

Article

Incremental Pose Map Optimization for Monocular Vision SLAM Based on Similarity Transformation

Wenlei Liu ^{1,*} , Sentang Wu ¹, Zhongbo Wu ² and Xiaolong Wu ³ 

¹ School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China; woost@sina.com

² Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China; wuzhongbo17@mails.ucas.ac.cn

³ Navigation and Control Technology Institute of NORINCO Group, Beijing 100089, China; wuaero@buaa.edu.cn

* Correspondence: liuwenlei@buaa.edu.cn

Received: 16 October 2019; Accepted: 10 November 2019; Published: 13 November 2019



Abstract: The novel contribution of this paper is to propose an incremental pose map optimization for monocular vision simultaneous localization and mapping (SLAM) based on similarity transformation, which can effectively solve the scale drift problem of SLAM for monocular vision and eliminate the cumulative error by global optimization. With the method of mixed inverse depth estimation based on a probability graph, the problem of the uncertainty of depth estimation is effectively solved and the robustness of depth estimation is improved. Firstly, this paper proposes a method combining the sparse direct method based on histogram equalization and the feature point method for front-end processing, and the mixed inverse depth estimation method based on a probability graph is used to estimate the depth information. Then, a bag-of-words model based on the mean initialization K-means is proposed for closed-loop feature detection. Finally, the incremental pose map optimization method based on similarity transformation is proposed to process the back end to optimize the pose and depth information of the camera. When the closed loop is detected, global optimization is carried out to effectively eliminate the cumulative error of the system. In this paper, indoor and outdoor environmental experiments are carried out using open data sets, such as TUM and KITTI, which fully proves the effectiveness of this method. Closed-loop detection experiments using hand-held cameras verify the importance of closed-loop detection. This method can effectively solve the scale drift problem of monocular vision SLAM and has strong robustness.

Keywords: similarity transformation; incremental pose map; monocular vision SLAM; bag-of-words; sparse direct method; histogram equalization; probability graph

1. Introduction

With the continuous development of artificial intelligence technology, research of visual slam has also made great progress, and new achievements are emerging. This technology plays an important role in vision navigation. Vision slam is widely used in robot autonomous navigation, augmented reality, 3D reconstruction, unmanned driving, and other fields. According to the different equipment used, visual SLAM can be divided into visual SLAM based on a depth camera RGB-D, visual SLAM based on a binocular or structural camera, and visual SLAM based on a monocular camera. The difficulty of the above three methods gradually increases, and each method has its own advantages and disadvantages [1,2]. For an RGB-D camera, it enjoys the advantages of obtaining the depth information of the object more accurately, which is conducive to the accuracy of SLAM positioning; but it is limited by a narrow field of vision, limited measurement range, poor anti-interference capacity, and is mostly

used indoors. The binocular camera has a strong anti-interference capacity and depth information can be obtained utilizing its baseline. In addition, the binocular camera can be used indoors and outdoors. However, the camera configuration and calibration of the binocular camera are complex, and the positioning accuracy is restricted by the baseline of the binocular camera. Moreover, with a large amount of calculation, hardware equipment is usually required for acceleration. When it comes to the monocular camera, it is superior due to its simple structure, low cost, and wide range of use. The disadvantage is that the monocular camera cannot obtain the depth information of objects, which can only be obtained by the depth estimation method. Additionally, the monocular camera has the problem of scale drift, so its accuracy is relatively poor. This paper focuses on the problem of monocular visual SLAM. Aiming at the uncertainty of depth measurement by the monocular camera, an incremental pose map optimization method based on similarity transformation is proposed by using the inverse depth estimation method based on a probability graph for scale drift.

The optical flow method and direct method are widely used in visual SLAM. Usually, the calculated camera pose is used to provide a good initial value for further optimization. The optical flow method was introduced in detail, and some improvement measures were proposed for the traditional optical flow method, such as describing most of the algorithms and their extensions in a consistent framework, and an effective inverse synthesis algorithm is proposed in [3]. Engel et al. [4] gave a detailed introduction to the sparse direct method, which combines the minimization of photometric errors with the uniform joint optimization of all model parameters, independent of key points and descriptors. In [5,6], aiming at the low computational speed and lack of consistency of the direct method, a semi-direct method was proposed. This method eliminates the feature extraction and robust matching techniques of motion estimation, and has the characteristics of high accuracy, good robustness, and fast speed. In [7], a monocular vision inertial ranging algorithm based on the Extended Kalman Filter (EKF) method and the direct photometric error minimization method was proposed. The data of the vision and inertial measurement were fused. Aiming at the problems of the optical flow method and direct method in practical application, a sparse direct method based on histogram equalization was proposed, which can effectively guarantee the photometric invariance in the measurement process.

Depth estimation plays an important role in monocular vision SLAM. The accuracy of depth estimation will directly affect the location accuracy of monocular vision SLAM. Civera et al. [8] applied the inverse depth parametrization estimation to monocular vision SLAM, which can measure features in a large depth range and reduce the influence of parallax on depth estimation. However, directly parameterizing inverse depth increases the complexity of parameter optimization. Engel et al. [9] presented a novel real-time range measurement method by a monocular camera. The key of this method is to continuously estimate the half-density inverse depth map of the current frame and use the image alignment to track the camera motion. Perdices et al. [10] used semi-direct visual localization (SDVL) to estimate the inverse depth, and used the exception rejection mechanism to eliminate the dislocation features, thus improving the efficiency and accuracy of feature matching. In [11], a method of depth estimation based on Bayesian estimation and convex optimization image processing was proposed. Newcombe et al. [12] proposed a real-time density tracking and mapping method, which estimates detailed texture depth mapping on selected key frames to generate surface mosaics composed of vertices. Inverse depth estimation is widely used in visual odometry because of its high accuracy and good robustness. In this paper, a mixed inverse depth estimation method based on probability graph is proposed.

The estimation and optimization of the pose of the monocular camera is the core content of visual SLAM. There are many methods for pose optimization. The more accurate the pose estimation is, the higher the measurement accuracy is. Triggs et al. [13] gave a detailed introduction to the bundle adjustment methods commonly used in camera measurement, including the selection of the cost function, and the introduction of numerical optimization methods, like the sparse Newton method and the linear approximation recursive method. The re-projection was adopted to eliminate errors, which greatly improved the accuracy of estimation. Strasdat et al. [14] proposed a two-level optimization

method, which combines the precise point constraints of the main regions of interest with the stable edges of the position-attitude soft constraints, dynamically choosing the inner and outer window members and optimizing them simultaneously. Dubbelman et al. [15] proposed a closed-loop online pose chain method to solve the pose, which can accurately estimate the depth of visual odometry and reduce the influence of scale drift on pose estimation. In [16], an open-square smoothing filtering algorithm was proposed to estimate the pose of the camera. This method decomposes the correlation information matrix or the measured Jacobian matrix into square roots, so the calculation speed is faster, and the accuracy is higher. In [17], a new data structure, namely the Bayesian tree algorithm, was introduced. The method of a probability graph was used to infer the pose, and sparse matrix decomposition was combined so as to improve the speed and accuracy of operation. Kaess et al. [18] adopted incremental smoothing and mapping to decompose the information matrix, in which only the matrix items that actually changed were updated, thus improving the efficiency. In addition, the uncertainty estimation algorithm based on the factor information matrix was utilized to improve the real-time information processing. In order to reduce the amount of calculation and ensure the accuracy of the optimization, the incremental pose map optimization method is adopted in this paper.

The commonly used loop closure detection method for monocular vision SLAM is bag-of-words, which establishes a dictionary by extracting the features of all images, and then searches for matches through the dictionary. In addition to this, other methods have been proposed by researchers. Redmond et al. [19] presented a method of initializing the k-means clustering algorithm, which uses the kd-tree to estimate the density of data at different locations, and then uses a modification of the Katsavounidis algorithm. Arthur et al. [20] proposed a simple random seeding technique to increase k-means, and obtained a $\Theta(\log k)$ -competition and optimal clustering algorithm. In [21], a probabilistic recognition method based on position representation was proposed, which can not only be used for positioning, but also determine that a new observation comes from a new place to enhance its map. Cummins et al. [22] proposed a new SLAM formula based on the appearance only, which is suitable for very large-scale local recognition. The system navigates in outer space and assigns each new observation value to a new or previously visited location without reference to the metric location. Che et al. [23] proposed an image retrieval framework based on robust distance measurement and information fusion, which improved the retrieval performance of SVT relative to the baseline. In [24], a new method based on the stacked denoise autoencoder (SDA) was proposed. SDA is a multi-layer neural network, which learns compression representation from original input data independently in an unsupervised way. In order to prevent the dictionary from falling into the local optimum in the process of dictionary generation, a K-means clustering method based on mean initialization is proposed in this paper.

According to the different application environments and conditions of visual SLAM, scholars have proposed various solutions. In [25], a new direct tracking method based on *sim*(3) was proposed to detect scale drift explicitly and incorporate the effect of the noisy depth value into the tracking. The algorithm allows the construction of large-scale and consistent environmental maps. Mur-Artal et al. [26,27] presented a feature-based monocular SLAM system, namely Oriented FAST and Rotated BRIEF SLAM (ORB-SLAM). The system has strong robustness to severe motion clutter, allows wide baseline loops to be closed and repositioned, and includes automatic initialization. It has a faster operation speed and higher accuracy. Zhou et al. [28] proposed a semi-dense monocular simultaneous location and mapping (SLAM) method that can deal with pure camera rotation motion, and established a probabilistic depth map model based on Bayesian estimation. Frost et al. [29] proposed a novel monocular vision SLAM method. In addition to point measurement, target detection is also considered to solve the scale ambiguity and drift. By expecting the size of the object, the scale estimation can be seamlessly integrated into a bundle adjustment. In [30], the relative position of the first key frame was calculated by using the calibrated object, and the depth map was initialized. Meanwhile, the metric was introduced into the reconstruction, so that the absolute scale of the world coordinate system could be obtained.

In order to eliminate the influence of scale drift on positioning accuracy, an incremental pose map optimization method based on similarity transformation is proposed in this paper.

The system architecture of this paper is shown in Figure 1. It consists of three threads: Front-end processing, closed-loop detection, and back-end processing. Different processing methods are provided in different threads.

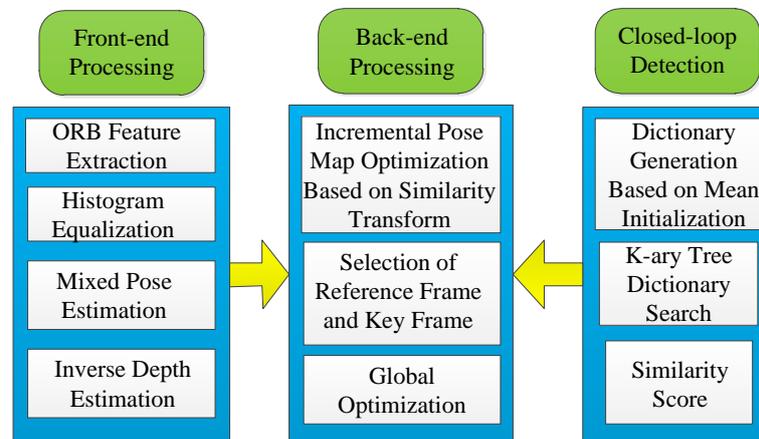


Figure 1. System schematic diagram of incremental pose map optimization for monocular vision SLAM based on similarity transformation.

In front-end processing, this paper combined the direct method with the feature point method, using the direct method in a certain threshold range, and the feature point method when it exceeded a certain threshold. In the direct method, the equalization based on a histogram is proposed, which can ensure that the assumption of gray invariance is satisfied in the process of the direct method. In this paper, a mixed inverse depth estimation method based on a probability graph was used to estimate the depth information of monocular vision, which not only improves the accuracy but also improves the robustness. In the closed-loop detection, this paper used the appearance-based bag-of-words. In the process of dictionary generation, the K-means clustering method based on mean initialization was proposed to ensure the global optimal characteristics of cluster centers. The data structure was analyzed by the Thiessen polygon, which is convenient for dictionary representation, and a similar scoring function is expressed by information theory. In the back-end processing, the main innovation is to propose a method of incremental pose optimization based on similarity transformation, which takes the uncertainty of scale into account to ensure the consistency of monocular vision slam. At the same time, the incremental pose optimization facilitates the expansion of observation nodes and reduces the computational complexity. Reference frames and key frames are determined according to the number of feature points, and global optimization is carried out by using the results of closed-loop detection. In addition, the detected closed-loop features can be used for global optimization to further reduce the error.

The organizational structure of this paper is as follows: Section 2 will focus on the front-end processing method based on the combination of the direct method and feature point method; Section 3 will focus on the closed-loop detection based on bag-of-words; Section 4 will mainly introduce the method of incremental pose map optimization based on similarity transformation; Section 5 will introduce indoor environment experiments, outdoor environment experiments, and closed-loop detection experiments of hand-held cameras; and Section 6 will summarize the methods provided in this paper.

2. Front-End Processing

In visual SLAM, the front end is called visual odometry (VO). According to the information of adjacent images, the motion of the camera is roughly estimated, and the initial value is provided for back-end processing. Common VO can be divided into two categories:

(a) Direct method: The direct method can be used to estimate the motion of the camera directly by using the gray information of the image, without calculating the key points and descriptors, as long as there are light and shade changes in the scene. The direct method uses the gradient size and direction of the local gray intensity to optimize, which gets rid of the dependence on image features. In the case of the camera motion being blurred, it has strong robustness. The direct method evolved from the optical flow method. The optical flow method describes the motion in the pixel, while the direct rule takes into account the motion model of the camera at the same time. The assumption of gray invariance must be taken into account when using direct method; that is, the grayscale value of a pixel in the same spatial point should remain unchanged in each image.

(b) Method based on feature points: The usual processing methods include the following: Firstly, distinct features, such as feature points or feature lines, are extracted from each image; secondly, invariant feature descriptors are used to match them; and then, epipolar geometry is used to restore the pose of the camera; finally, the pose is optimized by minimizing the re-projection error. The classical invariant feature extraction method is ORB feature extraction, which mainly consists of key points and descriptors. Through Features From Accelerated Segment Test (FAST) corner extraction, the obvious changes of local pixels can be obtained, and the main direction of feature points can be calculated at the same time. The Binary Robust Independent Elementary Features (BRISF) descriptor is used to describe the image region around the extracted feature points, which is suitable for real-time image matching. The advantage of the feature point method is that it runs stably, is insensitive to light and dynamic objects, and has strong robustness. The disadvantage is that the extraction of feature points and the calculation of descriptors are time-consuming, and the processing error of images without obvious texture features is relatively large.

The feature point method is suitable for scenes with abundant feature points, and is easy to extract. Because enough feature points contain more redundant information, the calculation of the pose will be more accurate. The direct method gets rid of the dependence on feature points, and calculates the pose of the camera by using the gray changes of the front and back frames. It can be applied to scenes with fewer feature points. In this paper, the direct method and the feature point method are combined to make full use of the advantages of both, so as to make full use of the advantages and avoid disadvantages. It not only extends the application scope and scene of the vision front-end but also improves the accuracy of front-end processing.

The results of front-end processing will be taken as the initial value of back-end optimization. The accuracy of the calculation will affect the results of the back-end optimization, and the speed of the calculation will affect the real-time performance of the whole SLAM. In order to ensure the accuracy of pose estimation and the speed of operation, a relatively simple switching method is adopted, which uses the number of feature points and gray changes to judge whether to use the feature point method or the direct method. When the number of feature points is less than a certain threshold and the gray level changes of the two frames are not obvious, the sparse direct method based on a histogram equalization is adopted; otherwise, front-end processing based on the ORB feature is used. According to the different application scenarios, the threshold can be adjusted appropriately to get better optimization results.

2.1. Sparse Direct Method Based on Histogram Equalization

In order to satisfy the hypothesis of an invariant gray level, histogram equalization is usually used to preprocess the image to eliminate the influence of strong and weak light on the gray level characteristics. The histogram is a statistical method that uses graphics to represent the distribution of data. Usually, the abscissa represents the grayscale value corresponding to the pixel, and the ordinate represents the number of statistical samples per pixel.

Histogram equalization is an image enhancement method that enhances image contrast by stretching the range of image pixels. For an image, the important feature pixels are usually concentrated in a certain gray interval. Therefore, stretching the uniform distribution of the gray interval in the whole gray level range can make the contrast of the features significantly enhanced and the image has a strong sense of clarity. Histogram equalization reduces the area with high grayscale value in the image, increases the area with low grayscale value, and compresses the gray level with a lesser number of pixels, expanding the gray interval with a greater number of pixels. The method of histogram equalization has the advantages of fast operation, better reflection of the details of the image, effective restraint of the drastic changes of brightness between images, reduction of the difference of the grayscale values between adjacent images, and better maintenance of the gray consistency of images.

In the sparse direct method, the estimation of camera pose can be computed with fewer feature points without calculating the descriptor or matching feature. The basic idea is to estimate the position of corresponding matching points by using the current camera pose. By minimizing the photometric error, the pose of the camera is optimized. The optimal error function is:

$$\min_{\xi} \Delta(\xi) = \sum_{i=1}^N \|e_i\|^2 = \sum_{i=1}^N \|I(p_{1,i}) - I(p_{2,i})\|^2 \quad (1)$$

where $I(p_{1,i})$ is the grayscale value of the pixel of the i -th feature point in the first image and $I(p_{2,i})$ is the grayscale value of the pixel in the second image corresponding to the i -th feature point in the first image. N is the number of feature points.

In the second image, the corresponding feature point, $p_{2,i}$, is obtained by the camera pose transformation. R is the rotation matrix; t is the translation vector; ξ is the Lie algebra of R and t ; ρ_i is the corresponding inverse depth information; and K is the camera's internal parameter matrix. The conversion formula is as follows:

$$p_{2,i} = \rho_i K(RP_i + t) = \rho_i K(\exp(\xi^\wedge)P_i) \quad (2)$$

In this paper, the perturbation model of Lie algebra is used to solve the error. After Lie algebra left-multiplies by a small perturbation, $\delta\xi$, we can get:

$$\begin{aligned} e(\xi \oplus \delta\xi) &= I_1(\rho_1 KP) - I_2(\rho_2 K \exp(\delta\xi^\wedge) \exp(\xi^\wedge)P) \\ &\approx I_1(\rho_1 KP) - I_2(\rho_2 K \exp(\xi^\wedge)P + \rho_2 K \delta\xi^\wedge \exp(\xi^\wedge)P) \end{aligned} \quad (3)$$

The coordinate of the perturbation component in the second camera coordinate system is set as q , namely, $q = \delta\xi^\wedge \exp(\xi^\wedge)P$, and its corresponding pixel coordinate is m , namely, $m = \rho_2 Kq = \rho_2 K \delta\xi^\wedge \exp(\xi^\wedge)P$. By using the first order Taylor formula, the error function can be transformed into the following formula:

$$\begin{aligned} e(\xi \oplus \delta\xi) &\approx I_1(\rho_1 KP) - I_2(\rho_2 K \exp(\xi^\wedge)P) - \frac{\partial I_2}{\partial m} \frac{\partial m}{\partial q} \frac{\partial q}{\partial \xi} \delta\xi \\ &= e(\xi) - \frac{\partial I_2}{\partial m} \frac{\partial m}{\partial q} \frac{\partial q}{\partial \xi} \delta\xi = e(\xi) - J(\xi) \delta\xi \end{aligned} \quad (4)$$

where $\frac{\partial I_2}{\partial m}$ is the gradient of pixels at m . $\frac{\partial m}{\partial q}$ is the derivative of the pixel coordinates in the camera coordinate system, and $\frac{\partial q}{\partial \xi}$ is the derivative of the spatial point to Lie algebra. By calculating the point $q = [X, Y, Z]^T$, $Z = 1/\rho$ in the camera coordinate system, the following formula can be obtained:

$$\frac{\partial m}{\partial \xi} = \frac{\partial m}{\partial q} \frac{\partial q}{\partial \xi} = \begin{bmatrix} \rho f_x & 0 & -\rho^2 f_x X & -\rho^2 f_x XY & f_x + \rho^2 f_x X^2 & -\rho f_x Y \\ 0 & \rho f_y & -\rho^2 f_y Y & -f_y - \rho^2 f_y Y^2 & \rho^2 f_y XY & \rho f_y X \end{bmatrix} \quad (5)$$

Therefore, we can find that the Jacobian matrix of Lie algebra is $J = -\frac{\partial I_2}{\partial m} \frac{\partial m}{\partial \xi}$. Using the LM algorithm to solve the error function, the optimized pose can be obtained.

2.2. Front-End Processing Based on ORB Features

Based on the front-end processing of ORB features, the ORB features of each image are extracted first, then the fast library for approximate nearest neighbor (FLANN) algorithm is used to match all features, and finally the position and pose are optimized by the bundle adjustment (BA) algorithm using matched feature points.

Assuming that the coordinates of a point in three-dimensional space are $U_i = [X_i, Y_i, Z_i]^T$, the coordinates transformed into camera coordinates are $U' = [X', Y', Z']^T, \rho' = 1/Z'$, and the projected pixel coordinates are $\mu_i = [u_i, v_i]^T$. The inner parameter matrix of the camera is K , and the Lie algebras of the camera's pose, R and t , are recorded as ξ . The transformation relationship between the spatial point and pixel coordinate system is as follows:

$$s_i \mu_i = K \exp(\xi^\wedge) U_i = K U' = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} \quad (6)$$

After eliminating the proportional coefficient, s_i , the error function can be obtained as follows:

$$\begin{cases} e_u = f_x \frac{X'}{Z'} + c_x - u_i = \rho' f_x X' + c_x - u_i \\ e_v = f_y \frac{Y'}{Z'} + c_y - v_i = \rho' f_y Y' + c_y - v_i \end{cases} \quad (7)$$

The Jacobian matrix, J_p , of the re-projection error can be obtained by deriving the pose using the chain rule:

$$J_p = \frac{\partial e}{\partial \delta \xi} = \frac{\partial e}{\partial U'} \frac{\partial U'}{\partial \delta \xi} \quad (8)$$

The first is the derivative of the error with respect to the projection point. It can be obtained that:

$$\frac{\partial e}{\partial U'} = \begin{bmatrix} \frac{\partial e_u}{\partial X'} & \frac{\partial e_u}{\partial Y'} & \frac{\partial e_u}{\partial Z'} \\ \frac{\partial e_v}{\partial X'} & \frac{\partial e_v}{\partial Y'} & \frac{\partial e_v}{\partial Z'} \end{bmatrix} = \begin{bmatrix} \frac{f_x}{Z'} & 0 & -\frac{f_x X'}{Z'^2} \\ 0 & \frac{f_y}{Z'} & -\frac{f_y Y'}{Z'^2} \end{bmatrix} \quad (9)$$

The second term is the derivative of the transformed point on Lie algebra. It can be obtained that:

$$\frac{\partial U'}{\partial \delta \xi} = \begin{bmatrix} I & U'^\wedge \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & -Z' & Y' \\ 0 & 1 & 0 & Z' & 0 & -X' \\ 0 & 0 & 1 & -Y' & X' & 0 \end{bmatrix} \quad (10)$$

Therefore, the Jacobian matrix of the re-projection error can be obtained as follows:

$$J_p = \frac{\partial e}{\partial \delta \xi} = \begin{bmatrix} \rho' f_x & 0 & -\rho'^2 f_x X' & \rho'^2 f_x X' Y' & -f_x - \rho'^2 f_x X'^2 & \rho' f_x Y' \\ 0 & \rho' f_y & -\rho'^2 f_y Y' & f_y + \rho'^2 f_y Y'^2 & -\rho'^2 f_y X' Y' & -\rho' f_y X' \end{bmatrix} \quad (11)$$

The $\min_{\xi} \Delta(\xi) = \sum_{i=1}^N \|e_i\|^2 = \sum_{i=1}^N \|h(x_i) - \mu_i\|^2$ is solved by using the method of non-linear least squares, and the optimal pose is finally obtained.

2.3. Mixed Inverse Depth Estimation Based on the Probability Graph

In the process of initialization, depth estimation is very important to the SLAM operation results of monocular vision. However, the accuracy and robustness of depth estimation are poor and vulnerable to external noise. In [31,32], in order to improve the accuracy and robustness, a mixed inverse depth estimation method based on a probability map is adopted.

Figure 2 is the probability diagram of the Gauss-uniform mixture probability distribution model [32]. Assume that $X = \{x_1, \dots, x_N\}$ is the inverse depth value of the sensor observation, $\rho = \{\rho_1, \dots, \rho_N\}$ is the real inverse depth value, $\pi = \{\pi_1, \dots, \pi_N\}$ is the proportion of good measurement data, $1-\pi$ is the proportion of interference, and the interference signal obeys the uniform distribution form, $U[\rho_{\min}, \rho_{\max}]$. ρ_{\min}, ρ_{\max} are the minimum inverse depth and the maximum inverse depth measured by the sensor, respectively. $\lambda = \{\lambda_1, \dots, \lambda_N\}$ is the accuracy of the Gauss distribution, where $\lambda = \frac{1}{\tau}$, in which τ is the variance of the Gauss distribution. Given the true inverse depth, ρ , the accuracy, λ , of the Gauss distribution, and the proportional coefficient, π , of the correct data, the probability distribution of the inverse depth measured by the mixed model is as follows:

$$p(x_n | \rho_n, \lambda_n, \pi_n) = \pi_n N(x_n | \rho_n, \lambda_n^{-1}) + (1 - \pi_n) U(x_n) \tag{12}$$

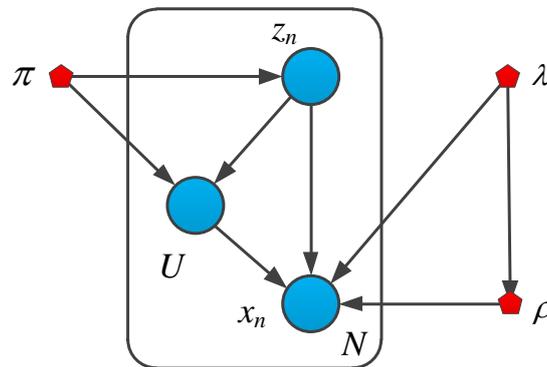


Figure 2. Probability diagram of Gauss-uniform mixed probability distribution model.

All potential discrete variables are recorded as $Z = \{z_{1k}, z_{2k}, \dots, z_{nk}\}$, in which z_{ik} is a binary random variable. Using the expression of “1-of-K”, one element is 1, the rest are 0. Among them, $z_{i1} = 1$, which means the i th measurement value is good measurement data, and $z_{i0} = 0$ which means the i th measurement value is interference data. Therefore, the distribution of potential variables is as follows:

$$p(x_n | \rho_n, \lambda_n, \pi_n, z_{nk}) = (\pi_n N(x_n | \rho_n, \lambda_n^{-1}))^{z_{nk}} ((1 - \pi_n) U(x_n))^{1-z_{nk}} \tag{13}$$

In this paper, the conjugate prior probability distribution of parameter ρ, λ, π is introduced, in which the conjugate prior probability of the mixing coefficient, π , obeys a Beta distribution, and the conjugate prior distribution of the mean, ρ , and precision, λ , is a Gauss-Gamma distribution.

According to the Bayesian theorem, $posterior \propto likelihood \times prior$, the joint probability distribution of all random variables can be obtained in the form of:

$$P(X, Z, \pi, \rho, \lambda) = p(X | Z, \rho, \lambda) p(Z | \pi) p(U | Z, \pi) p(\rho | \lambda) p(\lambda) p(\pi) \tag{14}$$

By using the method of variational inference to estimate the parameters, the recursive formula of inverse depth estimation can be obtained.

3. Closed-Loop Detection

With the sequence movement of the camera, the camera pose and the depth information estimated by depth estimation will produce errors. Periodic global back-end optimization for the whole SLAM system will eliminate some errors, but there will still be some accumulated errors. In order to eliminate cumulative errors, a closed-loop detection method is usually used, which eliminates the cumulative errors in a larger time scale through similar data collected by the camera in the same place, so that the whole SLAM has a globally consistent estimation.

Appearance-based bag-of-words (BoW) is a common method for the detection of a closed loop. Bag-of-words usually uses the features of an image to describe an image. A word is used to represent a feature, and all the features are composed into a dictionary. Therefore, in the matching process, only the word vectors contained in each image need to be matched. The bag-of-words improves the efficiency and accuracy of closed-loop detection.

3.1. Dictionary Generation

Dictionary generation is a clustering problem and an unsupervised machine learning method. The learning classification is automatically performed according to the features extracted from the image, which improves the efficiency of dictionary generation. The common clustering method is the K-means clustering method, which is simple and effective. In this paper, a K-means clustering method based on mean initialization is proposed, so that the selection of clustering centers is not trapped in the local optimum, so as to ensure its global optimum characteristics.

Assuming that D is the feature point data set and C is the cluster center set, the K-means clustering method based on mean initialization is [33]:

(1) Center point initialization: The coordinates of all feature points are summed up, and then the average is taken. The feature point closest to the average is taken as the first cluster center, c_1 . If many points are identical, the point near the edge is selected as the first cluster center, and c_1 is added to set C . Then, the feature point furthest from c_1 is selected as the second cluster center, c_2 , which is also added to set C . Then, other centers are chosen according to the following formula. Among them, $\min_{c_j \in C}(d(x_i, c_j))$ means to select the center point nearest to the feature point, calculate its Euclidean distance, and then select the feature point corresponding to the largest distance among all the distances as the new cluster center, and add it to set C :

$$\max_{x_i \in D} \left(\min_{c_j \in C} (d(x_i, c_j)) \right) \quad (15)$$

(2) For each feature point, the distance between each feature point and each center point is calculated and the point is classified into the nearest center points.

(3) According to the existing classification, the average value of feature points in each class is used to recalculate the central point, c_j , of each class, and then the sum of squares, D , of the new cumulative distance is calculated, where n is the number of all feature points:

$$D = \sum_{i=1}^n \left(\min_{c_j \in C} (d(x_i, c_j)) \right)^2 \quad (16)$$

(4) If the change of D is small and convergent, then the algorithm converges, and the selected center point is suitable. Otherwise, the calculation will be restarted, returning to the second step.

Through the K-means clustering algorithm, feature points can be divided into K specific classes, so as to generate K words; that is, K words represent an image. The words of all images can form a dictionary. The dictionary is sorted according to the nature of the words and a vector is generated. This is not only convenient for the generation of dictionaries but also for the word searching. According to the words provided by the dictionary, feature matching can be carried out efficiently.

3.2. Layered Thiessen Polygon Data Structure

In order to facilitate the establishment and search of dictionaries, a K-ary tree is usually used to express a dictionary, while the K-ary tree can be abstractly represented by a Thiessen polygon [34], and its hierarchical way is shown in the following figure.

Figure 3 shows the generation process of the K-ary tree dictionary by using the Thiessen polygon. The classification using the Thiessen polygon is mainly based on the fact that the distance from any feature point in a Thiessen polygon to the feature control point constituting the polygon is less than

the distance to the feature control point of other polygons. The feature control point here is the cluster center of the K-means clustering method. This paper takes the ternary tree as an example to introduce its principle. Among that, the green point is the control point. According to the main control point of each layer, a multi-layer Thiessen polygon can be generated. The classification results of each layer use the K-means clustering algorithm initialized by the mean. In the figure, A_1 , A_2 , and A_3 are the results of the first layer classification; A_{11} , A_{12} , and A_{13} are the results of the classification of A_1 ; and A_{121} , A_{122} , and A_{123} are the results of the classification of A_{12} . Among them, the whole image is equivalent to the root node of the K-ary tree, the number of classifications in each layer is equivalent to the branch of the K-ary tree, and the number of layers is the depth of the K-ary tree. According to the above method, the whole image can be divided into n regions, which are the leaf nodes of the K-ary tree. All the leaf nodes make up n words of the whole image.

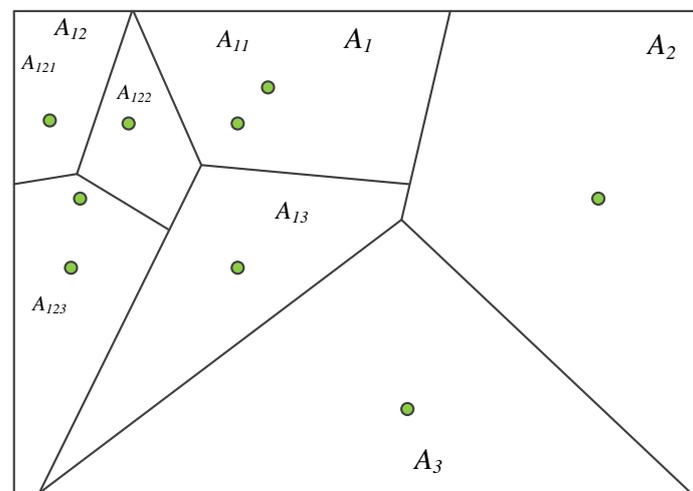


Figure 3. A schematic diagram of the generation process of a K-ary tree dictionary using the Thiessen polygon.

The words generated by the layered Thiessen polygon are convenient for the dictionary generation and word searching of the dictionary, which ensures the searching efficiency of the dictionary.

3.3. Computation of the Similarity Scoring Function

Assuming that the number of occurrences of the word W_i in an image is n_i and the image contains n words, the probability of the occurrence of the word W_i in an image is $P(W_i) = \frac{n_i}{n}$. According to information theory, it can be concluded in this paper that in the same image, each word contains a different amount of information. The words with a higher occurrence frequency often contain less information, while the words with a lower probability contain more information. The logarithmic function of inverse probability is usually used to express the amount of information contained in a word with the expression of $\log\left(\frac{1}{P(W_i)}\right) = -\log(P(W_i)) = \log\left(\frac{n}{n_i}\right)$. In information theory, the degree of uncertainty of a word measured in an image is expressed by self-entropy as follows [35]:

$$g(W_i) = P(W_i) \log\left(\frac{1}{P(W_i)}\right) = -P(W_i) \log(P(W_i)) = \frac{n_i}{n} \log\left(\frac{n}{n_i}\right) \quad (17)$$

Therefore, in an image, the uncertainty vector of the word W_i composed by all words in the dictionary is as follows:

$$G = [g(W_1), g(W_2), \dots, g(W_m)]^T \quad (18)$$

where m is the number of all words contained in the dictionary.

Supposing that G_a is used to represent the uncertainty vector of image a and G_b is used to represent that of image b , the similarity between the two images can be calculated with the similarity score function. The expression is as follows:

$$Y = 1 - \left\| \frac{G_a}{\|G_a\|_2} - \frac{G_b}{\|G_b\|_2} \right\|_1 = 1 - \sum_{i=1}^m \left| \frac{g_a(W_i)}{\|G_a\|_2} - \frac{g_b(W_i)}{\|G_b\|_2} \right| \quad (19)$$

When the value of the similarity score reaches a certain threshold, it can be considered that image a is similar to image b and a closed loop is constituted. With the closed-loop characteristic, the accumulative error caused by the camera in the sequence motion process can be eliminated, and the positioning accuracy of visual SLAM can be greatly improved.

4. Back-End Processing

The front-end processing calculates the pose of the moving object, and its error is relatively large. Usually, this calculation result provides the initial value for the back-end optimization of the pose, and selects the key frame from the sequence image for optimization. The initial value of the scale factor is provided by the inverse depth estimation method based on a probability graph. The back end of this paper uses the method based on a similar transformation to optimize, and considers the role of the scale factor. In short, from the front end to the back end is a rough-to-fine continuous optimization process. The front-end processes the data preliminarily, and the back end processes the data more precisely, so that the positioning accuracy can be continuously improved while ensuring the real-time performance.

In monocular vision SLAM, one of the most important problems is scale uncertainty and scale drift. This paper presented an incremental pose map optimization method based on the similarity transformation group, which effectively solves the problems of rotation, translation, and scale drift in closed-loop detection.

Similarity transformation is a combination of Euler transformation and uniform transformation. Generally, the matrix of similar transformation is expressed as:

$$Sim(3) = \left\{ S = \begin{bmatrix} sR & t \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4} \right\}, s \in \mathbb{R}^+(3), R \in SO(3), t \in \mathbb{R}^3 \quad (20)$$

where s is the scale scaling factor; R is the 3×3 rotation matrix; and t is the 3×1 translation vector. There is a total of seven degrees of freedom. The invariants of similar transformation are the ratio of two lengths and the included angle between two straight lines.

$sim(3)$, the corresponding Lie algebra of $Sim(3)$ is:

$$sim(3) = \left\{ \zeta | \zeta = \begin{bmatrix} \rho \\ \phi \\ \sigma \end{bmatrix} \in \mathbb{R}^7, \zeta^\wedge = \begin{bmatrix} \sigma I + \phi^\wedge & \rho \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{4 \times 4} \right\} \quad (21)$$

where the element of Lie algebra $sim(3)$ is a seven-dimensional vector, ζ ; ρ corresponds to the translation part; ϕ corresponds to the rotation part; and σ corresponds to the scale scaling factor. In the derivation process of this paper, the relevant formulas about similarity transformation are shown in Appendix A.

Figure 4 is a schematic diagram of an incremental pose map optimization method based on similarity transformation, in which each node represents the relevant random quantity. Among them, x_i represents the node of the camera pose; l_i is the landmark node; o_{ij} is the observation node of the j -th landmark by the i -th camera; f_{ij} is the pose error factor function, representing the pose relationship error value of the j -th camera relative to the i -th camera; and q_i is the re-projection error factor function, representing the re-projection error of the i -th camera.

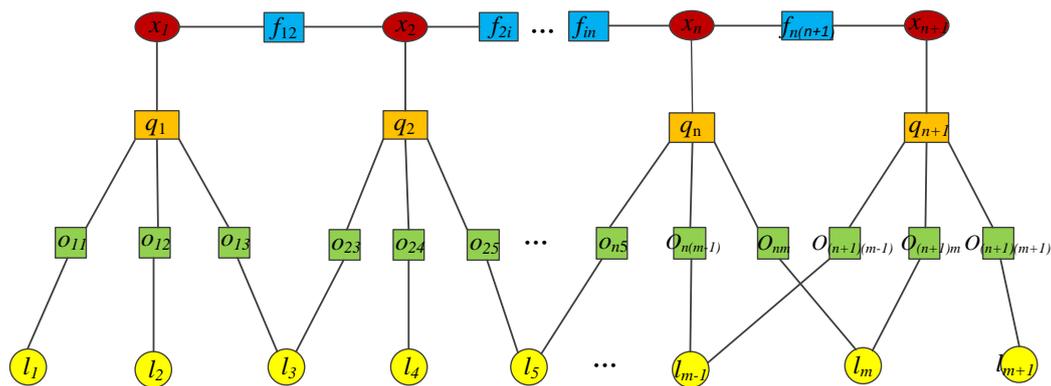


Figure 4. Schematic diagram of the incremental pose map optimization method.

As can be seen from Figure 4, when the camera moves to the new position, x_{n+1} , the pose of x_{n+1} will be optimized by the re-projection error factor function, q_{n+1} . The pose factor function, $f_{n(n+1)}$, is used to correlate with the original pose, x_n , and the relative pose variation is determined to obtain the motion information of the camera. When the camera moves to a new position, the above process will be repeated. Therefore, the formation of an incremental motion calculation process only requires calculation of the motion relationship between the new and the original pose. It does not need to optimize the global information every time, which avoids a lot of repeated calculation, reduces the calculation cost, and improves the calculation speed.

The selection of key frames and reference frames has an important impact on the results of incremental pose map optimization. In this paper, the following principles are usually applied to the selection of key frames: 1. After global optimization, 20 frames are passed. 2. At least 100 feature points can be extracted from the image. 3. The number of matching points of the current frame is reduced by 80% compared with the reference frame. In general, key frames with less than 20% matching points of the reference frames are regarded as new reference frames.

In order to improve the computing speed and reduce the data storage, redundant key frames need to be removed. The method is to delete the remaining two key frames when the matching degree of feature points of the key frames of three consecutive images exceeds 90% and the middle frame is retained, which can reduce the number of key frames. At the same time, it reduces unnecessary closed loops and improves the efficiency of closed-loop detection.

Assuming that the pose error factor function is a zero-mean Gauss distribution:

$$f_{ij}(x_i, x_j) = \frac{1}{\sqrt{2\pi * \Sigma_{ij}^2}} \exp\left(-\frac{(\ln(S_{ij}^{-1}S_i^{-1}S_j))^2}{2\Sigma_{ij}^2}\right) \quad (22)$$

where $S_{ij} = S_i^{-1}S_j$, S_{ij} is the similarity transformation of pose x_j relative to pose x_i ; S_i is the similarity transformation of pose x_i relative to the original pose x_0 ; and S_j is the similarity transformation of pose x_j relative to the original pose x_0 . Σ_{ij} is the covariance matrix of the pose error. Usually, in order to simplify the calculation, it can be made as a unit matrix.

Supposing that the re-projection error factor function satisfies the following form of a Gaussian distribution:

$$q_i(x_i) = N(h(x_i)|\mu_i, \sigma_i^2) = \frac{1}{\sqrt{2\pi * \sigma_i^2}} \exp\left(-\frac{(h(x_i) - \mu_i)^2}{2\sigma_i^2}\right) \quad (23)$$

where $h(x_i)$ is the re-projection of the i -th camera; μ_i is the corresponding pixel coordinate; $h(x_i) - \mu_i$ is the re-projection error; and σ_i^2 is the variance of the re-projection.

The joint probability distribution of all variables is expressed in the form of a factor product:

$$F(X) = \frac{1}{Z} \prod_i q_i(x_i) \prod_{i,j} f_{ij}(x_i, x_j) \quad (24)$$

where $Z = \sum \prod_{i,j} q_i(x_i) \prod_{i,j} f_{ij}(x_i, x_j)$, Z is usually called the normalization factor to ensure that $F(X)$ is a properly defined probability. It is difficult to accurately calculate the value of Z , and in practice, it is not necessary to obtain the exact value of Z .

The maximum posteriori probability inference takes the form of:

$$X_{\max} = \operatorname{argmax}_X F(X) = \operatorname{argmax}_X \frac{1}{Z} \prod_i q_i(x_i) \prod_{i,j} f_{ij}(x_i, x_j) \quad (25)$$

After taking logarithm of the upper formula, ignoring the normalization factor and other constant terms, the maximum posteriori probability inference problem can be transformed into the problem of a nonlinear least squares sum:

$$X_{\max} = \operatorname{argmin}_X \frac{1}{2} \left(\sum_i \|h(x_i) - \mu_i\|^2 + \sum_{i,j} (\ln(S_{ij}^{-1} S_i^{-1} S_j)) \right)^2 \quad (26)$$

There is an important defect in the formula. If there is a mismatch in the calculation, and the error is very large, it will dominate the whole error, erase the influence of other correct errors, and produce wrong results. In order to solve the problem of the large error and fast growth of two norms, this paper adopted the Huber kernels method. The improved formula is as follows:

$$\begin{aligned} X_{\max} &= \operatorname{argmin}_X \left(\sum_i H(h(x_i) - \mu_i) + \sum_{i,j} H(\ln(S_{ij}^{-1} S_i^{-1} S_j)) \right), \\ H(h(x_i) - \mu_i) &= \begin{cases} \frac{1}{2} (h(x_i) - \mu_i)^2, & |h(x_i) - \mu_i| \leq \beta_1, \\ \beta_1 (|h(x_i) - \mu_i| - \frac{1}{2} \beta_1), & \text{other.} \end{cases} \\ H(\ln(S_{ij}^{-1} S_i^{-1} S_j)) &= \begin{cases} \frac{1}{2} (\ln(S_{ij}^{-1} S_i^{-1} S_j))^2, & \ln(S_{ij}^{-1} S_i^{-1} S_j) \leq \beta_2, \\ \beta_2 (\ln(S_{ij}^{-1} S_i^{-1} S_j) - \frac{1}{2} \beta_2), & \text{other.} \end{cases} \end{aligned} \quad (27)$$

where β_1 is the threshold of the re-projection error and β_2 is the threshold of the pose error. The main function of the Huber kernel function is that when the error exceeds the threshold, it changes from a quadratic function to a primary function, which limits the growth rate of the error and weakens the influence of mismatching on the optimization results.

Firstly, the Jacobian matrix, J_p , of the re-projection error function is solved by the BA method, and the re-projection function is linearized to obtain:

$$h(x_i) = h(x_i^0 + \Delta x_i) = h(x_i^0) + J_p \Delta x_i \quad (28)$$

where J_p is the Jacobian matrix with re-projection error, $\Delta x_i = x_i - x_i^0$, and x_i^0 is the initial pose.

Therefore, the error function can be expressed in the following form:

$$\begin{cases} e^2 = \|h(x_i^0) + J_p \Delta x_i - \mu_i\|^2 = \|J_p \Delta x_i - b\|^2 \\ b = \mu_i - h(x_i^0) \end{cases} \quad (29)$$

Then the pose error is solved. S_{ij} , S_i , and S_j measurements can be obtained by a front-end visual odometry. The pose error function is as follows:

$$e_{ij} = \ln(S_{ij}^{-1}S_i^{-1}S_j)^\vee = \ln\left(\exp\left((-c_{ij})^\wedge\right)\exp\left((-c_i)^\wedge\right)\exp\left((c_j)^\wedge\right)\right)^\vee \quad (30)$$

Next, derivation is made for c_i and c_j , respectively. According to the derivation method of Lie algebra, c_i and c_j left-multiply by perturbations δc_i and δc_j , respectively, so the error function becomes as follows:

$$\hat{e}_{ij} = \ln\left(S_{ij}^{-1}S_i^{-1}\exp\left((-\delta c_i)^\wedge\right)\exp\left((\delta c_j)^\wedge\right)S_j\right)^\vee \quad (31)$$

By using the adjoint matrix property of similar transformation, the perturbation term can be moved to the rightmost side of the function. As a result, the Jacobian matrix in the form of right multiplication can be derived by using the Baker Campbell Hausdorff (BCH) formula. The error function is as follows:

$$\begin{aligned} \hat{e}_{ij} &= \ln\left(S_{ij}^{-1}S_i^{-1}S_j\exp\left((-Ad_s(S_j^{-1})\delta c_i)^\wedge\right)\exp\left((Ad_s(S_j^{-1})\delta c_j)^\wedge\right)\right)^\vee \\ &\approx \ln\left(S_{ij}^{-1}S_i^{-1}S_j\left[I - (Ad_s(S_j^{-1})\delta c_i)^\wedge + (Ad_s(S_j^{-1})\delta c_j)^\wedge\right]\right)^\vee \approx e_{ij}^0 + \frac{\partial e_{ij}}{\partial \delta c_i}\delta c_i + \frac{\partial e_{ij}}{\partial \delta c_j}\delta c_j \end{aligned} \quad (32)$$

where e_{ij}^0 is the initial observation error value. The Jacobian matrix for c_i and c_j is:

$$\begin{cases} \frac{\partial e_{ij}}{\partial \delta c_i} = -J_r(e_{ij})^{-1}Ad(S_j^{-1}) \\ \frac{\partial e_{ij}}{\partial \delta c_j} = J_r(e_{ij})^{-1}Ad(S_j^{-1}) \end{cases} \quad (33)$$

where:

$$J_r(e_{ij})^{-1} = \left(I + \frac{1}{2} \begin{bmatrix} \omega^\wedge + vI & \tau^\wedge & -\tau \\ 0 & \omega^\wedge & 0 \\ 0 & 0 & 0 \end{bmatrix} \right), Ad(S_j^{-1}) = \begin{bmatrix} sR & t^\wedge R & -t \\ 0 & R & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (34)$$

In the incremental pose map optimization method based on a factor graph, the usual processing method is to fix the optimized pose, and only optimize the pose of the newly added points. In Equation (32), $\delta c_i = 0$ is usually used, so the error function can be simplified as follows:

$$\hat{e}_{ij} \approx e_{ij}^0 + \frac{\partial e_{ij}}{\partial \delta c_j}\delta c_j = e_{ij}^0 + J_r(e_{ij})^{-1}Ad(S_j^{-1})\delta c_j = e_{ij}^0 + J_e\delta c_j \quad (35)$$

where the Jacobian matrix of pose error is $J_e = J_r(e_{ij})^{-1}Ad(S_j^{-1})$.

Therefore, the pose error can be written as follows:

$$e_{ij}^2 = \frac{1}{2}(e_{ij}^0 + J_e\delta c_j)^2 \quad (36)$$

Taking the first three rows of the Jacobian matrix of the pose error, e_{ij} , we can get a 3×7 Jacobian matrix, J'_e . In order to ensure the calculation of the minimization equation, the Jacobian matrix of the re-projection error, e , should be expanded to a 3×7 Jacobian matrix, J'_p , and a one-dimensional scale variable shall be added to the variable ξ to become a seven-dimensional variable, c_j , and b is also extended to a seven-dimensional vector, b' . Therefore, the re-projection error can be written as:

$$e^2 = \|J'_p\Delta x_i - b\|^2 = \|J'_p\delta c_j - b'\|^2 \quad (37)$$

The non-linear least squares formula using the Huber kernel function is changed to:

$$X_{\max} = \underset{X}{\operatorname{argmin}} \left(\sum_i H(h(x_i) - \mu_i) + \sum_{i,j} H(\ln(S_{ij}^{-1} S_i^{-1} S_j)) \right), \quad (38)$$

$$= \begin{cases} \frac{1}{2} \|J'_p \delta c_j - b'\|^2 + \frac{1}{2} (e_{ij}^0 + J_e \delta c_j)^2, & |h(x_i) - \mu_i| \leq \beta_1, \text{ and, } \ln(S_{ij}^{-1} S_i^{-1} S_j) \leq \beta_2, \\ \frac{1}{2} \|J'_p \delta c_j - b'\|^2 + \beta_2 \left((e_{ij}^0 + J_e \delta c_j) - \frac{1}{2} \beta_2 \right), & |h(x_i) - \mu_i| \leq \beta_1, \text{ and, other,} \\ \beta_1 \left(\|J'_p \delta c_j - b'\| - \frac{1}{2} \beta_1 \right) + \frac{1}{2} (e_{ij}^0 + J_e \delta c_j)^2, & \ln(S_{ij}^{-1} S_i^{-1} S_j) \leq \beta_2, \text{ and, other.} \end{cases}$$

In the above part, the problem of optimizing the pose by re-projection and similarity transformation is transformed into the problem of least squares optimization using the process in which the camera pose and scale scaling factor are optimized by using the incremental pose map based on a factor graph. The selection of reference frames is significant for the optimization. In this paper, a simple method was proposed for the selection of reference frames. Firstly, the current reference frame is used as the reference frame to match the feature points of the image of its next key frame, and the number of matching feature points is counted and is taken as the reference matching number, m_{base} , of the current reference frame. Then, the remaining sequence images are respectively matched with the reference frame by feature points, and the number m_{other} is counted. The number of m_{other} is compared with the number of references matching m_{other} . If $m_{\text{other}} \geq 0.2 * m_{\text{base}}$, the original reference frame remains unchanged. If $m_{\text{other}} < 0.2 * m_{\text{base}}$, the reference frame needs to be changed and the current key frame is taken as a new reference frame. The above operations are conducted for all the key frames of the remaining sequence images, which can automatically complete the updated iteration of the reference frame and improve the efficiency of the whole optimization process.

In the whole process of visual SLAM, global optimization plays a very important role, because it can reduce the pose error and scale drift error of the whole system in visual SLAM, improve the measurement accuracy, and ensure more accurate navigation and positioning. The conditions of global optimization are as follows: 1. When the closed loop appears in the system, the closed-loop characteristic is needed to optimize the whole system; and 2. when the reference frame passes through 50 frames and there is no closed loop in it, the whole information is needed to carry out a global optimization.

Generally, the global optimization method is to use the equation information of all incremental pose map optimization and calculate this by taking all pose Lie algebras, c_i , as unknown variables, so as to obtain the optimal solution of all poses. This result can be used as the initial value of the next optimization.

5. Experiments

The data set used in the indoor and outdoor environment experiments in this paper mainly comes from the open data set on the Internet, including the TUM data set, KITTI data set, and EuRoC MAV data set [26,27]. The method provided in this paper was compared with the widely used OBR_SLAM2 method. In the closed-loop detection experiment, in order to ensure the sufficiency of the experiment, a USB camera was used to collect image data, and the real-time image was used for the experiment.

5.1. Indoor Environmental Experiments

The indoor environment experiment mainly adopts the TUM data set and EuRoC MAV data set. The experimental results are shown in the following figure.

In Figure 5, the black line is the track curve drawn according to the true value provided by the data set, the blue line is the track drawn according to the actual measured key frame, and the red line represents the distance error between the measured value and the corresponding true value. According to the error values obtained in Figure 5, the experimental error comparison results in Table 1 can be obtained.

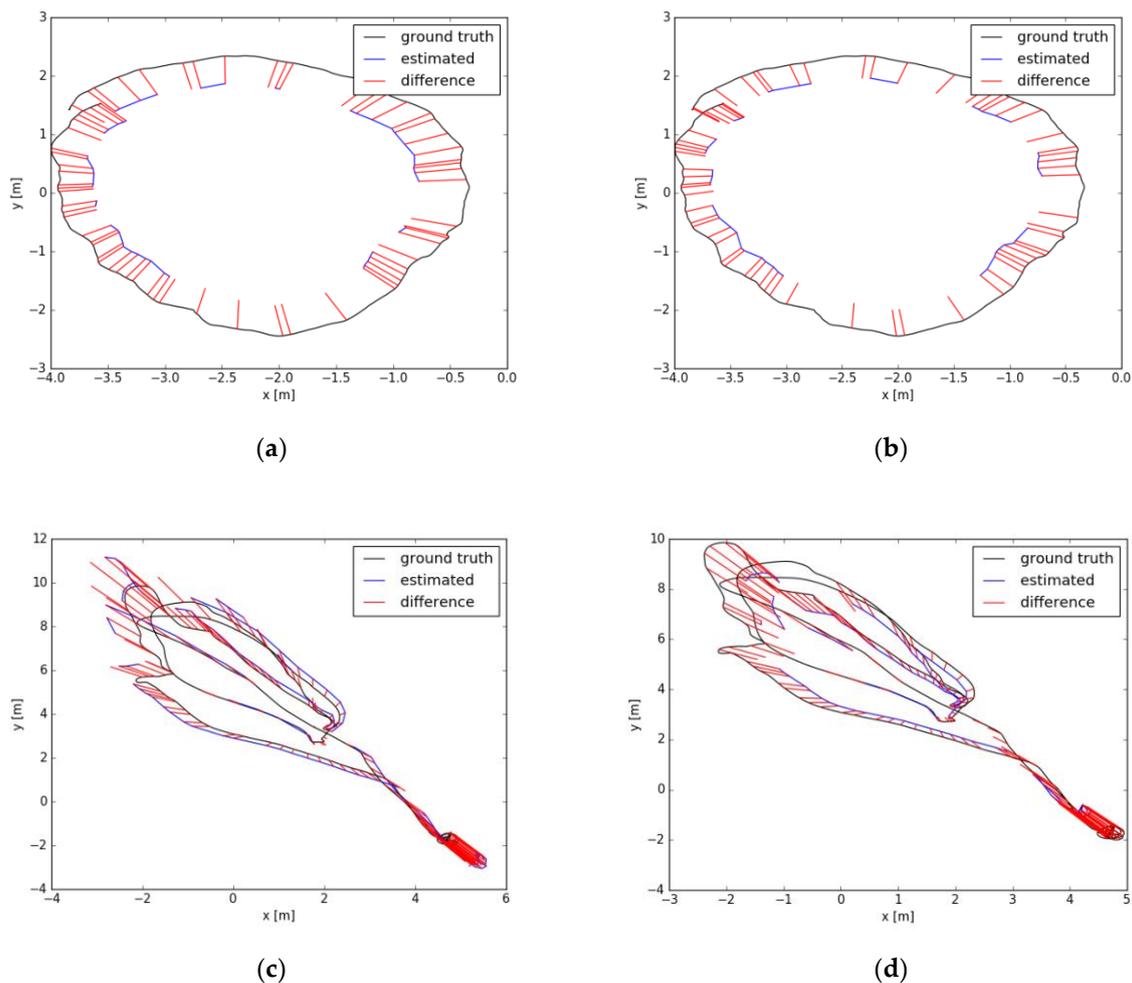


Figure 5. Comparison of TUM and EuRoC data sets before and after optimization. (a) Before TUM data set optimization; (b) After TUM data set optimization; (c) Before EuRoC data set optimization; (d) After EuRoC dataset optimization.

Table 1. Comparison of indoor experimental errors.

	TUM Data Set		EuRoC Data Set	
	ORB-SLAM2	Incremental Pose SLAM	ORB-SLAM2	Incremental Pose SLAM
Root mean square error (RMSE)	0.41 m	0.34 m	0.86 m	0.75 m
Mean error	0.40 m	0.34 m	0.76 m	0.66 m
Median error	0.41 m	0.35 m	0.81 m	0.69 m
Standard deviation of error	0.05 m	0.04 m	0.40 m	0.36 m
Minimum error	0.31 m	0.26 m	0.06 m	0.06 m
Maximum error	0.51 m	0.44 m	1.50 m	1.36 m
Average tracking time (s)	0.037 s	0.038 s	0.042 s	0.043 s

The results of Table 1 show that the incremental pose SLAM presented in this paper outperforms the experimental results of ORB-SLAM2. The root mean square error, mean error, median error, standard deviation of error, minimum error, and maximum error are all reduced, therefore, the positioning accuracy is further improved. The main reason is that the incremental pose optimization based on similarity transformation not only considers the optimization of the camera's own pose but also takes into account the optimization of the relative pose based on the scale factor, which greatly improves the estimation accuracy of the camera pose. In this paper, inverse depth estimation based on a probability graph not only improves the accuracy of depth estimation but also improves its robustness.

The average tracking time in the experiment is relatively stable, as it benefits from the stable light intensity of the indoor environment. Additionally, the combination of the sparse direct method and

the feature point method combines the advantages of the direct method and the feature point method, which improves the robustness of the tracking. Therefore, in the course of the experiment, there was no tracking failure.

The experimental comparison shows that the accuracy of using the TUM data set is better than that using the EuRoC data set. The main reason is that most of the TUM indoor data sets were obtained by hand-held cameras with slow speeds, which are relatively stable, and easy for tracking and depth estimation, so the error is relatively small. However, the data of the EuRoC data set was collected by Unmanned Air Vehicle (UAV) with greater volatility, which is susceptible to interference, and the image quality is relatively poor, affecting the estimation of depth, so the error is relatively large.

5.2. Outdoor Environmental Experiments

The KITTI data set was used for the outdoor experiment. It collects real image data of urban, rural, and highway environments by a car-mounted camera. It provides a large number of outdoor environment materials, which can evaluate the performance of computer vision in a vehicle environment. In this paper, sequences 08 data sets were used to carry out experiments. In the experiment, we used the GPS data as the benchmark to compare the experimental results of traditional ORB-SLAM2 and incremental pose SLAM. In Figure 6, the dotted line is the track drawn according to the GPS data, and the solid line is the track drawn according to the key frame in the experiment. The comparative results of outdoor experiments based on the experimental results are shown in Table 2.

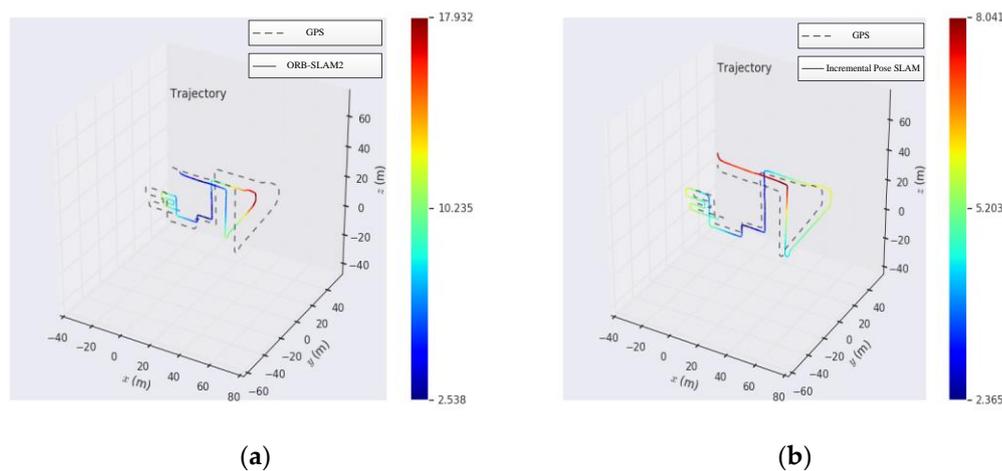


Figure 6. Experimental trajectory image of the KITTI dataset. (a) Comparison of GPS and ORB_SLAM2; (b) Comparison of GPS and incremental pose SLAM.

Table 2. Comparison of outdoor experimental errors.

	ORB-SLAM2	Incremental Pose SLAM
Root mean square error (RMSE)	10.52 m	6.25 m
Mean error	10.21 m	5.98 m
Median error	10.11 m	6.02 m
Standard deviation of error	4.68 m	2.01 m
Minimum error	3.89 m	2.13 m
Maximum error	18.98 m	10.63 m

The experimental results show that the trajectory of incremental pose SLAM is relatively smooth. Compared with GPS data, the experimental error is relatively small, and the scale drift phenomenon is relatively small. However, the relative error of traditional ORB-SLAM2 is relatively large, and the scale drift phenomenon is relatively serious. In this paper, an incremental pose map optimization

SLAM based on similarity transformation is proposed to mitigate the influence of scale drift on the experimental results.

Generally speaking, the trajectories obtained by experiments show that in urban and rural roads, the trajectories obtained are relatively accurate and the positioning accuracy is relatively high because of the relatively slow speed and the number of feature points that can be extracted, which is convenient for camera positioning, depth estimation, and tracking. On the freeway, because of the fast speed, it may appear that the speed of computer image processing is slower than that of the vehicle, resulting in tracking failure, and fewer feature points can be extracted on the freeway, and its stability is relatively poor, thus reducing the positioning accuracy.

An important characteristic of the outdoor environment relative to the indoor environment is that the outdoor environment is susceptible to the influence of light. In the same environment, the feature extraction will change obviously in the light direction and backlight, which increases the difficulty of feature matching. By comparing the results, we can see that the sparse direct method based on histogram equalization proposed in this paper can effectively reduce the influence of light on feature points, ensure the assumption of photometric invariance, and improve the robustness of feature extraction. Therefore, the tracking effect of incremental pose SLAM is more stable and can be applied to various experimental environments. When there are loops in the motion of the vehicle camera, this method can extract features effectively, and use the bag-of-words model to complete image matching, so as to optimize the whole situation to eliminate errors and improve the accuracy of global positioning.

When there are fewer feature points or moving too fast, the camera easily loses its target. The repositioning method based on the bag-of-words adopted in this paper can effectively complete the camera relocation. The combination of the sparse direct method and the feature point method can effectively process images with relatively few feature points and accurately estimate the camera pose.

5.3. Closed-Loop Detection Experiment of A Hand-Held Camera

In this paper, a 720P USB camera was used for experiments. In the experiment, the IMU was fixed on the camera, and the hand-held camera was moved with the computer to get real-time images, and the data of the IMU were used as benchmark data for comparison, and the results of the closed loop detection were judged according to the comparison results. The experimental results are shown in the following figure.

It can be seen from Figure 7 that SLAM without closed-loop detection is prone to static error, especially when the handheld camera turns, its error increases relatively, and with the increase of the running route and time, its error will become larger and larger. When the camera passes through the same point, its cumulative error will be close to 4 m. However, when closed-loop detection is added to monocular vision SLAM, the scale drift and cumulative error of the monocular camera can be effectively reduced, and the positioning accuracy can be improved, and its RMSE can be reduced to 0.62 m. In this paper, closed-loop detection based on bag-of-words was mainly used. The bag-of-words uses a dictionary to represent the features of each image, and image matching can be done by searching the dictionary of each image, thus eliminating the cumulative errors caused by the camera in motion. In the process of dictionary generation, the K-means clustering method based on mean initialization can effectively prevent cluster centers from falling into the local optimum and improve the accuracy of image feature classification.

In order to test the performance of the closed-loop detection based on mean initialization proposed in this paper, many experiments were carried out, and the recall precision experimental data were obtained. In this paper, we used the data sets of TUM, KITTI, and EuRoC to experiment. We compared the closed-loop detection based on ORB features, the closed-loop detection based on the line band descriptor (LBD), and the closed-loop detection method proposed in this paper [36–38]. The experimental results are shown in Figure 8.

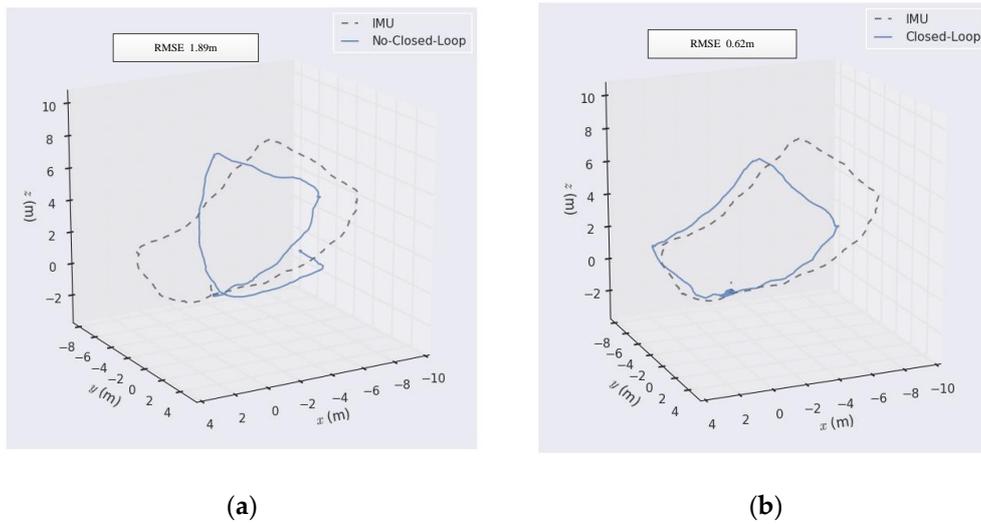


Figure 7. The experiment of the hand-held camera with or without closed-loop detection. (a) Comparison of the IMU and no-closed loop; (b) Comparison of IMU and closed-loop.

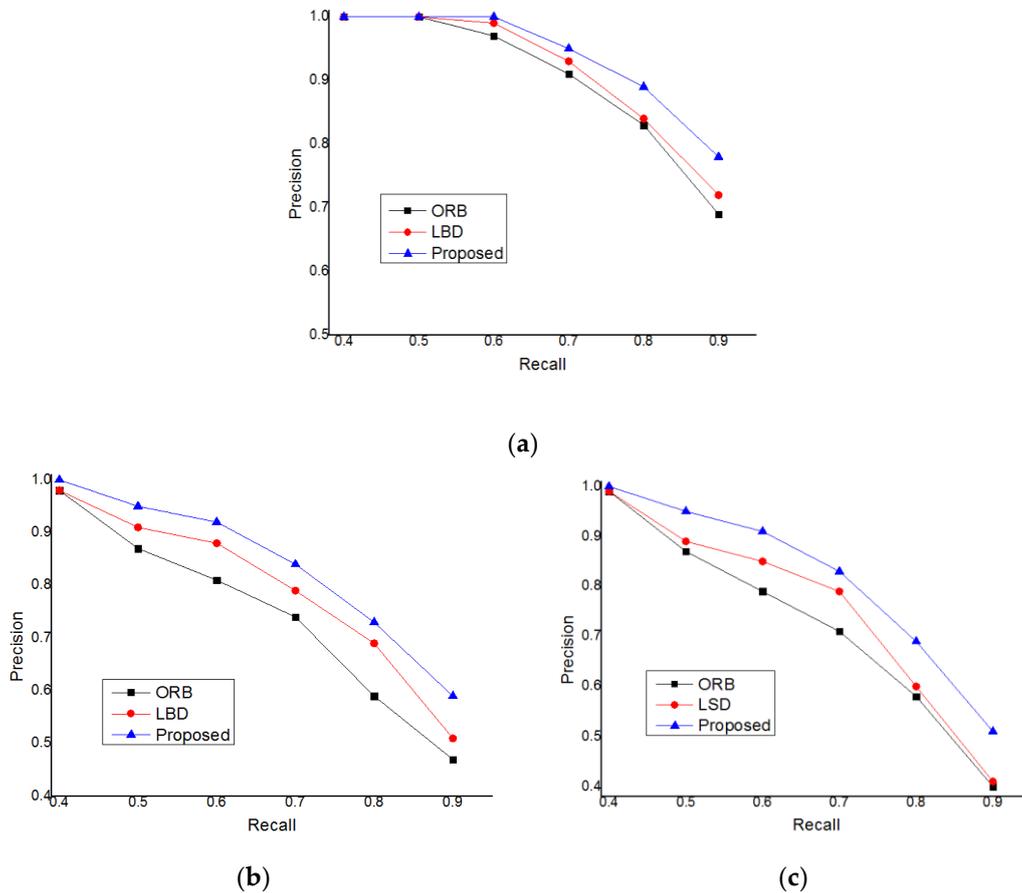


Figure 8. Precision-recall curves achieved by ORB, LBD, and proposed method. (a) Based on the data set of TUM; (b) Based on the data set of KITTI; (c) Based on the data set of EuRoC.

It can be seen from the experimental results that in the data sets of TUM, KITTI, and EuRoC, the experiments are carried out respectively, but the final experimental results are the same. The measurement accuracy is: Proposed > LBD > ORB. The method proposed in this paper improves the accuracy of closed-loop detection, which can effectively reduce the experimental error and improve the positioning accuracy of visual slam.

Through the experiments of hand-held cameras, we can also see that one of the major drawbacks of monocular SLAM is the uncertainty of scale. If there is no closed-loop detection, after the camera moves a circle, the starting point and the end point have a large offset. In order to improve the accuracy of closed-loop detection, the inverse depth estimation method based on a probability graph is used in front-end processing to improve the accuracy of initial depth estimation; in back-end optimization, the incremental pose optimization method based on similarity transformation is used to improve the accuracy of camera pose estimation. When the closed loop is detected in the camera motion process, global optimization will be carried out at the same time to further improve the accuracy of the scale factor and camera pose, so that the cumulative error of the whole system can be eliminated in a larger time scale.

The effectiveness and robustness of the proposed method were fully verified by indoor and outdoor environmental experiments using open data sets. Through the closed-loop detection experiment of the hand-held camera, the necessity of closed-loop detection in monocular vision SLAM was fully verified. Closed-loop detection reduced the overall error and ensured the accuracy of the system.

6. Conclusions

Visual odometry (VO) is a preliminary estimation of the pose and depth of the camera in monocular SLAM. Usually, the quality of the initial values has a significant impact on the whole system, and with good initial values, satisfactory results can be produced through optimization. In this paper, the direct method and the feature point method were used for pose estimation. The feature points were selected through the sparsity. When the number of feature points was less than a certain threshold, the direct method was used; otherwise, the feature point method was adopted. In order to ensure the photometric invariance of the direct method, histogram equalization was proposed to process the image to reduce the influence of photometric changes on the results. In this paper, a mixed inverse depth estimation method based on a probability graph was used to estimate the camera depth, which not only improved the estimation accuracy but also improved the robustness.

In the closed-loop detection, the bag-of-words was used to process. A dictionary was built by extracting the features of each image, and feature matching was carried out by looking up the dictionary. Bag-of-words improved the efficiency and accuracy of image matching. In the dictionary generation stage, a K-means clustering method based on mean initialization was proposed, which ensured the global optimum of the cluster center. In the whole process of SLAM, when the closed loop was detected by using the bag-of-words, the information of all nodes stored in the back-end optimization would be used for global optimization, so as to eliminate the cumulative error of the system, improve the accuracy of positioning and mapping, and ensure the consistency of the whole process.

In the back-end optimization, an incremental pose map optimization method based on similarity transformation was proposed. During the optimization process, the scale factor was taken into account by using similarity transformation to reduce the influence of scale drift on the results. The application of incremental pose map optimization contributed to the expansion of nodes. When a new node was added, only the pose between the new node and the associated node was optimized, which greatly reduced the computational complexity and improved the computational speed.

In conclusion, incremental pose map optimization SLAM for monocular vision based on similarity transformation can reduce the scale drift of a monocular camera, reduce the computational complexity of optimization, and improve the positioning accuracy, which has strong robustness.

Author Contributions: Conceptualization and Methodology, W.L. and S.W.; Software and Data Curation, W.L. and Z.W.; Writing—Original Draft Preparation, W.L.; Writing—Review & Editing, X.W.

Funding: This work was funded by the Industrial Technology Development Program under Grant B1120131046.

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

Appendix A

The properties of the similarity transformation used in the derivation in this paper are as follows. The exponential mapping of $sim(3)$ is as follows:

$$\begin{aligned} \exp(\zeta^\wedge) &= \begin{bmatrix} e^\sigma \exp(\phi^\wedge) & W\rho \\ 0 & 1 \end{bmatrix}, \\ W &= \frac{e^\sigma - 1}{\sigma} I + \frac{A\sigma + (1-B)\theta}{\sigma^2 + \theta^2} a^\wedge + \left(\frac{e^\sigma - 1}{\sigma} - \frac{(B-1)\sigma + A\theta}{\sigma^2 + \theta^2} \right) a^\wedge a^\wedge, \\ \theta &= \|\phi\|_2, a = \frac{\phi}{\theta}, A = e^\sigma \sin(\theta), B = e^\sigma \cos(\theta) \end{aligned} \quad (A1)$$

By exponential mapping, the correspondence relations between the Lie groups of similar transformations and Lie algebras can be obtained:

$$s = e^\sigma, R = \exp(\phi^\wedge), t = W\rho \quad (A2)$$

The logarithmic mapping of $sim(3)$ is:

$$\log \begin{bmatrix} sR & t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \sigma I + \phi^\wedge & W^{-1}t \\ 0 & 0 \end{bmatrix} \quad (A3)$$

The adjoint matrix of $sim(3)$ is:

$$Ad_S = \begin{bmatrix} sR & t^\wedge R & -t \\ 0 & R & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (A4)$$

The product of two exponential mappings of Lie algebra is expressed linearly by the BCH (Baker–Campbell–Hausdorff) formula as follows:

$$\ln(\exp(\phi_1^\wedge) \exp(\phi_2^\wedge))^\vee \approx \begin{cases} J_l(\phi_2)^{-1} \phi_1 + \phi_2, \phi_1 \text{ tiny}, \\ \phi_1 + J_r(-\phi_1)^{-1} \phi_2, \phi_2 \text{ tiny}. \end{cases} \quad (A5)$$

where J_l is the left-multiplied approximation and J_r is the right-multiplied approximation.

The Lie brackets of $Sim(3)$ are:

$$\left[\begin{pmatrix} \rho \\ \phi \\ \sigma \end{pmatrix}, \begin{pmatrix} \tau \\ \omega \\ \nu \end{pmatrix} \right] = \begin{pmatrix} \phi \times \tau + \rho \times \omega + \sigma \tau - \nu \rho \\ \phi \times \omega \\ 0 \end{pmatrix} \quad (A6)$$

The Jacobian matrix in Li brackets is:

$$\frac{\partial}{\partial \zeta} [\zeta, \vartheta] = \begin{bmatrix} -\omega^\wedge - \nu I & -\tau^\wedge & \tau \\ 0 & -\omega^\wedge & 0 \\ 0 & 0 & 0 \end{bmatrix}, \zeta = \begin{pmatrix} \rho \\ \phi \\ \sigma \end{pmatrix}, \vartheta = \begin{pmatrix} \tau \\ \omega \\ \nu \end{pmatrix} \quad (A7)$$

References

1. Bahraïni, M.S.; Rad, A.B.; Bozorg, M. SLAM in Dynamic Environments: A Deep Learning Approach for Moving Object Tracking Using ML-RANSAC Algorithm. *Sensors* **2019**, *19*, 3699. [[CrossRef](#)] [[PubMed](#)]
2. Ma, S.; Bai, X.; Wang, Y.; Fang, R. Robust Stereo Visual-Inertial Odometry Using Nonlinear Optimization. *Sensors* **2019**, *19*, 3747. [[CrossRef](#)] [[PubMed](#)]
3. Baker, S.; Matthews, I. Lucas-kanade 20 years on: A unifying framework. *Int. J. Comput. Vis.* **2004**, *56*, 221–255. [[CrossRef](#)]

4. Engel, J.; Koltun, V.; Cremers, D. Direct sparse odometry. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 611–625. [[CrossRef](#)] [[PubMed](#)]
5. Forster, C.; Zhang, Z.; Gassner, M.; Scaramuzza, D. SVO: Semidirect visual odometry for monocular and multicamera systems. *IEEE Trans. Robot.* **2016**, *33*, 249–265. [[CrossRef](#)]
6. Forster, C.; Pizzoli, M.; Scaramuzza, D. SVO: Fast semi-direct monocular visual odometry. In Proceedings of the 2014 IEEE international conference on robotics and automation (ICRA), Hong Kong, China, 31 May–5 June 2014.
7. Tanskanen, P.; Naegeli, T.; Pollefeys, M.; Hilliges, O. Semi-direct EKF-based monocular visual-inertial odometry. In Proceedings of the In Proceedings of the 2015 IEEE/RSJ international conference on intelligent robots and systems (IROS), Hamburg, Germany, 28 September–3 October 2015.
8. Civera, J.; Davison, A.J.; Montiel, J.M.M. Inverse depth parametrization for monocular SLAM. *IEEE Trans. Robot.* **2008**, *24*, 932–945. [[CrossRef](#)]
9. Engel, J.; Sturm, J.; Cremers, D. Semi-dense visual odometry for a monocular camera. In Proceedings of the IEEE international conference on computer vision, Darling Harbour, Sydney, 1–8 December 2013.
10. Perdices, E.; Cañas, J.M. SDVL: Efficient and accurate semi-direct visual localization. *Sensors* **2019**, *19*, 302. [[CrossRef](#)]
11. Pizzoli, M.; Forster, C.; Scaramuzza, D. REMODE: Probabilistic, monocular dense reconstruction in real time. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–5 June 2014.
12. Newcombe, R.A.; Lovegrove, S.J.; Davison, A.J. DTAM: Dense tracking and mapping in real-time. In Proceedings of the 2011 international conference on computer vision, Barcelona, Spain, 6–13 November 2011.
13. Triggs, B.; McLauchlan, P.F.; Hartley, R.I.; Fitzgibbon, A.W. Bundle adjustment—A modern synthesis. In Proceedings of the International workshop on vision algorithms, Corfu, Greece, 21–22 September 1999.
14. Strasdat, H.; Davison, A.J.; Montiel, J.M.M.; Konolige, K. Double window optimisation for constant time visual SLAM. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011.
15. Dubbelman, G.; Browning, B. COP-SLAM: Closed-form online pose-chain optimization for visual SLAM. *IEEE Trans. Robot.* **2015**, *31*, 1194–1213. [[CrossRef](#)]
16. Dellaert, F.; Kaess, M. Square Root SAM: Simultaneous localization and mapping via square root information smoothing. *Int. J. Robot. Res.* **2006**, *25*, 1181–1203. [[CrossRef](#)]
17. Kaess, M.; Johannsson, H.; Roberts, R.; Ila, V.; Leonard, J.J.; Dellaert, F. iSAM2: Incremental smoothing and mapping using the Bayes tree. *Int. J. Robot. Res.* **2012**, *31*, 216–235. [[CrossRef](#)]
18. Kaess, M.; Ranganathan, A.; Dellaert, F. iSAM: Incremental smoothing and mapping. *IEEE Trans. Robot.* **2008**, *24*, 1365–1378. [[CrossRef](#)]
19. Redmond, S.J.; Heneghan, C. A method for initialising the K-means clustering algorithm using kd-trees. *Pattern Recognit. Lett.* **2007**, *28*, 965–973. [[CrossRef](#)]
20. Arthur, D.; Vassilvitskii, S. k-means++: The advantages of careful seeding. In Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics, New Orleans, Louisiana, 7–9 January 2007.
21. Cummins, M.; Newman, P. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *Int. J. Robot. Res.* **2008**, *27*, 647–665. [[CrossRef](#)]
22. Cummins, M.; Newman, P. Appearance-only SLAM at large scale with FAB-MAP 2.0. *Int. J. Robot. Res.* **2011**, *30*, 1100–1123. [[CrossRef](#)]
23. Che, C.; Yu, X.; Sun, X.; Yu, B. Image retrieval by information fusion based on scalable vocabulary tree and robust Hausdorff distance. *EURASIP J. Adv. Signal Process.* **2017**, *2017*, 21. [[CrossRef](#)]
24. Gao, X.; Zhang, T. Unsupervised learning to detect loops using deep neural networks for visual SLAM system. *Auton. Robot.* **2017**, *41*, 1–18. [[CrossRef](#)]
25. Engel, J.; Schöps, T.; Cremers, D. LSD-SLAM: Large-scale direct monocular SLAM. In Proceedings of the European conference on computer vision, Zurich, Switzerland, 6–12 September 2014.
26. 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](#)]

27. Mur-Artal, R.; Tardós, J.D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [[CrossRef](#)]
28. Zhou, Y.; Yan, F.; Zhou, Z. Handling pure camera rotation in semi-dense monocular SLAM. *Vis. Comput.* **2019**, *35*, 123–132. [[CrossRef](#)]
29. Frost, D.; Prisacariu, V.; Murray, D. Recovering stable scale in monocular SLAM using object-supplemented bundle adjustment. *IEEE Trans. Robot.* **2018**, *34*, 736–747. [[CrossRef](#)]
30. Li, Y.; Wang, S.; Yang, D.; Sun, D. Metric online monocular SLAM by using a known reference. In Proceedings of the 2016 12th World Congress on Intelligent Control and Automation (WCICA), GuiLin, China, 12–15 June 2016.
31. Liu, W.; Wu, S.; Wu, X.; Zhao, H. Calibration method based on the image of the absolute quadratic curve. *IEEE Access* **2019**, *7*, 29856–29868. [[CrossRef](#)]
32. Liu, W.; Wu, S.; Wu, X.; Li, K. Mixed Probability Inverse Depth Estimation Based on Probabilistic Graph Model. *IEEE Access* **2019**, *7*, 72591–72603. [[CrossRef](#)]
33. Nister, D.; Stewenius, H. Scalable recognition with a vocabulary tree. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 7–22 June 2006.
34. Robertson, S. Understanding inverse document frequency: On theoretical arguments for IDF. *J. Doc.* **2004**, *60*, 503–520. [[CrossRef](#)]
35. Aizawa, A. An information-theoretic perspective of tf-idf measures. *Inf. Process. Manag.* **2003**, *39*, 45–65. [[CrossRef](#)]
36. Lee, T.; Kim, C.; Cho, D.D. A monocular vision sensor-based efficient SLAM method for indoor service robots. *IEEE Trans. Ind. Electron.* **2018**, *66*, 318–328. [[CrossRef](#)]
37. Dong, R.; Wei, Z.; Liu, C.; Kan, J. A Novel Loop Closure Detection Method Using Line Features. *IEEE Access* **2019**, *7*, 111245–111256. [[CrossRef](#)]
38. Mukherjee, A.; Chakraborty, S.; Saha, S.K. Detection of loop closure in SLAM: A DeconvNet based approach. *Appl. Soft Comput.* **2019**, *80*, 650–656. [[CrossRef](#)]



© 2019 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 (<http://creativecommons.org/licenses/by/4.0/>).