

Robot Simultaneous Localization and Mapping Based on Self-Detected Waypoint

Yingmin Yi, Xiangru Hu

*Faculty of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048,
Shaanxi, China*

E-mail: yiym@xaut.edu.cn

Abstract: *The point of interest in this paper is the main content of autonomous navigation of robots. An algorithm for robot Simultaneous Localization And Mapping (SLAM) based on self-detected waypoint is introduced to realize robot's mapping in its area of interest. Robot's next step waypoint is performed using characteristics of large information in the area of interest and dense landmark, clustering the landmark in the area of interest, and guiding robot's movement with clustered central point. Robot clusters the observed area in its observation every time. It takes the clustered center based on the largest number of landmarks as the waypoint of the next step. Simulation experiment shows, that due to robot's movement toward the area of dense landmarks, the proposed method increases the number of landmarks observed by the robot and frequency of observation is increased. The proposed method enhances accuracy of robot's positioning and the robot realizes to detect its waypoint autonomously.*

Keywords: *Robot, Simultaneous Localization And Mapping (SLAM), K-means clustering, Extended Kalman Filter (EKF), waypoint.*

1. Introduction

The solution to the Simultaneous Localization And Map building (SLAM) problem is a "HolyGrail" of the autonomous vehicle research community in many respects. The ability of an autonomous vehicle, when is placed at unknown location in unknown environment, to build a map using only relative observations of the environment, and then to use this map simultaneously to navigate would indeed make such a robot "autonomous". Thus, the main advantage of SLAM is that it eliminates the need for artificial infrastructures or a priori topological knowledge of the environment.

The robot's simultaneous localization and mapping is the key step in its autonomous movement. The robot's SLAM study is categorized into passive SLAM

and autonomous directing SLAM according to robot's movement model. In passive SLAM, people controls robot's movement [1-3]. Document [1] puts forward that robot's position and direction is controlled by people's hands. In Document [2], vision sensor is installed on operator's head; Document [3] proposes an SLAM algorithm of stereoscopic vision on 6 degree of freedom, and this method adopts local subgraph to decrease calculating complexity but it is accomplished under operator's hand control on double vision sensor. These algorithms are all realized by people under certain training without any autonomy.

In autonomous detecting SLAM, robot is guided by target point and selects autonomously the best route to reach target point without running into barrier. The current study is mainly about SLAM guided by the target points [4-11]. Document [4] presents a real-time object-based SLAM system that leverages the largest object database to date; Document [5] solves the EKF-SLAM algorithm whereas the path planning problem is tackled via Q-learning; Document [6] studies the data association problem; Document [7] proposes clustering techniques for finding landmarks for realizing Loop Closure; Document [8] uses mutual independent subgraph technology to decrease the calculating complexity and increases mapping; Document [9] solves measure uncertainty; Document [10] presents a novel mechanism to initiate new views within the map building process for an EKF-based visual SLAM approach using omnidirectional images. This type of SLAM is about the fixed target point which guides simultaneous localization and mapping on the premise that the waypoint is pre-established and the robot has no autonomy on its selection.

Autonomous navigation of the robot is always movement to the targets. The targets generally have more feature points, that is, the area of interest for the robot. In order to let the robot have autonomous simultaneous localization and mapping in its area of interest, a type of robot Self-detected Waypoint SLAM (SWSLAM) is proposed based on K-means clustering. The robot clusters the observed area in each observation with the observed monotonic increasing number of landmarks, and it takes clustering centre on the largest number of landmarks as the next step waypoint. This is done with the robot's basic SLAM framework estimated on EKF-SLAM. The frequency of environment features observation can increase due to the guidance driven by the clustering center point in robot's area of interest. This method enhances accuracy of robot's positioning.

2. System model

2.1. SLAM system model

The described SLAM system is composed of robot's position and direction, and the observed coordinates on static environment landmark. The united state vector under k state is shown as

$$(1) \quad \mathbf{x}_k = [x_{vk}, y_{vk}, \theta_{rk}, x_1, y_1, \dots, x_N, y_N]^T = \begin{bmatrix} x_{vk} \\ m_k \end{bmatrix}.$$

In the formula, $x_{vk}, y_{vk}, \theta_{rk}$ are represented the robot's coordinates in two-dimensional space and direction angle, respectively. The map is static, parameter $m_k = [x_1, y_1, \dots, x_N, y_N]^T$ has no time index. The robot's movement model is rolling motion constraints (i.e., assuming zero wheel slip) [12],

$$(2) \quad \mathbf{x}_k = f_v(\mathbf{x}_{vk-1}, \mathbf{u}_k) = \begin{bmatrix} x_{vk-1} + v_k \Delta T \cos(\theta_{rk-1} + G_k) \\ y_{vk-1} + v_k \Delta T \sin(\theta_{rk-1} + G_k) \\ \theta_{rk-1} + \frac{v_k \Delta T}{B} \sin(G_k) \end{bmatrix}.$$

The time interval from $k-1$ up to k moment is ΔT , speed v_k and driving angle G_k are constant, which constitutes controlled quantity $\mathbf{u}_k = [v_k, G_k]^T$, and robot's wheel base is B .

2.2. SLAM observation model

The observed model is

$$(3) \quad \mathbf{z}_{ik} = h_i(\mathbf{x}_k) = \begin{bmatrix} \sqrt{(x_i - x_{vk})^2 + (y_i - y_{vk})^2} \\ \arctan \frac{y_i - y_{vk}}{x_i - x_{vk}} - \theta_{rk} \end{bmatrix},$$

where \mathbf{z}_{ik} is the observation vector at time k , and h_i is the model of the observation of the i -th landmark. The vector \mathbf{z}_{ik} is an observation of the landmark location p_i relative to the robot's location x_{vk} . This type of observation will be referred to as a vehicle-landmark observation or a VLM observation.

The model is not assumed to be perfect, so some sensor characteristics out of the model and noise corruptions are lumped into an observation error vector ω_{ik} . The observation error vector is again taken to be a temporally uncorrelated and zero means random sequence:

$$E[\omega_i] = 0, \\ \mathbf{E}[\omega_i(i) \cdot \omega_i(j)^T] = \begin{cases} \mathbf{R}_{ik} & \text{if } i = j = k, \\ \mathbf{0} & \text{otherwise,} \end{cases}$$

where \mathbf{R}_{ik} is the observation error covariance matrix at time k .

A linear observation model is employed often for the analysis in the paper, which is written as

$$(4) \quad \mathbf{z}_{ik} = \mathbf{H}_{p_i} \mathbf{p}_i - \mathbf{H}_v \mathbf{x}_{vk} + \omega_{ik},$$

where \mathbf{H}_{p_i} is the partitioned landmark observation model. This formulation of the observation equation makes the relative nature of the observation explicit.

2.3. Landmark model

Landmarks are fixed and conspicuous features of the environment. Landmarks may have many physical forms; corners, planes, rough surfaces, poles, natural or artificial terrain features can all be considered landmarks if they are repeatedly and reliably observed by a sensor. What constitutes a landmark exactly is driven by the

physics of the observing sensor – landmarks are conspicuous through the eyes of the observing sensor. This sensor-centric definition of a landmark means that it is not always possible to readily associate a landmark with visually perceived features.

Mathematically, landmarks are represented as a vector of parameters that are used to define the location and other properties of the landmark. This paper generally employs the simplest of all landmark models: A landmark is a stationary, point like entity in two dimensions. A point landmark is defined by two parameters specifying its position in Cartesian space with respect to some global coordinate frame. A point landmark is visible from all viewing angles. In general, more complex landmarks can be incorporated into the maps with models defined in this paper.

The relationship between the point landmark state at times $k + 1$ and k is trivial.

The landmark is stationary, so

$$(5) \quad \mathbf{m}_i(k+1) = \mathbf{m}_i(k) = \mathbf{m}_i.$$

Importantly, and in contrast to the vehicle model, there is no additive uncertainty term in the landmark model.

2.4. Map

A map consists of a set of landmarks with defined locations and properties. The set of landmarks within a map may be known or uncertain, and for the SLAM problem they are initially completely unknown.

Maps can generally be defined in two forms: Absolute and relative maps. In The absolute maps, all landmarks are registered in a single global coordinate frame. In the relative map, only the relationships between individual landmarks are described. The relationship between the two map types will then be investigated.

It is important to note that the maps discussed in this section are built only for navigation purposes. The maps need not correspond to a “human” description of the environment as it only contains sensor-centric landmarks.

3. Robot’s SLAM based on self-detected waypoint

Robot’s next step waypoint is performed using the characteristics of large information in the location of interest and dense landmarks, then clustering the landmarks in the location of interest, and guiding robot’s movement to the clustered central point. Due to the guidance driven by the clustering central point in area of interest, this method increases the number of the observed landmarks and the observed frequency of environment feature. It realizes a high degree accuracy of positioning.

3.1. Self-detected waypoint in area of interest

The robot’s selection in its area of interest is under no supervision and this problem can be solved through clustering. Due to the features of simplified calculation and accelerating convergence in K-means clustering algorithm, this paper regards

landmark concentration in the area of interest as the measurement criteria. So, the clustered central point will be chosen as the robot's next step waypoint. K-means clustering is non-supervision real-time clustering algorithm put forward by Mac Queen [13]. Steps of self-detected waypoint based on K-means clustering are as follows:

Step 1. Initialization

Take the observed landmark as object set, assign clustering class number, select randomly initial clustering center in object set, and set iteration suspended conditions.

Step 2. Performing iteration

Assign data object to the closest clustering center according to similarity to form the same class.

Step 3. Updating clustering center

Take mean vector of the same class as the new clustering center and re-assign data object.

Step 4. Repeating Step 2 and Step 3 until suspended conditions are satisfied.

Step 5. Statistics updating

Perform statistics upon landmark number in clustering, and take the clustering center of the most landmarks as the self-detected waypoint.

3.2. The robot's SLAM of self-detected waypoint based on K-means clustering

SWSLAM combines K-means algorithm and the observed process based on EKF-SLAM estimated framework. It functions as mobile robot's positioning algorithm. In positioning algorithm, SWSLAM autonomously selects partial waypoint and locates the area of interest autonomously.

SWSLAM positioning algorithm mainly includes seven steps: The robot's position and direction, observation of sensor, storage of landmark, updating and judging of landmark, predicting of measurement, matching of predicted value and real-observed value, and state updating and mapping.

Step 1. Prediction of position and direction

Predict the position and direction $\hat{x}_{k|k-1}$ of the robot and covariance matrix $P_{k|k-1}$ in k moment according to formulas (1) and (2); $\hat{x}_{k|k-1}$ is the state estimate at time k according to time $k - 1$; $P_{k|k-1}$ is the state estimate covariance at time k according to time $k - 1$.

Step 2. Real observation

Obtain the observation on environment feature i by using sensor, obtain real observation value z_{ik} and landmark number in k step.

Step 3. Observation of area of interest and selection of autonomous waypoint

If the number of landmarks $N(z_k)$ in continuous observation increases monotonously when the robot is within ΔT time, this area is regarded as an area of interest. At this moment the method described in Section 3.1 is adopted to select the waypoint autonomously.

Step 4. Measurement prediction

Predict $\hat{x}_{k|k-1}$ and observe environment landmark estimation value; $\hat{\mathbf{m}}_{k-1}$, $\hat{\mathbf{m}}_{k-1}$ is environment landmark estimation at time $k-1$. Calculate and predict the observed value \hat{z}_{ik} according to the robot's position and direction.

Step 5. Matching of the predicted value and real observed value

Calculate innovations Inn_k , calculate innovations covariance matrix S_k , calculate Mahalanobis distance, and match the really observed value and predicted value by using nearest neighbor relevance method [14]:

$$(6) \quad \text{Inn}_k = z_{ik} - h(\hat{x}_{k|k-1}, \hat{\mathbf{m}}_{k-1}),$$

$$(7) \quad D_k = \text{Inn}_k^T S_k^{-1} \text{Inn}_k.$$

Step 6. State updating

Update corresponding covariance matrix $P_{k|k}$ according to observation defined in formulas (3) and (6) to obtain the robot's position estimation $\hat{x}_{k|k}$ during k moment. The iteration algorithm can realize the robot's position and direction, recursive prediction and updating of covariance to get position and direction estimation during each moment.

Step 7. Mapping

State expanding, the mapping is

$$(8) \quad \begin{aligned} \mathbf{x}_{ik}^{\text{new}} &= h^{-1}(\mathbf{z}_{ik}, \mathbf{x}_{vk}), \\ \mathbf{x}_{i,k+1} &= \begin{bmatrix} \mathbf{x}_{ik} \\ \mathbf{x}_{ik}^{\text{new}} \end{bmatrix}, \end{aligned}$$

\mathbf{z}_{ik} is the observation of new landmark, $\mathbf{x}_{ik}^{\text{new}}$ are the newly observed landmarks.

4. Simulation and analysis

The experimental environment is about 100×100 m area, with three landmark clustering areas set in advance. K-means algorithm will be performed if the observed landmark number is increasing monotonously within three observation periods. 14 waypoints are used to guide robot's direction. 50 times of MonteCarlo simulation are done for EKF-SLAM, FastSLAM and SWSLAM respectively.

Fig. 1 shows the experimental results by the proposed algorithms. From Fig. 1, the robot selects autonomously numbers 4, 8, and 12 waypoints because it observes its area of interest. All the landmarks in landmark dense area are observed.

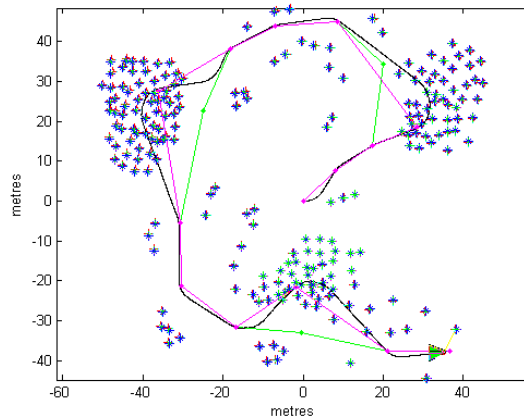


Fig. 1. SWSLAM algorithm simulation

The experiment analyzes the landmark in the robot's area of interest and its positioning accuracy.

4.1. The analysis of landmarks number in area of interest and observation frequency

The experiment names the route of the robot from the observed area of interest to the next step waypoint as one route session, which is marked respectively in Section 1, Section 2 and Section 3. The experiment adopts respectively EKF-SLAM, FastSLAM and SWSLAM to perform motion simulation. During route session, the number of controlling steps in SWSLAM is more than that of those in EKF-SLAM and FastSLAM and it also delays one session to control steps when passing every route session.

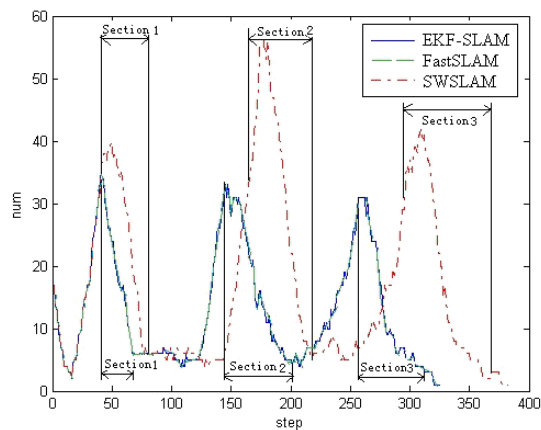


Fig. 2. Landmark number

Fig. 2 shows, that in the three route sessions, SWSLAM can acquire more landmarks than FastSLAM. From Tables 1, 2, 3, is clear that the total number of observed landmarks, the average number of one-step observed landmarks and the observed frequencies of single landmark in SWSLAM algorithm are larger than

those in EKF-SLAM and FastSLAM within three routs sessions. The experiment shows that the proposed algorithm guides the robot to observe the landmark dense area of interest. This way it increases the probability landmarks to be noticed in the area of interest.

4.2. Estimation analysis of robot's positioning accuracy and consistency

The experiment adopts Root-Mean-Square (RMS) of robot's position and direction as the judging criteria of algorithm. The experiment performs 50 times MonteCarlo simulation respectively.

Table 1 shows the comparison of the robot's position and variance average value. It is visible from Table 1 that the RMS average value of robot's position in SWSLAM is less than that in EKF-SLAM and FastSLAM. This shows that the proposed algorithm maintains a better positioning accuracy.

Table 2 compares the average value of robot's direction angle and variance. From Table 2 is clear that the RMS average value of robot's direction angle in SWSLAM is less than that in EKF-SLAM and FastSLAM. This shows that the proposed algorithm maintains a better positioning accuracy.

Table 1. Section 1

Items/Algorithms	The total number of observed landmarks	The average Number of one-step observed landmark	Observed frequency of single landmark
EKF-SLAM	564	19	14
Fast-SLAM	564	19	14
SWSLAM	1056	26	23

Table 2. Section 2

Items/Algorithms	The total number of observed landmark	The average number of one-step observed landmark	Observed frequency of single landmark
EKF-SLAM	1018	18	30
Fast-SLAM	1018	18	30
SWSLAM	1765	33	46

Table 3. Section 3

Items/Algorithms	The total number of observed landmarks	The average number of one-step observed landmark	Observed frequency of single landmark
EKF-SLAM	758	14	18
Fast-SLAM	758	14	18
SWSLAM	1365	19	28

Table 4. Position error

Items/Algorithms	Mean, rad	Covariance, rad ²
EKF-SLAM	0.3539	0.2537
FastSLAM	0.5611	0.2733
SWSLAM	0.2019	0.2286

Error analysis of robot's position, direction and direction angle shows that SWSLAM maintains the more accurate positioning than EKF-SLAM and FastSLAM.

Table 5. Direction angle error

Items/Algorithms	Mean, rad	Covariance, rad ²
EKF-SLAM	0.0081	0.0054
FastSLAM	0.0077	0.0052
SWSLAM	0.0061	0.0052

The estimation consistency is considered in the proposed algorithm. NEES (normalised estimation error squared) can be used to evaluate filtering performance index [15] to linear Gaussian filtering, i.e., the robot's position and direction is a three dimensional vector. The 50 times of Monte Carlo simulation probability is 95% and confidence interval is [2.36, 3.72]. An optimistic estimation could be greater than the upper limit of the interval; the conserved estimation is less than the lower limit.

The three algorithms are used to perform Monte Carlo simulation on robot for 50 times. The NEES mean value curving line of the robot's position and direction is shown on Fig. 3.

Fig. 3 shows that FastSLAM is conserved estimation, NEES mean value of robot's position and direction obeys χ^2 distribution. The curving line is around 4, between [2.36, 3.72], so the algorithm can be regarded as consistency estimation.

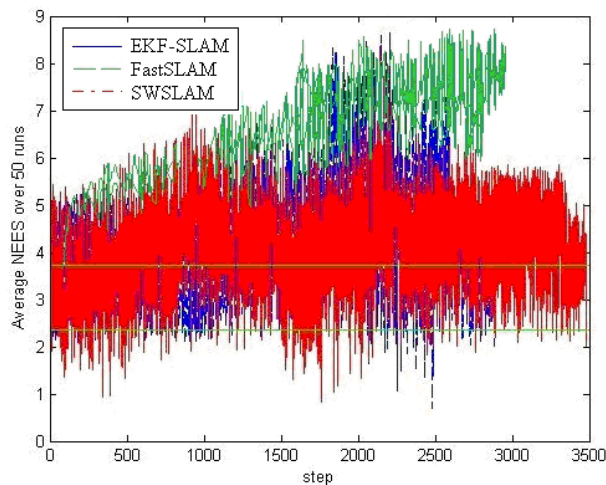


Fig. 3. Consistency of the SLAM: EKF-SLAM (blue); FastSLAM (green); SWSLAM (red)

5. Conclusions

An algorithm for robot SLAM on self-detected waypoint based on K-mean value clustering is introduced to realize robot's simultaneous localization and mapping in its area of interest. Through clustering landmarks in the area of interest, the clustering center is regarded as next step waypoint to direct robot's movement. The

presented method can increase the number of the observed landmarks by the robot, because it keeps moving towards landmark dense area. The frequency of observation also increases, thus realizing a high degree of accuracy in robot's positioning.

Acknowledgments: This work is supported by the National Natural Science Foundation of China (No 51275405) and the Natural Science Basic Research Plan in Shaanxi Province of China (No 2015JM6285).

References

1. Davison, A. J., I. D. Reid, N. D. Molton, O. Stasse. MonoSLAM: Real-Time Single Camera SLAM. – IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. **29**, 2007, No 6, pp. 1052-1067.
2. Clemente, L., A. Davison, I. Reid, J. Neira, J. D. Tardós. Mapping Large Loops with a Single Hand-Held Camera. – In: Robotics Science and Systems. Atlanta, Georgia, USA, MIT Press, 2008.
3. Paz, L. M., P. Pinies, J. D. Tardos, J. Neira. Large-Scale 6-DOF SLAM with Stereo-in-Hand. – IEEE Transactions on Robotics, Vol. **24**, 2008, No 5, pp. 946-957.
4. Gálvez-López, D., M. Salas, J. D. Tardós, J. M. M. Montiel. Real-Time Monocular Object SLAM. – Robotics and Autonomous Systems, Vol. **75**, 2016, pp. 435-449.
5. Wen, S., X. Chen, C. Ma, H. K. Lam, S. Hua. The Q-Learning Obstacle Avoidance Algorithm Based on EKF-SLAM for NAO Autonomous Walking under Unknown Environments. – Robotics and Autonomous Systems, Vol. **72**, 2015, pp. 29-36.
6. Valiente, D., A. Gil, L. Fernández, Ó. Reinoso. A Comparison of EKF and SGD Applied to a View-Based SLAM Approach with Omnidirectional Images. – Robotics and Autonomous Systems, Vol. **62**, 2014, pp. 108-119.
7. Ravankar, A. A., Y. Kobayashi, T. Emaru. Clustering Based Loop Closure Technique for 2D Robot Mapping Based on EKF-SLAM. – In: Modelling Symposium (AMS), July 2013, pp. 23-25.
8. Pinies, P., J. D. Tardos. Large-Scale SLAM Building Conditionally Independent Local Maps: Application to Monocular Vision. – IEEE Transactions on Robotics, Vol. **24**, 2008, No 2, pp. 1094-1106.
9. Clemens, J., T. Reineking, T. Kluth. An Evidential Approach to SLAM, Path Planning, and Active Exploration. – International Journal of Approximate Reasoning, Available Online, 11 February 2016.
10. Valiente, D., M. G. Jadidi, J. V. Miró, A. Gil, O. Reinoso. Information-Based View Initialization in Visual SLAM with a Single Omnidirectional Camera. – Robotics and Autonomous Systems, Vol. **72**, 2015, pp. 93-104.
11. Frintrop, S., P. Jensfelt. Attentional Landmarks and Active Gaze Control for Visual SLAM. – IEEE Transactions on Robotics, Vol. **24**, 2008, No 5, pp. 1054-1065.
12. Smith, R., P. M. Chesselman. Estimating Uncertain Spatial Relationships in Robotics. – Uncertainty in Artificial Intelligence, Vol. **2**, 1998, pp. 435-461.
13. MacQueen, J. B. Some Methods for Classification and Analysis of Multivariate Observations. – In: Proc. of 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 1967, pp. 281-297.
14. Singer, R. A., R. G. Sea. A New Filter for Optimal Tracking in Dense Multi-Target Environment. – In: Proc. of 9th Allerton Conference Circuit and System Theory, Urbana-Champaign, USA, Univ. of Illinois Press, 1971, pp. 1-21.
15. Bar-Shalom, Y., X. R. Li, T. Kirubarajan. Estimation with Applications to Tracking and Navigation. John Wiley and Sons, 2001, pp. 234-235.