600 m².

The adaptive relocalization method introduced a novel global descriptor known as the graphical feature ID, facilitating precise and rapid matching between the local and global maps. The approach demonstrated the capability to swiftly estimate the global pose when the prior landmarks map was available. The sensitivity analysis under measurement noise revealed a high success rate of 96% in matching local and global maps, particularly when the measurement noise was below 0.05 m. Furthermore, the relocalization module showcased impressive computational efficiency, with execution times of less than 28 ms.

Furthermore, InfoLa-SLAM outperformed Cartographer in terms of lightweight performance, with a 9.11% reduction in the CPU load and a significant 56.67% decrease in memory consumption under a very similar mapping accuracy. These results highlight the practical significance and potential of InfoLa-SLAM for real-world applications in resource-constrained environments.

Author Contributions: Conceptualization, X.D., Y.L. and S.X.; methodology, Y.L.; software, Y.L., W.Y. and H.D.; validation, W.Y.; writing—original draft, Y.L.; writing—review and editing, X.D., H.D. and S.X.; supervision, X.D.; project administration, X.D.; funding acquisition, X.D. and S.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was financially supported by the National Natural Science Foundation of China under Grant No. 52006137, National Natural Science Foundation of China under Grant No. 52376160, as well as the Shanghai Sailing Program, under Grant No. 19YF1423400.

Data Availability Statement: Not applicable.

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

Appendix A

1. Proof of $E(s) = 0$

The following proof shows that the first-order derivative of the log-likelihood, i.e., the score function, has an expectation of 0.

First the score function is

$$s(\theta) = \nabla \log L(\theta; X) \quad (\text{A1})$$

The expectation of score function with respect to the $L(\theta; x)$ is

$$\begin{aligned} E_{L(\theta;x)}[s(\theta)] &= E_{L(\theta;x)}[\nabla \log s(\theta)] = \int \nabla \log L(\theta; x) L(\theta; x) dx \\ &= \int \frac{\nabla L(\theta; x)}{L(\theta; x)} L(\theta; x) dx = \int \nabla L(\theta; x) dx = \nabla \int L(\theta; x) dx = 0 \end{aligned} \quad (\text{A2})$$

Above the gradient is with respect to the θ .

2. Proof of $I = -E[H_{\log L(\theta;x)}] = -E[\nabla^2 L(\theta; x)]$

The following proof shows that the information Matrix I is equal to the negative values of the expectation of the Hessian matrix for the log-likelihood.

The Hessian matrix of logarithmic likelihood is given by the Jacobian matrix of its gradient

$$\begin{aligned}
 H_{\log L(\theta;x)} &= J\left(\frac{\nabla L(\theta;x)}{L(\theta;x)}\right) \\
 &= \frac{\partial}{\partial \theta} \left(\frac{\nabla L(\theta;x)}{L(\theta;x)}\right) \\
 &= \frac{H_{L(\theta;x)}L(\theta;x) - \nabla L(\theta;x)\nabla L(\theta;x)^T}{L(\theta;x)L(\theta;x)} \\
 &= \frac{H_{L(\theta;x)}L(\theta;x)}{L(\theta;x)L(\theta;x)} - \frac{\nabla L(\theta;x)\nabla L(\theta;x)^T}{L(\theta;x)L(\theta;x)} \\
 &= \frac{H_{L(\theta;x)}}{L(\theta;x)} - \left(\frac{\nabla L(\theta;x)}{L(\theta;x)}\right)\left(\frac{\nabla L(\theta;x)}{L(\theta;x)}\right)^T
 \end{aligned} \tag{A3}$$

Expectation on the Hessian matrix is

$$\begin{aligned}
 E_{L(\theta;x)} \left[H_{\log L(\theta;x)} \right] &= E_{L(\theta;x)} \left[\frac{H_{L(\theta;x)}}{L(\theta;x)} - \left(\frac{\nabla L(\theta;x)}{L(\theta;x)}\right)\left(\frac{\nabla L(\theta;x)}{L(\theta;x)}\right)^T \right] \\
 &= E_{L(\theta;x)} \left[\frac{H_{L(\theta;x)}}{L(\theta;x)} \right] - E_{L(\theta;x)} \left[\left(\frac{\nabla L(\theta;x)}{L(\theta;x)}\right)\left(\frac{\nabla L(\theta;x)}{L(\theta;x)}\right)^T \right] \\
 &= \int \frac{H_{L(\theta;x)}}{L(\theta;x)} L(\theta;x) dx - E_{L(\theta;x)} \left[\nabla \log L(\theta;x) \nabla \log L(\theta;x)^T \right] \\
 &= H_{\int L(\theta;x) dx} - I(\theta) = 0 - I(\theta) = -I(\theta)
 \end{aligned} \tag{A4}$$

Thus, we have

$$I(\theta) = -E_{L(\theta;x)} \left[H_{\log L(\theta;x)} \right] \tag{A5}$$

References

1. Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J.J. Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Trans. Robot.* **2016**, *32*, 1309–1332. [\[CrossRef\]](#)
2. Hess, W.; Kohler, D.; Rapp, H.; Andor, D. Real-time loop closure in 2D LIDAR SLAM. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016.
3. Ali, A.J.B.; Kouroshli, M.; Semenova, S.; Hashemifar, Z.S.; Ko, S.Y.; Dantu, K. Edge-SLAM: Edge-Assisted Visual Simultaneous Localization and Mapping. *ACM Trans. Embed. Comput. Syst.* **2022**, *22*, 18. [\[CrossRef\]](#)
4. Kuo, J.; Muglikar, M.; Zhang, Z.; Scaramuzza, D. Redesigning SLAM for Arbitrary Multi-Camera Systems. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020.
5. Lin, Y.; Hongwei, M.; Yan, W.; Jing, X.; Chuanwei, W. A Tightly Coupled LiDAR-Inertial SLAM for Perceptually Degraded Scenes. *Sensors* **2022**, *22*, 3063.
6. Shan, T.; Englot, B.; Meyers, D.; Wang, W.; Ratti, C.; Rus, D. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 5135–5142.
7. Forster, C.; Zhang, Z.; Gassner, M.; Werlberger, M.; Scaramuzza, D. SVO: Semidirect Visual Odometry for Monocular and Multicamera Systems. *IEEE Trans. Robot.* **2017**, *33*, 249–265. [\[CrossRef\]](#)
8. Engel, J.; Koltun, V.; Cremers, D. Direct Sparse Odometry. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 611–625. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [\[CrossRef\]](#)
10. Qin, T.; Li, P.; Shen, S. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [\[CrossRef\]](#)
11. Harmat, A.; Sharf, I.; Trentini, M. *Parallel Tracking and Mapping with Multiple Cameras on an Unmanned Aerial Vehicle*; Springer: Berlin/Heidelberg, Germany, 2012.
12. Xie, P.; Su, W.; Li, B.; Jian, R.; Huang, R.; Zhang, S.; Wei, J. Modified Keyframe Selection Algorithm and Map Visualization Based on ORB-SLAM2. In Proceedings of the 2020 4th International Conference on Robotics and Automation Sciences (ICRAS), Chengdu, China, 6–8 June 2020; pp. 142–147.
13. Das, A.; Waslander, S.L. Entropy based keyframe selection for Multi-Camera Visual SLAM. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–3 October 2015.
14. Jiao, J.; Ye, H.; Zhu, Y.; Liu, M. Robust Odometry and Mapping for Multi-LiDAR Systems with Online Extrinsic Calibration. *IEEE Trans. Robot.* **2020**, *38*, 351–371. [\[CrossRef\]](#)
15. Zhang, J.; Singh, S. LOAM: Lidar odometry and mapping in real-time. In Proceedings of the Robotics: Science and Systems, Berkeley, CA, USA, 12–16 July 2014; pp. 1–9.

16. Piao, J.-C.; Kim, S.-D. Real-Time Visual–Inertial SLAM Based on Adaptive Keyframe Selection for Mobile AR Applications. *IEEE Trans. Multimed.* **2019**, *21*, 2827–2836. [[CrossRef](#)]
17. Tang, X.; Fu, W.; Jiang, M.; Peng, G.; Wu, Z.; Yue, Y.; Wang, D. Place recognition using line-junction-lines in urban environments. In Proceedings of the 2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), Bangkok, Thailand, 18–20 November 2019; pp. 530–535.
18. Arandjelovic, R.; Gronat, P.; Torii, A.; Pajdla, T.; Sivic, J. NetVLAD: CNN architecture for weakly supervised place recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 5297–5307.
19. Peng, G.; Zhang, J.; Li, H.; Wang, D. Attentional pyramid pooling of salient visual residuals for place recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 885–894.
20. Peng, G.; Huang, Y.; Li, H.; Wu, Z.; Wang, D. LSDNet: A Lightweight Self-Attentional Distillation Network for Visual Place Recognition. In Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 23–27 October 2022; pp. 6608–6613.
21. Calonder, M.; Lepetit, V.; Strecha, C.; Fua, P. BRIEF: Binary Robust Independent Elementary Features. In Proceedings of the European Conference on Computer Vision, Crete, Greece, 5–11 September 2010.
22. Jin, S.; Wu, Z.; Zhao, C.; Zhang, J.; Peng, G.; Wang, D. SectionKey: 3-D Semantic Point Cloud Descriptor for Place Recognition. In Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 23–27 October 2022; pp. 9905–9910.
23. Fan, Y.; He, Y.; Tan, U.X. Seed: A Segmentation-Based Egocentric 3D Point Cloud Descriptor for Loop Closure Detection. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020.
24. Wang, H.; Wang, C.; Xie, L. Intensity Scan Context: Coding Intensity and Geometry Relations for Loop Closure Detection. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020.
25. Wang, Y.; Sun, Z.; Xu, C.-Z.; Sarma, S.E.; Yang, J.; Kong, H. Lidar iris for loop-closure detection. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 5769–5775.
26. He, L.; Wang, X.; Zhang, H. M2DP: A novel 3D point cloud descriptor and its application in loop closure detection. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, 9–14 October 2016.
27. Kim, G.; Kim, A. Scan Context: Egocentric Spatial Descriptor for Place Recognition within 3D Point Cloud Map. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018.
28. Chen, X.; Läbe, T.; Milioto, A.; Röhling, T.; Vysotska, O.; Haag, A.; Behley, J.; Stachniss, C. OverlapNet: Loop Closing for LiDAR-based SLAM. In Proceedings of the Robotics: Science and Systems XVI, Virtual Event, 12–16 July 2021.
29. Scovanner, P.; Ali, S.; Shah, M. A 3-dimensional sift descriptor and its application to action recognition. In Proceedings of the 15th ACM International Conference on Multimedia, Augsburg, Germany, 25–29 September 2007.
30. Sipiran, I.; Bustos, B. A Robust 3D Interest Points Detector Based on Harris Operator. In Proceedings of the Eurographics Workshop on 3D Object Retrieval, Norrköping, Sweden, 2 May 2010.
31. Tsourounis, D.; Kastaniotis, D.; Theoharatos, C.; Kazantzidis, A.; Economou, G. SIFT-CNN: When Convolutional Neural Networks Meet Dense SIFT Descriptors for Image and Sequence Classification. *J. Imaging* **2022**, *8*, 256. [[CrossRef](#)] [[PubMed](#)]
32. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
33. Censi, A. On achievable accuracy for range-finder localization. In Proceedings of the Robotics and Automation, Roma, Italy, 10–14 April 2007.
34. Casella, G.; Berger, R.L. *Statistical Inference*; Cengage Learning: Boston, MA, USA, 2021.
35. Censi, A. On achievable accuracy for pose tracking. In Proceedings of the Robotics and Automation, Kobe, Japan, 12–17 May 2009.

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