

## Modified Indoor Navigation Mapping based on VSLAM

Zhilin Lin <sup>a</sup>, Guoliang Zhang <sup>b</sup>, Erliang Yao <sup>c</sup>, Xiao Jia <sup>d</sup> and Hui Xu <sup>e</sup>

Xi'an new high tech research institute, Xi'an 710025, China

<sup>a</sup>zl13468619594@163.com, <sup>b</sup>15091281454@163.com, <sup>c</sup>2448302309@qq.com,

<sup>d</sup>2462329075@qq.com, <sup>e</sup>2472417108@qq.com

**Keywords:** VSLAM, Navigation Mapping, Random Sample Consensus (RANSAC), Path Planning.

**Abstract.** Since the 2D maps built by traditional method are limited to describe single environment plane, the VSLAM-based indoor 2D map building method is proposed. The nonlinear optimization problem about camera poses is constructed based on VSLAM, and this problem is solved by Gauss-Newton method. Subsequently, the point cloud map of indoor environment is built. Then, based upon the RANSAC algorithm, the equation of ground plane is obtained. The point cloud of robots' passage area is projected onto the ground to form the grid map, and the indoor maps can be built by updating the grid occupied state based on Bayesian method. Experiments show that, compared with the 2D maps built by the traditional method, the maps built by our method contain the 3D structure information of environment. It helps robots avoid the suspended objects and the low objects successfully in the path planning experiment.

### 1. Introduction

Autonomous navigation is the basis and prerequisite for mobile robots to perform various tasks. The autonomous navigation of mobile robot needs to know the map information of the environment in advance. CAD (Computer Aided Design) and SLAM (Simultaneous localization and mapping) are commonly used in map building. The precision of CAD mapping technology is higher, but it needs the professional persons to operate, and the preparatory work is heavier. SLAM refers to robots' self-location in an unknown environment. The space environment model is established simultaneously [1]. SLAM only need robot to carry a sensor to inspect the environment. It operates simpler. With the progress of the sensor technology, SLAM has gradually improved the accuracy of the map. Therefore, SLAM is an important method for navigation mapping. The traditional SLAM methods use a laser or sonar to create 2D maps [2-3]. However, due to the complexity of the ground environment, and the limitation that the laser scanning range is a single plane, the mobile robot is prone to collision with the suspended object or the low object if it moves based on the map built by the traditional method. VSLAM (Visual Simultaneous localization and mapping), which only need a visual sensor to build a 3D environment map, can describe a complex environment structure fully. However, a 3D map need large storage space, and it will require high performances for computing platform when it is used to path planning. Therefore, it will be an effective method for the indoor navigation mapping to combine the 3D maps and the 2D maps. In this paper, for the sake of simplicity, the maps which are built by the traditional method are named as the traditional maps, and the maps which are built by the method proposed in this paper are named as the VSLAM-based maps.

In this paper, a method of indoor navigation map based on VSLAM is proposed to solve the limitation of traditional 2D map building methods. The specific process is shown in Fig.1. After building a 3D point cloud map of environment based on VSLAM, a 2D grid map can be generated by projecting the point cloud, which is within the height range of the robot, onto the ground plane. Also, a method for solving the ground plane equation based on RANSAC is proposed before intercepting the point cloud. The influence of outliers on the solution precision is eliminated to enhance the further projection of 3D point cloud and the maps' precision. The 2D navigation map is built by using this method. It can provide full 3D environmental information in path planning. So, the path planning can be more accurate and

robust than the traditional methods. Experiments illustrate that, compared with the traditional maps, the VSLAM-based maps have a higher practical value.

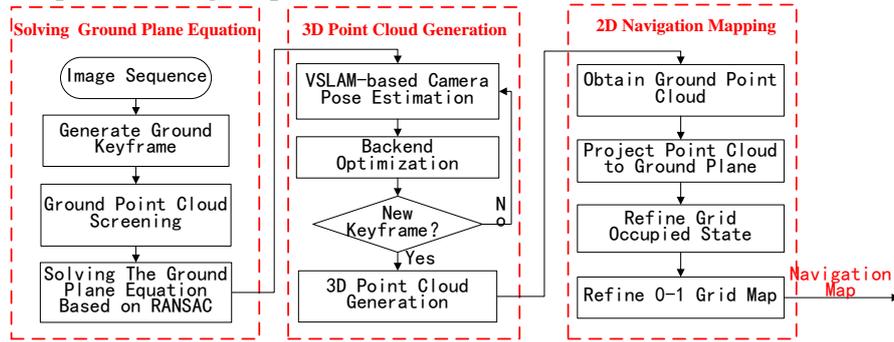


Fig.1 The procedure of 2D indoor navigation mapping based on VSLAM

## 2. 3D Point Cloud Map Generation Based on VSLAM

3D point cloud map generation based on VSLAM includes the following three parts: camera pose estimation, backend optimization, 3D point cloud generation.

### 2.1 Camera pose estimation

The position of the camera is represented by  $\xi \in se(3)$ , its exponential mapping is  $G = \exp_{se(3)}(\xi)$ . For a point  $P_w$  in the world coordinate system, its representation in the camera coordinate system is:

$$P_c = \exp_{se(3)}(\xi)P_w \quad (1)$$

Assuming the camera gets two frames:  $\mathcal{F}_1$  and  $\mathcal{F}_2$ , and a set of features are extracted from them:  $P \in \mathcal{F}_1$ ,  $Q \in \mathcal{F}_2$ . According to each feature's descriptor, two frames' features are matched to make them correspondent one by one:

$$P' = \{p_1, p_2, \dots, p_N\} \subset P \quad (2)$$

$$Q' = \{q_1, q_2, \dots, q_N\} \subset Q \quad (3)$$

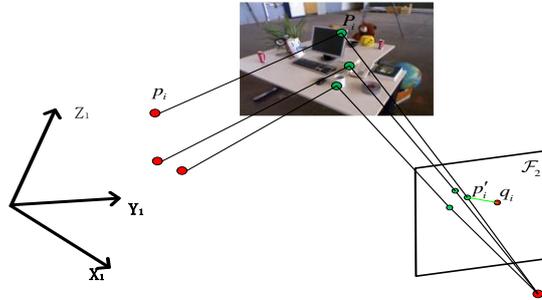


Fig.2 the procedure of reprojecting

As shown in Fig.2, the process of projecting the 3D point  $P_i$  to  $\mathcal{F}_2$  and getting the point  $p'_i$  is called re-projection [4], where  $P_i$  is corresponding to  $P_i$ .  $P_i$  and  $p'_i$  satisfy the formula:

$$p'_i = \omega(\exp_{se(3)}(\xi)P_i) \quad (4)$$

where  $\omega$  is camera projection model.

Ideally,  $p'_i$  is the same with  $q_i$ . In practice, due to the existence of matching errors, there is a re-projection error between them:

$$e_i(P_i, q_i, \xi) = \|p'_i - q_i\|_2 = \|\omega(\exp_{se(3)}(\xi)P_i) - q_i\|_2 \quad (5)$$

So the camera's pose can be obtained through *Guass-Newton* method by iteratively minimizing:

$$F = \sum_{i=1}^N e_i(P_i, q_i, \xi)^T \Omega_i e_i(P_i, q_i, \xi) \quad (6)$$

$$\xi^* = \underset{\xi}{\operatorname{argmin}} F(\xi) \quad (7)$$

where  $\Omega$  is the information matrix between  $P_i$  and  $q_i$ .

In each iteration process, an increment is computed:

$$\delta \xi^{(n)} = -\left(J^T \Omega J\right)^{-1} e^T \Omega J \quad (8)$$

where

$$J = \left. \frac{\partial e(\boldsymbol{\varepsilon} \circ \xi^{(n)})}{\partial \boldsymbol{\varepsilon}} \right|_{\boldsymbol{\varepsilon}=0} \quad (9)$$

is the Jacobin matrix of residuals  $e(\xi)$  with respect to an increment  $\boldsymbol{\varepsilon}$ ,  $J^T \Omega J$  the Gaussian-Newton approximation of the Hessian of  $F$ . The new estimate is obtained by multiplication with the computed update:

$$\xi^{(n+1)} = \delta \xi^{(n)} \circ \xi^{(n)} \quad (10)$$

where we define the symbol  $\circ$  as:

$$\xi_{ki} := \xi_{kj} \circ \xi_{ij} := \log_{SE(3)}(\exp_{se(3)}(\xi_{kj}) \cdot \exp_{se(3)}(\xi_{ij})) \quad (11)$$

Therefore, the pose of current frame  $\xi$  can be obtained.

## 2.2 Backend optimization and 3D point cloud generation

We use  $g^2o$  method<sup>[5]</sup> to decrease accumulative error. The optimization object is:

$$F = \sum_{(i,j) \in C} e(\xi_i, \xi_j, z_{ij})^T \Omega_{ij} e(\xi_i, \xi_j, z_{ij}) \quad (12)$$

where  $C$  is the set of images, as a set of nodes in the  $g^2o$  method.  $z_{ij}$  is the transformation matrix between the pose  $\xi_i$  and  $\xi_j$ , as measurements in the  $g^2o$  method.  $\Omega_{ij}$  is information matrix.  $e(\xi_i, \xi_j, z_{ij})$  is the error function, as the edge with respect to  $\xi_i$  and  $\xi_j$  in the  $g^2o$  method.

After backend optimization, we can obtain more accurate camera pose estimation  $\xi$ . Then, the point cloud of the keyframes can be generated. Keyframes are the sequence of images containing as much environmental information as possible. According to the camera projection model and camera position, the coordinates of the 3D points corresponding to pixels in the keyframes can be obtained<sup>[6]</sup>. For a 2D point  $(u, v)$ , it is corresponding to 3D point  $P_c$ . The coordinate in the world coordinate system is:

$$P_w = \exp^{-1}_{se(3)}(\xi) \cdot P_c \quad (13)$$

So, 3D point cloud can be generated by doing the same operation for all pixels in keyframes.

## 3. Representation of the ground plane

We assume that the working environment of the robots is horizontal except for the obstacles.

In order to get ground point cloud, it's wrong to intercept the point cloud along some dimension of the world coordinate system. It's because that the initial position of the camera is indefinite. In order to solve this problem, a method of obtaining the ground plane equation based on RANSAC is proposed.

### 3.1 Ground point cloud screening

In order to describe the ground plane, some of the images containing rich ground information are selected as ground keyframes. The map points corresponding to the three most gradients are selected as the ground point cloud. So we can choose as much and accurate point cloud as possible to obtain the equation of ground plane, which will improves the accuracy of equation.

As shown in Fig.3, image gradient of the pixel  $(i, j)$  is:

$$G_{ij} = (x P_{i,j} - x P_{i-1,j}, y P_{i,j} - y P_{i-1,j}, z P_{i,j} - z P_{i-1,j}) \quad (14)$$

where  $x P_{i,j}, y P_{i,j}, z P_{i,j}$  indicate the coordinate of map point which is corresponding to the pixel  $(i, j)$ .

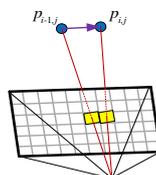


Fig.3 image gradient

### 3.2 Solving the ground plane equation based on RANSAC

After selecting the corresponding ground point cloud, the set of map points can be used to solve the ground plane equation. The ground plane equation is represented as:

$$ax + by + cz + d = 0 \quad (15)$$

where  $a, b, c, d$  is parameters that wait to be solved.

A method to solve the ground plane equation accurately is proposed. The main steps are as follows:

- 1) Selecting three sets of map points corresponding to the ground key frame randomly:  $P_1, P_2, P_3$ .
- 2) Sampling one point from  $P_1, P_2$  and  $P_3$  respectively, and then computing the plane equation where these three points are located in.
- 3) Computing all ground map points' distance to the plane obtained in 2), and then computing inliers' numbers. The distance threshold is  $D_{threshold}$ . Inliers are the map points whose distance from the ground plane is in a threshold range. As shown in Fig.4, red points are inliers, and black points are outliers.



Fig.4 schematic diagram of inlier point

- 4) After repeating 1)-3) several times, the plane equation with the most inliers is selected. On basis of inliers which are corresponding to it, the ground plane equation can be obtained by minimizing the function:

$$J(a, b, c, d) := \sum_{k=1}^{N_{in}} D_k \quad (16)$$

where  $D_k$  is the  $k_{th}$  points' distance from the plane.

Based on the above three steps, the points used in solving the ground plane equation are almost inliers, which removes influence of outliers and improves the accuracy of the ground plane equation.

## 4. Indoor navigation mapping

According to the ground plane equation obtained by the method of Section 3.2, the 3D point cloud is projected onto the ground plane along its normal. After that, by computing and updating the occupied state of each grid in the plane, the grid map can be generated. At first a 2D uniform grid map is generated. A 2D uniform grid map divides a 2D plane into uniform grids, and each grid corresponds to the possibility of existing barrier [7].

### 4.1 Method of building grid map

For a grid:

$$P(F) + P(E) = 1 \quad (17)$$

where  $P(F), P(E)$  indicate the probability with obstacles or without obstacles for a grid respectively.

Assuming that there are  $n$  points in a grid.  $n_{max}$  is the maximum of map points' number in a grid in the current detecting range, and then the occupied state of the grid is:

$$P(E) = \begin{cases} 1 & n < n_l \\ 1 - \frac{n}{n_{max}} & n_l < n < n_h \\ 0 & n > n_h \end{cases} \quad (18)$$

$$P(F) = 1 - P(E)$$

We define  $E_1, F_1$  as the last occupied state,  $E_2, F_2$  as the current occupied state. If the robot detects the same grid at next time, the occupied state of the grid is updated based on Bayesian formula:

$$\begin{cases} P(E_2|E_1) = \frac{P(E_1|E_2)P(E_2)}{P(E_1)} \\ P(E_2|F_1) = \frac{P(F_1|E_2)P(E_2)}{P(F_1)} \end{cases} \quad (19)$$

where  $P(E_1|E_2)$  and  $P(F_1|E_2)$  is the priori probability.

We set the parameter  $\lambda$ , and use the linear interpolation method to update grid occupied state:

$$\begin{cases} P(E_2) = \lambda P(E_2|E_1) + (1-\lambda)P(E_2|F_1) \\ P(F_2) = 1 - P(E_2) \end{cases} \quad (20)$$

Since the use of 0-1 grid maps can reduce computational consumption, the decision rules of grid map are required.

#### 4.2 Decision rule of grid map

0-1 grid maps divide the 2D plane into uniform grids [8-10], each grid corresponds to a value: 1 indicates that there is an obstacle in the grid, 0 indicates that there is no obstacle in the grid, and -1 indicates that the grid occupied state is unknown.

We define  $m$  as grid occupied state, and set the threshold  $m_l$  and  $m_h$ . Then, the decision rule is:

$$m = \begin{cases} 1, & P(F) > m_h \\ -1, & m_l < P(F) < m_h \\ 0, & P(F) < m_l \end{cases} \quad (21)$$

Therefore, according to the decision rule, the 2D uniform grid maps can be converted into 0-1 grid maps, which can be used conveniently.

### 5. Experiment and analysis

The simulated indoor scene in the laboratory is set as the experimental scene (Fig.6), and the scene range is  $8.0m \times 5.0m$ . The robot carries the sensor (Fig.5), and its line velocity is  $0.15m/s$ , angular velocity is  $0.15 rad/s$ , and image acquisition frequency is  $15Hz$ . The computer's configuration is Intel I7 CPU running at 2.5GHz, no GPU acceleration, and the system is Ubuntu14.04.

#### 5.1 Navigation mapping based on VSLAM



Fig.5 Mobile robot



(a) observing simulated indoor scene from the left side



(b) observing simulated indoor scene from the right side

Fig.6 graph of simulated indoor scene

Based on VSLAM<sup>[11]</sup>, we generate 3D map. The result of 3D map building is shown in Fig.7. At this experiment,  $P(E_1|E_2)$  is 0.90,  $P(F_1|E_2)$  is 0.10 and  $\lambda$  is 0.50. In order to test whether the map built by two different methods can describe environmental information fully, a suspended box and a low box are set in the scene, as shown in Fig.6.

The observation direction of Fig.6 (b) is the same with Fig.8. As shown in Fig.7, the point cloud describes the suspended box and the low box. As shown in Fig.8, based on the ground plane equation, it is very well to intercept the 3D map from 0.05m to 0.5m in the ground. It can be seen that the ground can be completely cut off, which indicates that the method of intercepting point cloud works well.



Fig.7 the point cloud map of indoor scene

Fig.8 the interception result of point cloud

Compared with Fig.10, Fig.9 describes the overall structure of obstacles in the scene fully. What's more, the suspended box and the low box are described in Fig.9. It will help the robots avoid many obstacles if the robots carry out some tasks based on the map.

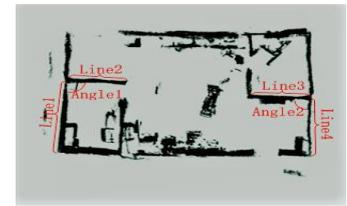
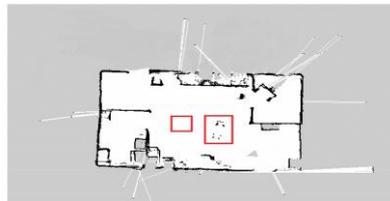


Fig.9 navigation mapping based on VSLAM

Fig.10 traditional map building based on laser

Fig.11 measurement object

**5.1 Precision analysis**

In order to evaluate the accuracy of maps, the length error and the angle error ( $l_{error}, \theta_{error}$ ) are used as the criterion. Four segments and two angles (Fig.11) are selected, and the errors are computed (Tab.1).

As shown in Tab.1, the mapping accuracy based on two different methods are similar and both high. The length errors are both within 0.06m and the angle error are both within  $1.3^\circ$ . Moreover, due to the application of global optimization, the VSLAM-based map is more accurate.

Tab.1 The table of measurement error

Measurement Object	Truth	Traditional Method		VSLAM-based Method	
		Measurement	Error	Measurement	Error
Line1	3.00m	3.06m	0.06m	3.04m	0.04m
Line 2	2.00m	2.02m	0.02m	1.98m	0.02m
Line 3	2.00m	2.06m	0.06m	1.95m	0.05m
Line 4	2.45m	2.47m	0.02m	2.48m	0.03m
Angle1	90 °	91.3 °	1.3 °	91.0 °	1.0 °
Angle 2	91 °	91.9 °	0.9 °	91.9 °	0.9 °

**5.2 Application of the map**

In order to test the practical value of the maps built by the two methods, the maps are used for the path planning experiments. It can be seen that the path in Fig.13 passes through the suspended object, which will lead to the collision with suspended object when the robots move along the path. As shown in Fig.12, the VSLAM-based method plans a path to avoid the suspended object, so that the robots can reach the target point safely.

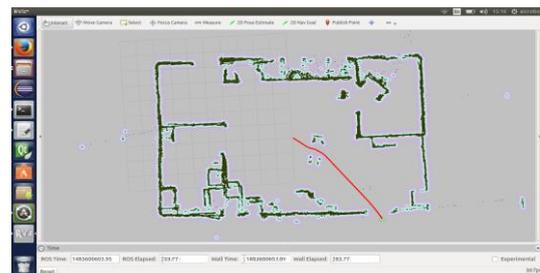
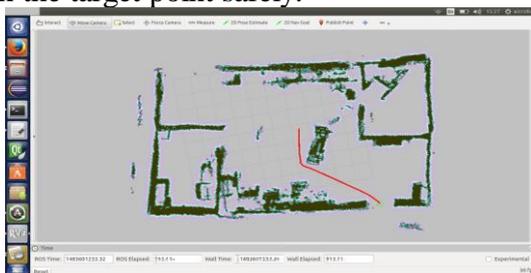


Fig.12 using the map built by VSLAM for path planning

Fig.13 using the map built by method based on laser for path planning

Experiments show that the VSLAM-based indoor navigation mapping method can describe the environmental structure more fully. Also, the precision of the map is enough high. What's more, it can improve the robustness of the subsequent application such as the path planning, navigation and so on. Compared with the traditional maps, the VSLAM-based maps have a higher practical value.

## 6. Conclusion

In this paper, a modified VSLAM-based navigation mapping method is proposed to respond the demand of the 2D maps in autonomous navigation. On the one hand, a new method to solve the ground plane equation is proposed, and it helps to realize the interception of the 3D point cloud maps effectively. On the other hand, by considering the limitations of the traditional single-plane mapping, a 2D map based on 3D point cloud map is built, so as to facilitate the follow-up tasks. Experiments show that the VSLAM-based navigation mapping method has a higher practical value than the traditional 2D mapping method.

## References

- [1] AI Q L, YU J, HU K Y, et al. Realization of SLAM based on improved O R B keyframe detection and matching for robot [J]. *Journal of Mechanical & Electrical Engineering*, 2016, 05:513-520.
- [2] Grisetti G, Stachniss C, Burgard W. Improved techniques for grid mapping with rao-blackwellized particle filters [J]. *IEEE Transactions on Robotics*, 2007, 23(1): 34-46.
- [3] Huang S, Dissanayake G. Convergence and consistency analysis for extended Kalman filter based SLAM [J]. *IEEE Transactions on Robotics*, 2007, 23(5): 1036-1049.
- [4] F Fraundorfer, D Scaramuzza, *Visual Odometry Part I: The First 30 Years and Fundamentals [C]*//*IEEE Robotics and Automation Magazine*, 2011.
- [5] XIN J, GOU J L, MA X M, et al. ZHANG Y M. A Large Viewing Angle 3-Dimensional V-SLAM Algorithm with a Kinect-based Mobile Robot System [J]. *Robot*, 2014, 05:560-568.
- [6] FU M Y, LV X W, LIU T, et al. Real-time SLAM Algorithm Based on RGB-D Data [J]. *Robot*, 2015, 06:683-692.
- [7] CAO H Y. *Mobile robot localization and mapping based on information fusion [D]* Beijing: Beijing University of Posts and Telecommunications, 2010:23-50.
- [8] Ribo M, Pinz A. A comparison of three uncertainty calculi for building sonar-based occupancy grids[J]. *Robotics and Autonomous Systems*, 2001, 35(3): 201-209.
- [9] Schultz A C, Adams W. Continuous localization using evidence grids[C]//*Robotics and Automation*, 1998. Proceedings. 1998 IEEE International Conference on. IEEE, 1998, 4: 2833-2839.
- [10] Thrun S. Learning occupancy grids with forward models [C] // *Intelligent Robots and Systems*, 2001. Proceedings. 2001 IEEE/RSJ International Conference on. IEEE, 2001, 3: 1676-1681.
- [11] Mur-Artal R, Montiel J M M, Tardós J D. Orb-slam: a versatile and accurate monocular slam system [J]. *IEEE Transactions on Robotics*, 2015, 31(5): 1147-1163.