

# Improved Approach to Global Localization based on Odometry Error Modeling for Mobile Robots

Jingdong Yang<sup>1</sup> and Jinghui Yang<sup>2,\*</sup>

<sup>1</sup> School of Optical-Electrical and Computer Engineering, Shanghai Key Lab of Modern Optical System, University of Shanghai for Science and Technology, Shanghai, 200093, China

<sup>2</sup> School of Business Management, Shanghai Second Polytechnic University, 201209 Shanghai, China

Received: 18 Mar. 2013, Revised: 19 Jul. 2013, Accepted: 21 Jul. 2013

Published online: 1 Mar. 2014

**Abstract:** It is often the odometry accumulative error without bound after long-range movement that decreases the precision of global localization for wheeled mobile robots. Therefore, an efficient approach to odometry error modeling is proposed regarding gentle drive type mobile robots. The approximate functional expressions, between process input of odometry and non-systematic error as well as systematic error, are derived based on odometry error propagation law. Further, the odometry error model is applied to the global localization to compensate the accumulative error during long-time navigation. In addition, Because a lot of candidate poses of robots are generated in the process of monocular visual localization, particle swarm optimization is applied to acquire the optimal pose for mobile robots during global localization. The experiments denote that in spite of sacrificing a little computation time, the proposed method decreases odometry accumulative errors, and improves the global localization precision during autonomous navigation efficiently.

**Keywords:** Scale invariant feature transform, Global localization, Odometry error modeling, Non-systematic error covariance, Simultaneous localization and mapping, 3D reconstruction, Particle swarm optimization

## 1 Introduction

Global localization denotes that the robot is placed at some location in a certain environment, after the environment map has been built. The robot extracts the features of environment images and matches them with 3D landmarks in the database, further calculates the current pose in the world coordination. However, in the beginning of global localization, it is the odometer sensor that is used to obtain the global pose of robots, the larger accumulative error often occurs after long-time navigation, even gives rise to false localization [1]. The vision measurement cannot help odometer increase global localization precision fundamentally. Therefore, an efficient odometry error modelling should be built to improve the localization precision. In addition, a candidate pose of robot will be produced if 8 SIFT features matches with the landmarks in the database. So more candidate poses will be produced when the features matching with the landmarks exceed 8. The localization problems are transferred to solve optimization problem to

get the higher localization accuracy. In the current study, there are lots of solutions for global localization. The gradient descent is linear convergence. Convergence is very slow while approaching the optimal solution[2]. it is difficult to acquire the optimal pose of robots during real-time navigation. The BP neural network looks for the minimum of the error energy function in weight space using the method of gradient descent[3]. The error energy function shifts along the smallest gradient direction. Once the error energy function gets in local minimum in optimal searching, BP method cannot get out of local minimum autonomously and arrive at the global minimum. Genetic algorithm is a novel optimization method with the characteristic of global optimization because it regulates population in a probability way[4]. But local optimal search ability is insufficient, and the premature phenomena often happen in the beginning of iterative process. The particle filter(PF) is often used to optimize the global pose of robots[5]. However, it needs lots of particles to calculate the posterior probability so that to improve the algorithm complexity. In addition, the

\* Corresponding author e-mail: [jhyang@sspu.cn](mailto:jhyang@sspu.cn)

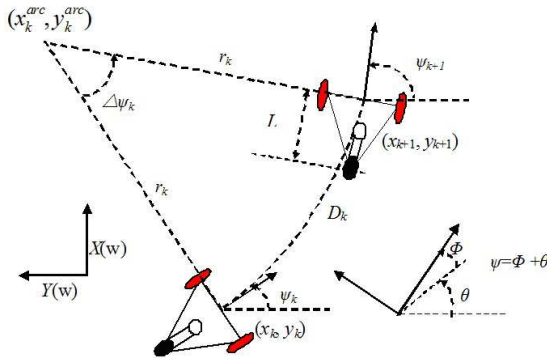


Fig. 1: The localization and navigation for ERSR robot

resampling process often decreases the efficiency and diversity of samples. So the particle degeneracy and sample impoverishment are main problems that PF method is used to acquire pose of robots. PSO (Particle swarm optimization) is an efficient optimization method with the characteristic of global optimization[6]. It can get out of lots of local extremums and acquire the global optimum, and is applicable to the optimal problem that is provided with high-dimensional variable, precision appropriate. Therefore, PSO algorithm is often used to acquire the most effective pose of robots to achieve the requirement of real-time. In this paper, an efficient odometer error model is built with respect to odometry error transform rules during autonomous navigation firstly. The accumulative errors are obtained and mapped to the input end of control system in real-time so that the localization precision is greatly improved during long-time navigation. In addition, the vision measurement will be used to help odometer locate robot itself. The SIFT features of images are extracted and matched with landmarks in the database to improve the odometry localization, when the odometry accumulative error drift occurs. Furthermore, PSO algorithm is used to optimize the candidates of robot poses to improve the localization accuracy.

## 2 Odometry Model

To construct a platform with high performing odometry, it is important to carefully think about the kinematics of robots. There are different odometry kinematics model for different drive types of mobile robots. In the following, a new odometry model is proposed for differential drive robots and synchronous drive mobile robots. It is assumed that the motion path is circular arcs. The path from the pose  $s_k$  to the pose  $s_{k+1}$  can be considered as a section of arc with radius  $r$ . In the world coordination, the coordinate of arc center is  $(x_k^{arc}, y_k^{arc})$ , and the pose of robot is  $(x(w), y(w), \psi)$ . In robot coordinates, the robot

pose is  $(x^R, y^R, \theta)$ ,  $\psi = \theta + \phi$ , where  $\psi$  is the arc angle of arc motion,  $\theta$  is the orientation angle in robot coordinates, and the steer angle,  $\phi$  is the intersection angle between the front wheel and center axis, which is used to control robot turn in the motion. Introduce  $\Delta\psi$  as the heading change of motion from one step to the next.

$$\psi_{k+1} = \psi_k + \Delta\psi_k$$

According to kinematic principle, the relations between the any two poses  $(x_k, y_k)$  and  $(x_{k+1}, y_{k+1})$  can be described as the followings:

$$\begin{cases} x_k = x_k^{arc} + r\cos(\psi_k - \pi/2) = x_k^{arc} + r\sin\psi_k \\ y_k = y_k^{arc} + r\sin(\psi_k - \pi/2) = y_k^{arc} - r\cos\psi_k \end{cases}$$

and

$$\begin{cases} x_{k+1} = x_k^{arc} + r\cos(\psi_{k+1} - \pi/2) = x_k^{arc} + r\sin\psi_{k+1} \\ y_{k+1} = y_k^{arc} + r\sin(\psi_{k+1} - \pi/2) = y_k^{arc} - r\cos\psi_{k+1} \end{cases}$$

Let  $u_k = (D_k, \Delta\psi_k)$  be the input to odometry model, where  $D_k$  is the distance traveled along the curve and  $\Delta\psi_k$  is the change in motion direction. Supposing when the rotation direction is clockwise,  $r_k$  is negative. Otherwise,  $r_k$  is positive.

$$r_k = \frac{D_k}{\Delta\psi_k} \tag{1}$$

So the robot pose for time  $k+1, s_{k+1}$ , can be presented as some nonlinear function of the pose for time  $k, s_k$ , the process input,  $u_k$ , and the noise,  $w_k$

$$s_{k+1} = f(s_k, u_k, w_k, \Delta\psi_k) = \begin{pmatrix} x_k + \frac{D_k}{\psi_k} (\sin(\psi_k + \Delta\psi_k) - \sin\psi_k) \\ y_k - \frac{D_k}{\psi_k} (\cos(\psi_k + \Delta\psi_k) - \cos\psi_k) \\ \psi_k + \Delta\psi_k \end{pmatrix}$$

### 2.1 Odometry Non-systematic Error Modeling

The non-systematic error refers to the uncertain error and is often characterized by the error mean and the error covariance. In most cases, the error mean is zero after calibration and is often ignored. Therefore, the research on non-systematic error co-variance has become the main aspect of research on non-systematic error.

**Definition 1.** The partial differential of function  $f(s_k, u_k)$  is defined as follows

$$\Delta s_k f = \Delta s_k, \Delta u_k f = \Delta u_k$$

**Definition 2.** In the context,  $c$  and  $s$  have been used to denote cosine and sine in the following equation respectively.

**Definition 3.** The covariance matrix of non-systematic error,  $Q_k$ , can be made up of the sum of  $Q_k^i$ , when  $N \rightarrow \infty$ .

**Definition 4.** The covariance matrix for the odometry input  $u_k, \Sigma_k$  can be assumed to be diagonal matrix

$$\Sigma_k = \begin{pmatrix} \delta_{D_k}^2 & 0 \\ 0 & \delta_{\Delta\psi_k}^2 \end{pmatrix}$$

where  $\delta_{D_k}^2$  is the variance in distance traveled,  $\delta_{\Delta\psi_k}^2$  is the variance in motion direction traveled. If the distance traveled is associated with motion direction in the odometry model, many parameters are introduced and the modeling efficiency is degraded. It is even difficult to reflect the actual motion for mobile robots. The error caused by the correlation between the parameters is not discussed here. The variance in distance traveled,  $\delta_{D_k}^2$  is assumed to only depend on the distance traveled,  $D_k$ . The variance in change in motion direction,  $\delta_{\Delta\psi_k}^2$  depends on both distance traveled,  $D_k$  and the change in motion direction,  $\Delta\psi_k$ , i.e.

$$\delta_{D_k}^2 = k_D |D_k| = k_D |r_k \Delta\psi_k| \quad (2)$$

$$\delta_{\Delta\psi_k}^2 = (k_\psi^\Psi + k_\psi^D |r_k|) |\Delta\psi_k| = k_\psi |\Delta\psi_k| \quad (3)$$

where  $k_D, k_\psi, k_\psi^\Psi, k_\psi^D$  are constants, referring to uncertain parameters between the ground and the wheels. Pose estimate,  $s_{k+1}$ , is a Markov procedure, which has only a connection with the pose estimate for time  $k$ , and has no relations with the pose estimate for other time. To deduce the error covariance of the pose,  $s_{k+1}$ , it is assumed that first order Taylor's expansion does not introduce significant higher order errors. And the pose error for time  $k$  is not correlated with the input error of odometry, the covariance matrix of pose for time  $k+1$ ,  $P_{k+1}$  can be evaluated as follows on the basis of Definition 1 and Definition 2 [7][8].

$$\begin{aligned} P_{k+1} &= (\nabla s_k) P_k (\nabla s_k)^T + (\nabla u_k) \sum_k (\nabla u_k)^T \\ &= (\nabla s_k) P_k (\nabla s_k)^T + Q_k \end{aligned} \quad (4)$$

where the two Jacobian matrixes above equation are given as follows according to Equation (1)

$$\nabla s_k = \begin{pmatrix} 1 & 0 & c\psi_{k+1} - c\psi_k \\ 0 & 1 & s\psi_{k+1} - s\psi_k \\ 0 & 0 & 1 \end{pmatrix} \quad (5)$$

$$\nabla u_k = \begin{pmatrix} \frac{1}{\Delta\psi_k} (s\psi_{k+1} - s\psi_k) & \frac{-r_k}{\Delta\psi_k} (s\psi_{k+1} - s\psi_k) + r_k c\psi_{k+1} \\ \frac{-1}{\Delta\psi_k} (c\psi_{k+1} - c\psi_k) & \frac{r_k}{\Delta\psi_k} (c\psi_{k+1} - c\psi_k) + r_k s\psi_{k+1} \\ 0 & 1 \end{pmatrix} \quad (6)$$

**Theorem 1.** Suppose that the odometry error for time  $k$  is not correlated with the error introduced by the input of odometry, the covariance matrix of odometry non-systematic error for time  $k$  can be expressed by some function of the input of odometry[9][10][11]:

Suppose that the odometry error for time  $k$  is not correlated with the error introduced by the input of odometry, the covariance matrix of odometry non-systematic error for time  $k$  can be expressed by some function of the input of odometry:

$$Q_k = g(r_k, \Delta\psi_k, \psi_k, \psi_{k+1}, D_k, k_D, k_\psi, k_\psi^\Psi, k_\psi^D)$$

**proof. (1) Circular arc motion**

Now divide the  $k$ :th step into  $N$  sub-steps,  $i = 1, \dots, N$ , such as  $s_k^0 = s_k$  and  $s_k^N = s_{k+1}$  as well as  $P_k^0 = P_k$  and  $P_k^N = P_{k+1}$ . Because of the above analysis, Equation (4) can be written as follows

$$P_{k+1} = (\nabla s_k^{N-1}) P_k^{N-1} (\nabla s_k^{N-1})^T + (\nabla u_k^{N-1}) \sum_k^{N-1} (\nabla u_k^{N-1})^T \quad (7)$$

By recursively expanding  $P_k^i$  and  $\sum_k^i$  of Equation (7), the following equation is gotten.

$$\begin{aligned} P_{k+1} &= \left( \prod_{i=0}^{N-1} \nabla s_k^i \right) P_k \left( \prod_{i=0}^{N-1} \nabla s_k^i \right)^T \\ &+ \sum_{i=0}^{N-1} \left[ \left( \prod_{j=i+1}^{N-1} \nabla s_k^j \right) (\nabla u_k^i) \sum_k^i (\nabla u_k^i)^T \left( \prod_{j=i+1}^{N-1} \nabla s_k^j \right)^T \right] \end{aligned} \quad (8)$$

The following equation, regarding to equation (5), can be obtained.

$$\left( \prod_{j=i+1}^{N-1} \nabla s_k^j \right) = \begin{pmatrix} 1 & 0 & c\psi_k^{N-1} - c\psi_k^{i+1} \\ 0 & 1 & s\psi_k^{N-1} - s\psi_k^{i+1} \\ 0 & 0 & 1 \end{pmatrix} \quad (9)$$

To analyze  $Q_k^i$ , start with the second term of (8). The path traveled between the time  $k$  and  $k+1$  is split in infinitesimal small segments, and for each small segment, When  $N \rightarrow \infty$ ,  $\Delta\psi_k^i \rightarrow 0$ , thus  $\psi_k^{N-1} \rightarrow \psi_{k+1}$ . The extended error between the step  $N$  and the step  $N-1$  also approaches to zero, while  $N \rightarrow \infty$ . Based on the following equation

$$\lim_{\epsilon \rightarrow 0} \sin(\alpha + \epsilon) - \sin(\alpha) = \epsilon \cos(\alpha) \quad (10)$$

$$\lim_{\epsilon \rightarrow 0} \cos(\alpha + \epsilon) - \cos(\alpha) = -\epsilon \sin(\alpha) \quad (11)$$

the following relations can be deduced.

$$\begin{aligned} &\lim_{N \rightarrow \infty} (\nabla u_k^i)^T \left( \prod_{j=i+1}^{N-1} \nabla s_k^j \right) \\ &= \lim_{N \rightarrow \infty} \begin{pmatrix} \frac{1}{\Delta\psi_k^i} (s\psi_k^{i+1} - s\psi_k^i) & \frac{-r_k}{\Delta\psi_k^i} (s\psi_k^{i+1} - s\psi_k^i) + r_k c\psi_k^{N-1} \\ \frac{-1}{\Delta\psi_k^i} (c\psi_k^{i+1} - c\psi_k^i) & \frac{r_k}{\Delta\psi_k^i} (c\psi_k^{i+1} - c\psi_k^i) + r_k s\psi_k^{N-1} \\ 0 & 1 \end{pmatrix}^T \\ &= \begin{pmatrix} c\psi_k^i & r_k (c\psi_{k+1} - c\psi_k^i) \\ s\psi_k^i & r_k (s\psi_{k+1} - s\psi_k^i) \\ 0 & 1 \end{pmatrix}^T \end{aligned}$$

Using Definition 4 and above equation, the matrix  $Q_k^i$  can be rewritten as follows.

$$\begin{aligned} Q_k^i &= \lim_{N \rightarrow \infty} \left( \prod_{j=i+1}^N \nabla s_k^j \right)^T (\nabla u_k^i) \sum_k^i (\nabla u_k^i)^T \left( \prod_{j=i+1}^N \nabla s_k^j \right) \\ &= \begin{pmatrix} c\psi_k^i & r_k (c\psi_{k+1} - c\psi_k^i) \\ s\psi_k^i & r_k (s\psi_{k+1} - s\psi_k^i) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \delta_{D_k}^2 & 0 \\ 0 & \delta_{\Delta\psi_k^i}^2 \end{pmatrix} \begin{pmatrix} c\psi_k^i & r_k (c\psi_{k+1} - c\psi_k^i) \\ s\psi_k^i & r_k (s\psi_{k+1} - s\psi_k^i) \\ 0 & 1 \end{pmatrix}^T \end{aligned}$$

Thus the elements of matrix  $Q_{k,11}^i$  can be expressed as follows

$$Q_{k,11}^i = \delta_{D_k}^2 c^2 \psi_k^i + \sigma_{\delta \psi_k}^2 r_k^2 (c\psi_{k+1} - c\psi_k^i)^2$$

Using Definition 3, the elements sum of matrix  $Q_k^i$  can be rewritten as Riemann integral regarding to  $\psi_k^i$ ,

$$Q_{k,11} = \lim_{N \rightarrow \infty} \sum_{i=0}^{N-1} Q_{k,11}^i$$

$$= \frac{1}{2} (k_D |r_k| + r_k^2 k_\psi) \text{sign}(\Delta \psi_k) (\Delta \psi_k + \frac{s(2\psi_{k+1}) - s(2\psi_k)}{2}) \\ + r_k^2 k_\psi \text{sign}(\Delta \psi_k) (\Delta \psi_k c^2 \psi_{k+1} - 2c\psi_{k+1}(s\psi_{k+1} - \psi_k))$$

In the same manner, the other elements of  $Q_k$  can be got as follows:

$$Q_{k,12} = Q_{k,21} = \frac{-(k_D |r_k| + r_k^2 k_\psi)}{4} \text{sign}(\Delta \psi_k) (c(2\psi_{k+1}) - c(2\psi_k))$$

$$+ r_k^2 k_\psi \text{sign}(\Delta \psi_k) (\frac{1}{2} \Delta \psi_k s(2\psi_{k+1}) + c(2\psi_{k+1}) - c(\psi_k + \psi_{k+1}))$$

$$Q_{k,13} = Q_{k,31} = |r_k| k_\psi (\Delta \psi_k c \psi_{k+1} - s\psi_{k+1} + s\psi_k)$$

$$Q_{k,22} = \frac{1}{2} (k_D |r_k| + r_k^2 k_\psi) \text{sign}(\Delta \psi_k) (\Delta \psi_k - \frac{s(2\psi_{k+1}) - s(2\psi_k)}{2})$$

$$+ r_k^2 k_\psi \text{sign}(\Delta \psi_k) (\Delta \psi_k s^2 \psi_{k+1} + 2s\psi_{k+1}(c\psi_{k+1} - c\psi_k))$$

$$Q_{k,23} = Q_{k,32} = |r_k| k_\psi (\Delta \psi_k s \psi_{k+1} + c\psi_{k+1} - c\psi_k)$$

$$Q_{k,33} = k_\psi |\Delta \psi_k|$$

where  $\text{sign}(\Delta \psi_k) = \text{sign}(r_k) = |r_k|/r_k \neq 0$

## (2) Straight Line Motion

When  $r_k \rightarrow \infty, \Delta \psi_k \rightarrow 0$ , the trajectory of arc motion turns to straight line for mobile robot, a special case of arc motion. The variance in motion direction traveled satisfies the following relation  $\delta_{\Delta \psi_k}^2 = k_\psi^D |D_k| = k_\psi^D |r_k| |\Delta \psi_k|$ . Satisfying the above same conditions and  $N \rightarrow \infty$ , the followings can be gotten.

$$\psi_{k+1} = \psi_k^i + \frac{N-i}{N} \Delta \psi_k, D_k \rightarrow \frac{D_k}{N}, \Delta \psi_k \rightarrow \frac{N-i}{N} \Delta \psi_k \quad (12)$$

Therefore, given the conditions, such as (1),(2),(3),(10),(11),(12),  $D_k = r_k \Delta \psi_k, Q_{k,11}^i$  can now be rewritten by above equations.

$$\lim_{N, r_k \rightarrow \infty} Q_{k,11}^i = \frac{k_D |D_k| c^2 \psi_k}{N} + k_\psi^D |D_k| (D_k)^2 s^2 \psi_k \frac{(N-i)^2}{N^3}$$

Besides, when  $n \rightarrow \infty$ , the following derivation can be used with respect to the limit sums formula of infinitude function,

$$\sum_{x=0}^n x = \frac{n(n+1)}{2} \approx \frac{n^2}{2}$$

$$\sum_{x=0}^n x^2 = \frac{n(n+1)(2n+1)}{6} \approx \frac{n^3}{3}$$

The elements of  $Q_k$  can now be rewritten by the above equations, when  $N \rightarrow \infty$ .

$$Q_{k,11} = |D_k| (k_D c^2 \psi_k + \frac{k_\psi^D D_k^2 s^2 \psi_k}{3})$$

$$Q_{k,12} = Q_{k,21} = \frac{|D_k| s 2\psi_k}{2} |D_k| (k_D - \frac{k_\psi^D D_k^2}{3})$$

$$Q_{k,13} = Q_{k,31} = \frac{-k_\psi^D |D_k| D_k s \psi_k}{2}$$

$$Q_{k,22} = |D_k| (k_D s^2 \psi_k + \frac{k_\psi^D D_k^2 c^2 \psi_k}{3})$$

$$Q_{k,23} = Q_{k,32} = \frac{k_\psi^D |D_k| D_k c \psi_k}{2}$$

$$Q_{k,33} = k_\psi^D |D_k|$$

## (3) Rotation on some spot

When  $r_k \rightarrow 0$ , the curve motion turns into rotation on some spot. On the basis of the above derivation, the elements of  $Q_k^i$  can be rewritten as follows

$$Q_{k,11} = Q_{k,12} = Q_{k,21} = Q_{k,13} = Q_{k,31} = Q_{k,23} = Q_{k,32} = Q_{k,22} = 0$$

$$Q_{k,33} = k_\psi^D |\Delta \psi_k|$$

So the proof of Theorem1 is completed. Through the above analysis, the non-systematic error can be expressed as some function with the odometry process input and some uncertain parameters between the ground and the wheels. If mapping the uncertain non-systematic error,  $Q_k$ , into the process input of odometry,  $u_k$ . the control system obtains the compensation by adjusting the non-systematic error gain,  $K_{k1}$ , effectively.

## 2.2 Odometry Systematic Error Modeling

Systematic error refers to the inner error caused by unreasonable inherent structure or low precision of odometry. In general, the fine difference in wheel radius leads to slippage so that large odometry accumulative errors are produced in motion. However, the systematic error can be obtained online and compensated by error feedback. In most cases, the systematic error incurred from unequal radius on drive wheels is placed on much importance.

Figure2 shows an assumption that a relative slippage happens when the robot is moving.  $\phi$  means the steer angle, and the change in orientation is  $\Delta \theta$ . The wheels are mechanically linked, both in translation and rotation. If there is some slight difference between the radii of both drive wheels, it will result in slippage and there is a friction force,  $F_{fric}$ , between ground and wheels, which causes a moment of rotation,  $M_{fric}$ . So the rotational moment is given by

$$M_{fric} = k_{rigid} \Delta \theta = -L \sin \phi F_{fric} = -\mu mg L \sin \phi \propto -\sin \phi \quad (13)$$

where  $k_{rigid}$  is the rotational rigid factor and  $\mu$  is the kinetic friction factor. The friction force,  $F_{fric}$ , depends

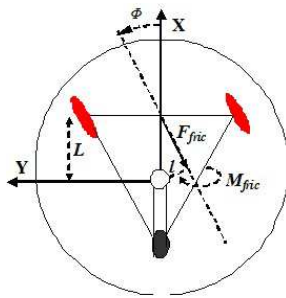


Fig. 2: The orientation change for slippage

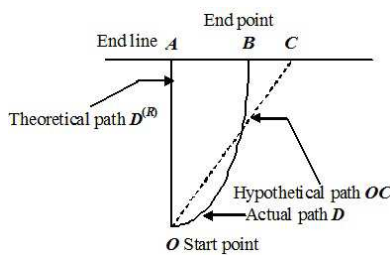


Fig. 3: The distance change for fine difference in wheel radius

on the surface properties. To compensate for the systematic error caused by friction, the friction force must be estimated online. When relative slippage occurs between wheels and the ground, the following relational expression can be gotten from Equation (13).

$$\Delta\theta \propto -\eta \sin\phi \quad \text{deg/s} (\eta < 0)$$

According to the above relations, the approximate sine relations have been inferred between the steer angle,  $\phi$  and the change in orientation,  $\Delta\theta$ . It means the orientation deviation of distance traveled per meter in steer angle direction, while the robot turns during its motion. Different parameter,  $\eta$ , depends on different mechanical structure. As Figure 3 shows, the sketch map declares the relation between the actual path,  $D$  and the theoretical path,  $D^{(R)}$  for mobile robot. It is supposed that the robot move the distance  $D^{(R)}$  along the upright direction of  $AB$  line, and arrives at point  $A$ . The slight difference between drive wheel radii drives the robot away from the straight line,  $AO$ . So the robot arrives at point  $B$  along the arc path,  $OB$ , obviously the relation,  $D > D^{(R)}$ , is satisfied. The diagonal line,  $AC$ , which starts from point,  $A$ , takes the place of the actual path,  $OB$ , approximately. Therefore, the relation between  $AO$  and  $CO$  approximates direct proportion.

$$D = \beta D^{(R)} \quad (\beta > 1)$$

IF mapping the distance deviation,  $\Delta D$ , and angle deviation,  $\Delta\theta$  into the process input of odometry,  $D > D^{(R)}$  and  $\phi$ , the control system obtains the compensation by

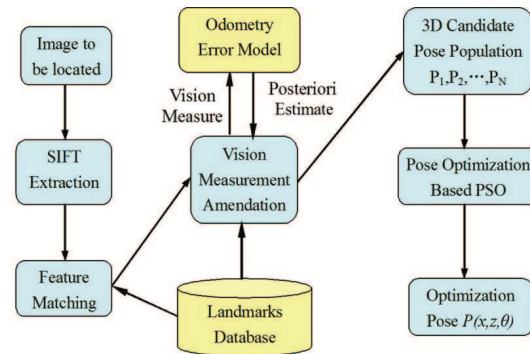


Fig. 4: The flow diagram of the global localization block

adjusting the systematic error gain,  $K_{k2}$ , effectively. Therefore, the method of systematic error feedback online reduces the inherent odometry accumulative error and improves the localization accuracy.

### 3 Global Localization Algorithm based on Particle Swarm Optimization

According to the definition of global localization, the environment map is built before global localization. With the matched 3D landmarks, the robot pose in the world coordination is calculated. The global localization process is expressed as follows:

- (1) The SIFT features of an current image is extracted in the current position,
- (2) Extract, match the features with the landmarks in the database, and obtain the sets of landmarks. Each landmark corresponds with certain images in the mapping.
- (3) The image, which is matched with most of landmarks, is the nearest frame
- (4) Calculate the robot pose with respect to the matched features between the current image and the nearest frame

The vision sensor takes pictures for the same object in different position and direction, extracts and recognizes the SIFT features to calculate the current pose in the space corresponding with the landmarks. Therefore, global localization needs to ascertain the position and orientation of robot by vision sensor. The whole global localization is given as Figure 4.

#### 3.1 Related Research

- (1) SIFT algorithm  
The features extraction and matching is the premise and basis of vision localization, object recognition and

tracking, etc, which does not only satisfy the traits of scale invariance, affine transformation invariance, but also adapts to the real-time of image processing. Lowe proposed a feature extraction method named SIFT (Scale Invariant Feature Transform) extraction [12][13][14]. The method denotes some extrema of a difference of Gaussian function are gotten from the image, which is transformed, rotated, and scaled for many times. The extrema are named as SIFT features. The SIFT features are invariant to image scale and rotation, and can provide robust matching across a substantial range of affine distortion, addition of noise, change in viewpoint, and change in illumination.

Therefore, the scale space of an image is defined as a function,  $L(x, y, \sigma)$ , which is produced from the convolution of a variable-scale Gaussian,  $G(x, y, \sigma)$ , with an input image,  $I(x, y)$ : The Gaussian kernel and its derivatives are the only possible smoothing kernels for scale space analysis.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$

where  $*$  is the convolution operation in  $x$  and  $y$ , and  $G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$ .

To efficiently detect the stable locations of matched features in scale space, SIFT method uses (Lowe, 1999) scale-space extrema in the difference-of-Gaussian function convolved with the image,  $D(x, y, \sigma)$ , which can be computed from the difference of two nearby scales separated by a constant multiplicative factor  $k$ :

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned}$$

## (2) PSO algorithm

Particle swarm optimization (PSO) [15] [16] is a stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, who were inspired by social behavior of bird flocking or fish schooling. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved so far. (The fitness value is also stored.) The "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the neighbors of the particle. This location is called  $pBest$ . When a particle takes all the population as its topological neighbors, the best value is a global best and is called  $gBest$ . The particle swarm optimization concept consists of, at each time step, changing the velocity of (accelerating) each particle toward its  $pBest$  and  $gBest$  locations (local version of PSO). Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward  $pBest$  and  $gBest$  locations.

$$\begin{aligned} v_{id}(k+1) &= w \cdot v_{id}(k) + c_1 * r_1 \bullet [P_{id}(k) - x_{id}(k)] \\ &\quad + c_2 * r_2 \bullet [g_{id}(k) - x_{id}(k)] \\ x_{id}(k+1) &= a \bullet x_{id}(k) + b \bullet v_{id}(k+1) \end{aligned}$$

where  $i = 1, 2, N, N$  is particle number,  $d = 1, 2, \dots, D, d$  is the dimension in solution space. The symbol  $\bullet$  denotes element-by-element vector multiplication. At iteration  $k$ , the velocity  $v_{id}(k)$  is updated based on its current value affected by a momentum factor  $w$  and on a term which attracts the particle towards previously found best positions: its own previous best position  $P_{id}(k)$  and globally best position in the whole swarm  $g_{id}(k)$ . The factor of acceleration is given by the coefficients  $c_1$  and  $c_2$ , which determines the longest step for  $pBest$  and  $gBest$  respectively. The particle position  $x_{id}(k)$  is updated using its current value and the newly computed velocity  $v_{id}(k+1)$ , affected by coefficient weights  $a$  and  $b$  respectively. It is shown later that  $a$  and  $b$  can be set to unity without loss of generality. Randomness useful for good state space exploration is introduced via the vectors of random numbers  $r_1$  and  $r_2$ . They are usually selected as uniform random numbers in the range  $[0, 1]$ . Generally, the relations between  $v_{idmax}$  and  $x_{idmax}$  is given as follows:  $v_{idmax} = k \bullet x_{idmax}$ ,  $0.1 \leq k \leq 0.2$ . If the current value  $v_{id}$  or  $x_{id}$  surpasses the extreme, it will be set as the new extreme in the region. When the iterations reach the largest or particles exploitation has been to a stable position, the convergence happens.  $w$ , nonnegative, is named as inertia factor. The larger  $w$  is, the faster the global convergence will be, otherwise, local convergence is more obvious. Supposing that  $w_{max}$  and  $w_{min}$  are the maximum and minimum weighted coefficient,  $run$  is the iterations, and  $runMax$  is the total iterations.

$$w = w_{max} - run * \frac{w_{max} - w_{min}}{runMax}$$

## 3.2 Pose Determination based on Monocular Vision Measurement

### (1) Position determination

To ascertain the orientation of robots in the world space at current time, match the features from the current image with the landmarks in the database, calculate the essential matrix,  $E$ , with the orientation angle,  $\theta$ , which satisfies the epipolar geometry. Further, obtain the orientation angle by trying feasible value [17][18][19].

**Theorem 2.**  $(X_i, Y_i, Z_i)$  and  $(X_j, Y_j, Z_j)$  are the projection of the 3D point,  $M(X, Y, Z)$  in the coordinates of camera  $i$  and camera  $j$ , with which the landmarks in the database  $(X_i^T, Y_i^T, Z_i^T)$  and  $(X_j^T, Y_j^T, Z_j^T)$  are matched. Supposing there are 4 matched points ( $k \geq 4$ ), essential matrix

$$E = \begin{pmatrix} 0 & e_1 & 0 \\ e_2 & 0 & e_3 \\ 0 & e_4 & 0 \end{pmatrix}, e_1, e_2, e_3, e_4 \text{ are the constants, so the}$$

current orientation angle in the world space can be expressed as follows:

$$(1) k \in (4, 8), \theta = \tan^{-1} \left( \frac{e_1 e_3 - e_2 e_4}{e_1 e_2 + e_3 e_4} \right)$$

$$(2) k \in (8, +\infty), \theta = \sum_{n=1}^{C_k^2} \left( \theta_n - \frac{1}{C_k^2} \sum_{n=1}^{C_k^2} \theta_n \right)^2$$

$$= \sum_{\substack{j=i+1 \\ i=1}}^k \tan^{-1} \frac{BC-AD}{AC+BD}$$

where

$$A = X_i^T - X_j^T, B = Z_i^T - Z_j^T, C = X_i - X_j, D = Z_i - Z_j$$

**Proof.**

(1) Supposing that  $X_1 = [X_1, Y_1, Z_1]^T$  and  $X_2 = [X_2, Y_2, Z_2]^T$  are the projection of the 3D point,  $P(X, Y, Z)$  in the coordinate of camera1 and camera2,  $I_1 = [x_1, y_1, 1]^T$  and  $I_2 = [x_2, y_2, 1]^T$  are the pixel images and matches with  $X_1$  and  $X_2$  respectively.  $I_1$  is matched with  $I_2$

$$\lambda_1 x_1 = x_2, \lambda_2 x_2 = x_2$$

If  $R$  and  $T$  express the transformation and rotation matrices in the camera coordinate 1 and coordinate 2 respectively, the following relations can be given

$$X_2 = RX_1 + T$$

According to the above equations, the follows are gotten

$$\lambda_2 x_2 = R \lambda_1 x_1 + T$$

Computing the cross product of the above equation with Matrix,  $T$

$$\lambda_2 T \times x_2 = \lambda_1 T \times R x_1$$

Computing the dot product of the above equation with Matrix,  $x_2^T$ , and eliminating  $\lambda_1/\lambda_2$  to obtain the following equation

$$x_2^T (T \times R) x_1 = 0$$

Where  $E = T'R$  is essential matrix,  $T'$  is the antisymmetric matrices of  $T$ .

Suppose that the robot transforms in the plane  $x$ - $z$ , and camera revolves on the axle  $y$ , naming  $t_y = 0$ ,  $T, R$ , and  $E$  can be obtained as follows:

$$\mathbf{T}' = \begin{pmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{pmatrix}, \mathbf{R} = \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix}$$

$$E = \begin{pmatrix} 0 & -t_z & 0 \\ t_z \cos \theta + t_x \sin \theta & 0 & t_z \sin \theta - t_x \cos \theta \\ 0 & t_x & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 & e_1 & 0 \\ e_2 & 0 & e_3 \\ 0 & e_4 & 0 \end{pmatrix}$$

when the matched points satisfies  $k \in (4, 8)$ ,  $e_1, e_2, e_3, e_4$  can be gotten by trying feasible value. Moreover, four

equations are obtained with respect to the expression of  $E$ , we eliminate  $t_x, t_z$ , and obtain the following equation

$$\theta = \tan^{-1} \left( \frac{e_1 e_3 - e_2 e_4}{e_1 e_2 + e_3 e_4} \right)$$

(2) When the tentative matches satisfies  $k \in (8, +\infty)$ , two tentative matches are selected from the matched list to calculate the alignment parameters,  $(x, z, \theta)$  randomly.

$$X = X_i - X_i' \cos \theta - Z_i' \sin \theta$$

$$Z = Z_i - Z_i' \cos \theta - X_i' \sin \theta$$

Where  $(X_i^T, Y_i^T, Z_i^T)$  is 3D landmark in the database and  $(X_i, Y_i, Z_i)$  is the 3D coordinate of the current frame in camera coordinates. With two tentative matches,  $i$  and  $j$ , the following equations are given

$$A \cos \theta + B \sin \theta = C \tag{14}$$

$$B \cos \theta - A \sin \theta = D \tag{15}$$

where

$$A = X_i^T - X_j^T, B = Z_i^T - Z_j^T, C = X_i - X_j, D = Z_i - Z_j$$

If the two tentative matches are correct, the distance between the two landmarks is invariant for this Euclidean transformation, the following constraint is applied to each sample selection:  $A^2 + B^2 \approx C^2 + D^2$ . That eliminates many samples containing wrong matches from further consideration efficiently. Solving (14) and (15), the equations are obtained.

$$\theta = \tan^{-1} \left( \frac{BC - AD}{AC + BD} \right)$$

When the tentative matches satisfies  $k > 8$ , the higher precision is gotten after taking the least-squares of  $\theta$ .

$$\theta = \sum_{n=1}^{C_k^2} \left( \theta_n - \frac{1}{C_k^2} \sum_{n=1}^{C_k^2} \theta_n \right)^2$$

$$= \sum_{\substack{j=i+1 \\ i=1}}^k \tan^{-1} \frac{BC-AD}{AC+BD}$$

(2) Orientation Determination

Supposing camera satisfies the epipolar line model, the origin of the world space is the origin of camera coordination when the robot is at the initial position. Namely, axle  $z$  is the optical axis of camera, the axis  $y$  is vertical downward, the axis,  $XYZ$  all satisfy right-hand rule. If the orientation,  $\theta$  is known, the translation in the world space can be gotten

**Theorem 3.** There is a 3D point in the space,  $P(X, Y, Z)$ , whose pixel coordinate is  $(u, v)$  at some time.  $M$  is the projection matrix,  $m_{ij}$  is the elements of  $M$ , satisfying

$i \in 1, 2, 3, j \in 1, 2, 3, 4, f_x, f_y, u_0, v_0$  is the intrinsic parameters. When the orientation,  $\theta$  is known, the translations in  $x$ - $z$  plane in the world space can be gotten.

$$\begin{pmatrix} x \\ z \end{pmatrix} = - \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} A^{-1} B$$

where

$$A = \begin{pmatrix} -f_x & u - u_0 \\ 0 & v - v_0 \end{pmatrix}$$

$$B = \begin{pmatrix} (m_{11} - um_{31})X + (m_{12} - um_{32})Y + (m_{13} - um_{33})Z \\ (m_{21} - vm_{31})X + (m_{22} - vm_{32})Y + (m_{23} - vm_{33})Z \end{pmatrix}$$

**Proof.** Supposing there is a certain 3D point  $P(X, Y, Z)$ , whose homogeneous equation is  $(X, Y, Z, 1)^T$ . And  $(u, v, 1)^T$  is the homogeneous equation of the projection image for point  $P$ . According to the projection matrix, the following relations are given.

$$\begin{aligned} z_c \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} &= M_1 M_2 \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = M \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \end{aligned} \tag{16}$$

$M$  is the projection matrix,  $m_{ij}$  is the elements of  $M$ , satisfying  $i \in \{1, 2, 3, 4\}$ ,  $M_1$  is the camera intrinsic matrix, which is related to the inner structure.  $M_2$  is the camera extrinsic matrix, which is determined by the orientation in the camera space relative to the world space. With respect to perspective projection theory and the relations between camera coordination and world coordination,  $M_1, M_2$  can be expressed as follows respectively.

$$M_1 = \begin{pmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$M_2 = \begin{pmatrix} R & t \\ 0^T & 1 \end{pmatrix} = \begin{pmatrix} \cos \theta & 0 & \sin \theta & t_x \\ 0 & 1 & 0 & 0 \\ -\sin \theta & 0 & \cos \theta & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where  $f_x, f_y, u_0, v_0$  is the known intrinsic parameter,  $R$  and  $t$  is rotation and transform matrix of camera coordination relative to the world coordination. With Equation (16), eliminate  $Z_c$  to get the following equations  
 $X(m_{11} - um_{31}) + Y(m_{12} - um_{32}) + Z(m_{13} - um_{33}) = um_{34} - m_{14}$   
 $X(m_{21} - vm_{31}) + Y(m_{22} - vm_{32}) + Z(m_{23} - vm_{33}) = vm_{34} - m_{24}$   
 Substitute the elements,  $m_{ij}$  into the above equation

$$\begin{aligned} &\begin{pmatrix} -f_x & u - u_0 \\ 0 & v - v_0 \end{pmatrix} \begin{pmatrix} t_x \\ t_z \end{pmatrix} \\ &= \begin{pmatrix} (m_{11} - um_{31})X + (m_{12} - um_{32})Y + (m_{13} - um_{33})Z \\ (m_{21} - vm_{31})X + (m_{22} - vm_{32})Y + (m_{23} - vm_{33})Z \end{pmatrix} \end{aligned}$$

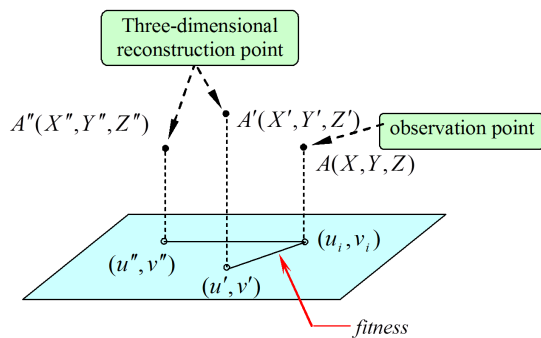


Fig. 5: The pose uncertainty for vision measurement

Supposing

$$A = \begin{pmatrix} -f_x & u - u_0 \\ 0 & v - v_0 \end{pmatrix}$$

$$B = \begin{pmatrix} (m_{11} - um_{31})X + (m_{12} - um_{32})Y + (m_{13} - um_{33})Z \\ (m_{21} - vm_{31})X + (m_{22} - vm_{32})Y + (m_{23} - vm_{33})Z \end{pmatrix}$$

Then

$$\begin{pmatrix} t_x \\ t_z \end{pmatrix} = A^{-1} B$$

The above matrix is the translation matrix in the camera space relative to the world space,  $t_x, t_z$  is the translation, after rotation first and translation later. With respect to the relations between two spaces, the translation,  $[x, z]^T$  of robot relative to the original in world coordination is given as follows:

$$\begin{pmatrix} X \\ Z \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} t_x \\ t_z \end{pmatrix} = - \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} A^{-1} B$$

For improving the stability of  $x, z$ , two or more two features are used to expand matrix  $A$  and  $B$  to obtain the solution  $x, z$ .

### 3.3 Pose uncertainty analysis

The Sensors are used to acquire the data of global localization. But the errors also occur when collecting outside data by sensors. Therefore, before the pose uncertainty is described, the pose measurement should be estimated by camera sensor. **Theorem 4.** Suppose there are  $n$  images in the space, whose positions are known relative to each other, they can be expressed by pinhole mode. There is a 3D point  $P = (x, y, z)^T$  in the space, whose pixel projection in  $i$ th image is  $m_i = (u_i, v_i)^T$ . The relation,  $s\tilde{m}_i = M_i \tilde{P}$ , is satisfied as follows:

$$\begin{aligned} u_i &= \frac{M_{11}X_i + M_{12}Y_i + M_{13}Z_i + M_{14}}{M_{31}X_i + M_{32}Y_i + M_{33}Z_i + M_{34}} \\ v_i &= \frac{M_{21}X_i + M_{22}Y_i + M_{23}Z_i + M_{24}}{M_{31}X_i + M_{32}Y_i + M_{33}Z_i + M_{34}} \end{aligned}$$



**Proof.** with the relation  $s\tilde{m}_i = M_i\tilde{P}$ , the following equation is gotten.

$$s \begin{pmatrix} u_i \\ v_i \\ 1 \end{pmatrix} = \begin{pmatrix} M_{11} & M_{12} & M_{13} & M_{14} \\ M_{21} & M_{22} & M_{23} & M_{24} \\ M_{31} & M_{32} & M_{33} & M_{34} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

After eliminating  $s$ , the following equation can be written as below

$$\begin{cases} u_i = \frac{M_{11}X_i + M_{12}Y_i + M_{13}Z_i + M_{14}}{M_{31}X_i + M_{32}Y_i + M_{33}Z_i + M_{34}} \\ v_i = \frac{M_{21}X_i + M_{22}Y_i + M_{23}Z_i + M_{24}}{M_{31}X_i + M_{32}Y_i + M_{33}Z_i + M_{34}} \end{cases}$$

**Definition 5.** there are some 3D points  $(X_1, Y_1, Z_1), (X_2, Y_2, Z_2), \dots, (X_n, Y_n, Z_n)$  when observing from a image.  $(u_i, v_i)$  is the pixel coordinates for the 3D points.  $M_{ii}$ ,  $i = 1, 2, n$  is the camera projection matrix,  $d_i$  is the distance between the current position and the object. The uncertainty of robot pose is represented as [20]:

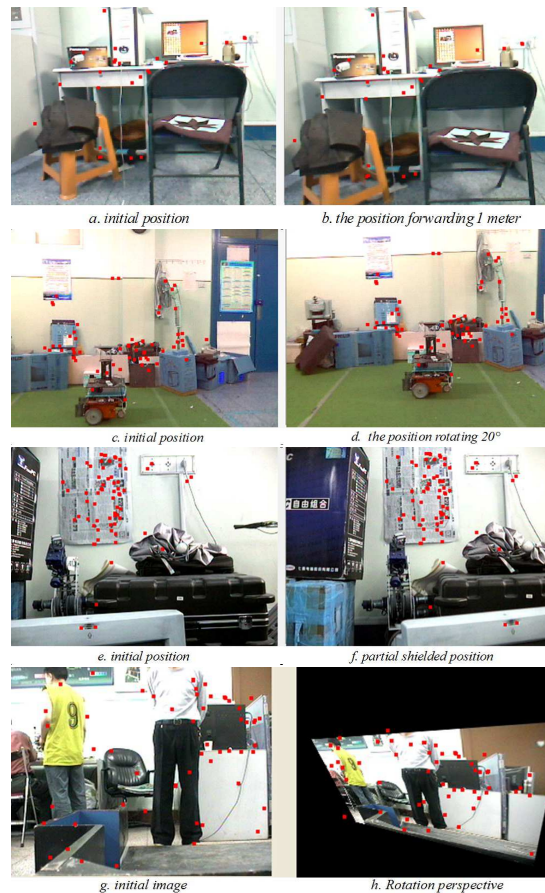
$$fitness = \frac{1}{n} \sum_{i=1}^n \left[ \left( u_i - \frac{M_{11}X_i + M_{12}Y_i + M_{13}Z_i + M_{14}}{M_{31}X_i + M_{32}Y_i + M_{33}Z_i + M_{34}} \right)^2 + \left( v_i - \frac{M_{21}X_i + M_{22}Y_i + M_{23}Z_i + M_{24}}{M_{31}X_i + M_{32}Y_i + M_{33}Z_i + M_{34}} \right)^2 \right]^{1/2}$$

The above function can be used to express the projection matrix,  $M$ , when  $(u_i, v_i)$  and  $(X_i, Y_i, Z_i)$  is all known. Namely, the uncertainty of robot pose can be expressed as the above equation when the robot can observe the  $i$ th landmarks. In addition, the function is often used as the level of localization precision, and is considered as the fitness of pose optimization for mobile robot as well.

## 4 Experiment and Analysis

### 4.1 SIFT Extraction and Matching for Monocular Vision

The visual localization can help revise odometry localization accuracy in real-time and improves stability of control system. So SIFT extraction and matching has influence on robots localization. In this paper, four classical experiments of SIFT algorithm are conducted. As is shown in Figure6(a,b), SIFT extraction and matching is shown for the two images at a distance of 1 meters. The matched features number is 32, and the time is 160ms. In the same manner, Figure6(c → h) shows the other features of SIFT features such as affine invariance, transform invariance, rotation perspective invariance. The matching features number is 42,40,51. The time is 190ms,180ms, 212ms respectively. The robust traits of SIFT features can ensure quick and accurate extraction and matching of SIFT features with landmarks in the map



**Fig. 6:** Typical characteristic of SIFT features

database for mobile robot during autonomous navigation, and then, gets the vision measurement of robot pose in the world space. The vision measurement accuracy can reach to more than 76% when the position error in XZ direction is smaller than 50cm. The accuracy can help robot shake off kidnapping quickly, and ensure the robustness of navigation.

### 4.2 Odometry Error Modeling

In order to identify the odometry error modeling in real-time is applicable and efficient, some localization experiments have been done for mobile robots in indoor environments.

As is shown in Figure 8(a), the typical robot is the differential drive mobile robot, Pioneer3-DX. This robot is equipped with a web camera, mounted on the top, and an array of IR sensors and bump sensors, which are used to measure the distance of obstacle and build occupancy grid map. As the experiment scene of Figure7 shows, the robot is driven to navigate autonomously in the room using a joystick. After building the initial environment map, it



Fig. 7: Global localization Scene

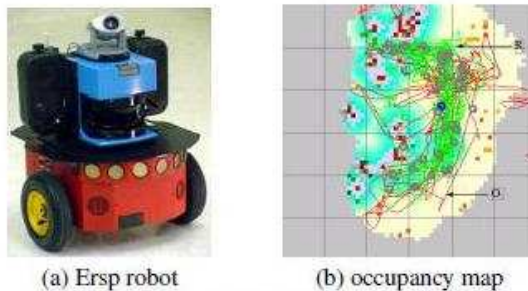


Fig. 8: Global localization and Navigation.

begins exploiting around the area according to the built map for about 20min. Figure 8(b) gives the true occupancy map related to the environment in Figure 7. The path of odometry, the red line, is not the same as the robot path, the green line. It is because that the non-systematic error results in the increasing of accumulative errors, and then gives rise to the odometry error drift.

### (1) Odometry Non-systematic Error Modeling

Figure 9 gives the comparison data of pose error, before and after, odometry non-systematic error modeling. The pose error is smaller at the beginning of navigation. With the increase of movement distance, the pose error of measurement is growing. When the robot moves to the group of obstacles near *Wall a* at the time of No.800 second, the collision and sideslip appears. Also the kidnapping happens. The pose error of robot is growing more significant. At this time, the pose error has made a great leap forward. The global localization is conducted by its own vision system, taking the place of odometer. The vision localization can help to reduce odometry accumulative error efficiently, which is caused by navigation for a long time. The landmark and environmental map is built at the same time. After simultaneous localization and mapping for about 100

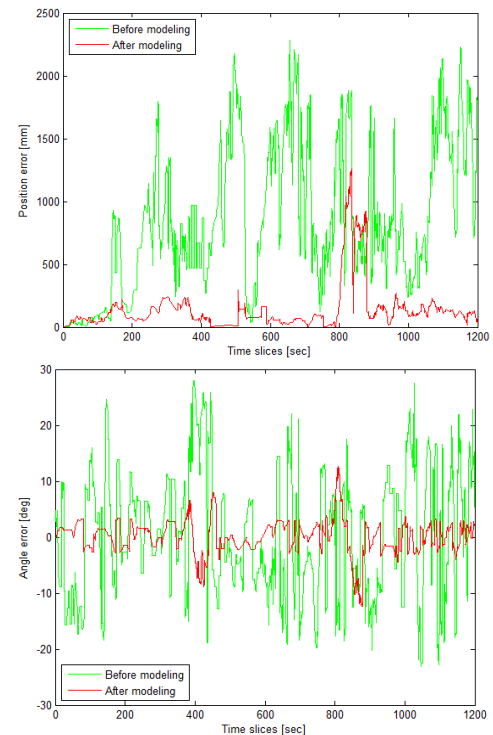


Fig. 9: The position error and angle error, before and after, odometry non-systematic error modeling

seconds, the kidnapping is over. The robot reuses odometer for localization instead of vision, and enters into autonomous navigation. At this time, the pose error is growing smaller and fluctuates steadily. It has shown both the accuracy of position error and angle error are all improved remarkably from Figure 9. The position error mean is 139.93mm, reduced by 83.52%, and the angle error mean is 2.25, reduced by 76.86%. Therefore, the odometry non-systematic error feedback in real-time can reduce the localization error efficiently and improve the accuracy of localization during autonomous navigation for mobile robots. Therefore, it proved that odometry non-systematic error can be obtained in real-time and compensated online at the feedback control end efficiently.

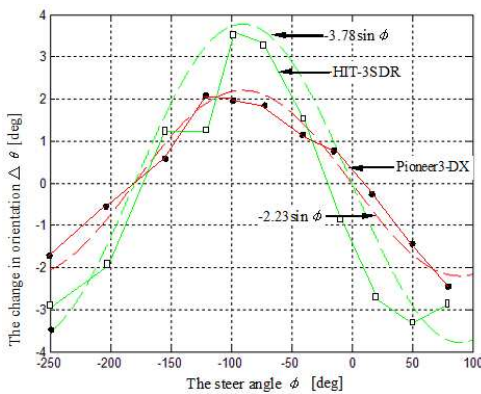
### (2) Odometry Systematic Error Modeling

Table 1 shows the data that the robot is driven forward about 10 meters for differential drive robot, Pioneer3-DX.  $\theta_{Start}$  is the initial angle of movement.

And  $\phi$  and  $\Delta\theta$  are the steer angle and angle error. Then the true distance,  $D$ , and the measurement value,  $D^{(R)}$ , are used to obtain the scaling factor,  $\beta = D/D^{(R)}$ . The mean of scaling factor is  $\beta \approx 1.0013$ . And the data in the same environment for synchronous drive robot, HIT-3SDR, satisfies  $\beta \approx 1.0027$ . Therefore, the distance

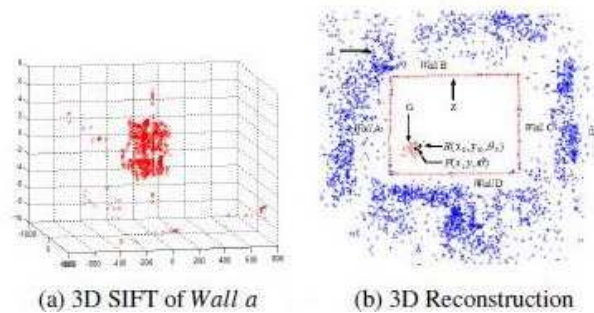
**Table 1:** The error of distance and angle with different starting angle along the same direction

$\theta$	$D$	$D^{(R)}$	$D/D^{(R)}$	$\phi$	$\Delta\theta$
/deg	/mm	/mm	/%	/deg	/deg
7.9	9825.4	9815.6	1.000998	82.1	-2.3
40.5	9887.6	9874.3	1.001467	49.5	-1.4
78.8	9858.4	9836.2	1.001347	14.2	-0.3
103.4	9734.6	9726.9	1.002257	-18.4	0.8
133.0	9785.5	9773.7	1.000792	-43.0	1.2
163.3	9778.9	9767.8	1.001207	-73.3	1.8
192.3	9956.5	9946.8	1.001136	-102.3	1.9
212.6	9946.9	9931.5	1.000975	-122.6	2.1
249.7	9978.9	9968.3	1.001551	-159.7	0.8
305.5	9875.4	9856.7	1.001063	-215.5	-0.5
340.5	9830.1	9815.7	1.001897	-250.5	-1.8



**Fig. 10:** The relations between  $\Delta\theta$  and  $\phi$

deviation caused by tiny differentia of radii exists in both synchronous drive and differential drive mobile robots generally. The systematic error has certain correction with the process input of odometry, and can be compensated at the feedback control end. As also seen in Figure10, there is the sampling curve and fitting curve in the same environment for different driving type mobile robots. The deep continuous line and discontinuous line present the sampling curve and fitting curve for Pioneer3-DX and the tint lines shows the data for HIT3-SDR. So the relation between  $\phi$  and  $\Delta\theta$  for Pioneer3-DX can be given as the expression:  $\Delta\theta \approx -2.23\sin\phi(deg/s)$ . The relation for HIT3-SDR is obtained as:  $\Delta\theta \approx -3.78\sin\phi(deg/s)$ . Therefore, the angle error, generated from the difference in wheel radius, has the approximate sine functional relations with the steer angle. The angle error,  $\Delta\theta$ , caused by tiny differentia of radius exists in both synchronous drive and differential drive mobile robots generally and can also be compensated by the change of the odometry process input,  $\phi$ , in real-time.



**Fig. 11:** Three-dimensional Reconstruction and Global localization.

### 4.3 Global Localization based on PSO Algorithm

For the scenes of Figure7, some experiments for global localization and mapping have been conducted in the indoor environment with four walls. The mobile robot moves around along a rectangle path. The PTZ camera takes pictures for the objects in front of every wall, and the SIFT features of the images are extracted. If the SIFT features will be matched with the landmarks in the database successfully, the features are represented in three-dimensional space. When the robot moves around along the four walls of the room, the 3D reconstruction of all matched SIFT features will be completed autonomously. As Figure11. shows 3D reconstruction, mapping, and global localization in the scenes as Figure7. Figure11(a) gives the 3D reconstruction features in the space in front of Wall a. The robot has known the 3D coordinates of the SIFT features before Wall a. On that basis, the global pose in the whole environment will be derived by odometer and camera. In Figure11(b)that ambient blue point denotes the 3D reconstruction points for the matched features. And the rectangle red line is the moving path for mobile robot, those scattered red points in the rectangle are the initial global pose population,  $G$ . In which the bigger black point is the optimal global pose,  $P$ , and  $R$  means the true pose of robot in the environment. In the optimizing process using PSO, the correlative parameters are given as follows:  $N = 30, D = 2, d \in [-100, 100], \theta \in [-\pi, \pi], k = 0.1, v_{idmax} = 10, w_{max} = 0.9, w_{min} = 0.4, a = 1, c_1 = c_2 = 2, R(78.73, 2.27, 0)$ , After iteration  $k=65$ , the convergent appears. The final running result is as follows:  $gBest(65)=P(12)(70.43, 75.62, 0), fitness(0.258, 0)=2.1918, w=0.872, t=125.345ms$ .

The experiment shows the translation error is smaller than 15cm, orientation error is smaller than 5. Therefore, the global localization method can meet the demand of the real-time of autonomous navigation for mobile robots. The global localization tests were completed in 40 different locations of the environment of Figure7 using PSO algorithm. Figure12 gives the data by using three

algorithms, such as before and after PSO algorithm, PF algorithm. The orientation angle is zero for the former 20 times. Namely, the orientation of robot movement is parallel to the map-building. The orientation angle grows increasingly in the later 20 times. Figure12(a) shows the translation errors for three localization algorithm. The translation error of PSOGL is less than 12cm in the former 20 times, the error is larger in the later 20 times. And the translation error increased with the increase of the orientation angle. But the largest error is less than 20cm. However, the translation error of GL is far more than PSOGL algorithm. The translation error of PF is less than GL, and more than PSOGL. Because the particle degeneracy and sample impoverishment decreases the localization accuracy. The average translation error of PSOGL decreased by 38.6% than GL, less than PF by 20.12%. Figure12(b) shows the orientation error for three global localization algorithms. From the data, some conclusion is known clearly. Similar to the translation error, the orientation error for PSOGL is less than 5 in the former 20 times, the average of which is less than 4. In the later 20 times, the error grows with the increase of orientation angle. But the orientation error is less than 15 in the later 20 times, most of which is less than 10. The statistics data shows the orientation error of PSOGL decreased by 57.21% than GL, and decreased by 26.34% than PF. The above data shows the localization accuracy is higher in the former 20 times because it is parallel for the moving direction of robot and the orientation of map-building. The localization error of two algorithms grows with the increase of orientation angle. It is because the matched features between the images to be located and landmarks decreased with the increase of direction deviation. Factors like the growth of the wrong ratio of features matching may decrease the localization precision. However, the localization error (translation error and orientation error) for PSOGL is lower than GL and PF.

#### 4.4 Global Localization Time Analysis

The real-time of global localization should be considered except for the localization accuracy too. Figure13 gives the localization time for three global localization algorithms. PSOGL algorithm needs more time than GL algorithm, and less than PF algorithm. It is because that PSOGL algorithm needs some time to eliminate the odometry accumulative error, matches with landmarks, and computes the fitness to get best pose by PSO algorithm. So the average localization time increased by 12.48% than GL algorithm. Compared with PF algorithm, PSOGL keeps the global searching strategy for the whole population, and avoids the intricate individual operation. The algorithm takes simple speed model, and uses special memory to search the current best solution dynamically. However, PF algorithm needs more time to update its particle weight with respect to the small amount of

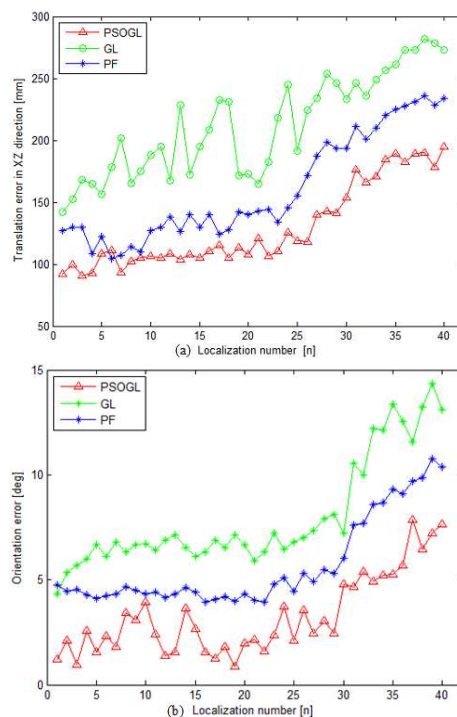


Fig. 12: Translation error and orientation error

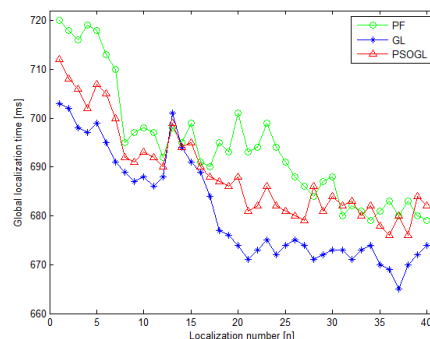


Fig. 13: The global localization time

particles, 30. Therefore, the whole localization time for PSOGL is less than PF by 32.29%, but the localization accuracy increased efficiently.

In the localization experiments, the average orientation error for PSOGL decreased by 57.21% than GL, and by 26.34% than PF. The translation error for PSOGL decreased by 38.62% than GL, and by 20.12% than PF. Though the whole localization process sacrifices a little time cost, which increased by 12.48%, a higher localization precision is achieved. The translation error and the orientation error decreased by 38.62% and 57.21% respectively. In terms of the return on investment

in economics, it is quite necessary to adopt PSOGL algorithm with odometry error modeling.

## 5 Conclusion

In this paper, a general approach for odometry error modeling is proposed by the existing questions of global localization for mobile robots. The modelling is used online to reduce non-systematic error and systematic error of odometry during long-range motion. In addition, a calculation method of robot pose in the world space is proposed and proved for monocular visual localization. Further, the optimal pose is gotten from lots of candidate poses that match with landmarks successfully by using PSO algorithm. The experiments denote that the improved algorithm improves the pose precision greatly. If the physics constraints of robot body are considered, the algorithm with odometry error feedback in real-time can be applied to other drive type mobile robots, such as all-wheel-drive or autonomous vehicle. More research in this area is still in need of work in the future.

## Acknowledgement

This work is supported by the National Natural Science Foundation of China(61202376,61374039)and discipline of Management Science and Engineering(XXXKY1313). And it is also supported by Innovation Program of Shanghai Municipal Education Commission(B51JG13R003, 13YZ075).Thanks for the help.

## References

- [1] Ming Cao, Brian D. O. Anderson, A. Stephen Morse. Sensor network localization with imprecise distance measurements. *Systems and Control Letters*[J], **55**, 87-93 (2006).
- [2] Fidan, B., Dasgupta, S., Anderson, B. D. O., Guaranteeing Practical Convergence in Algorithms for Sensor and Source Localization. *IEEE Transactions on Signal Processing*[J], **56**, 4458-4469 (2008).
- [3] Junchuan Liu, Yuru Zhang ,Zhen Li. Improving the Positioning Accuracy of a Neurosurgical Robot System. *IEEE/ASME Transactions on Mechatronics*[J], **12**, 527-533 (2007).
- [4] N. M. Kwok, D. K. Liu, G. Dissanayake. Evolutionary computing based mobile robot localization. *Engineering Applications of Artificial Intelligence*, **19**, 857-868 (2006).
- [5] Andreasson, H., Treptow, A. , Duckett, T. Localization for Mobile Robots using Panoramic Vision, Local Features and Particle Filter. *Robotics and Automation*, 2005. *ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, **18-22**, 3348-3353 (2005).
- [6] M. Locatelli. On the multilevel structure of global optimization problems. *Computational Optimization and Applications*, **30**, 5-22 (2005).
- [7] Agostino Martinelli, Nicola Tomatis, Roland Siegwart. Simultaneous localization and odometry self calibration for mobile robot. *Autonomous Robots*, **22**, 75-85 (2007).
- [8] S. J. Julier. The stability of covariance inflation methods for SLAM. In *Proceedings of the 2003 IEEE IROS Conference*, Las Vegas NV, USA, 2749-2754 (2003).
- [9] Patric Jensfelt, Henrik I. Christensen. Pose Tracking Using Laser Scanning and Minimalistic Environmental Models. *IEEE Transactions on Robotics and Automation*, **17**, 138-147 (2001).
- [10] YANG Jing-Dong, YANG Jing-Hui, HONG Bing-Rong. An Efficient Approach to Odometric Error Modeling for Mobile Robots. *ACTA AUTOMATIC ASINICA*, **35**, 168-173 (2009).
- [11] Martinelli, A. A possible strategy to evaluate the odometry error of a mobile robot. *Intelligent Robots and Systems*, 2001. *Proceedings. 2001 IEEE/RSJ International Conference on*, **4**, 1946-1951 (2001).
- [12] D. G., Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, **60**, 91-110 (2004).
- [13] Lowe, D. G., Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, Corfu, Greece, 1150-1157 (1999).
- [14] Jim Mutch and David G. Lowe. Object class recognition and localization using sparse features with limited receptive fields[J]. *International Journal of Computer Vision*, **80**, 45-57 (2008).
- [15] Veenhuis, C., Koppen, M., Vicente-Garcia, R.. Evolutionary multi-objective optimization of Particle Swarm Optimizers. *IEEE Congress on Evolutionary Computation*, 2273-2280 (2007).
- [16] Zhong Qu. Two Algorithms of Image Segmentation and Measurement Method of Particles Parameters. *Applied Mathematics and Information Science.*, **6-1S**, 105S-109S (2012).
- [17] Stephen Se, David G. Lowe, James J. Little. Vision-Based Global Localization and Mapping for Mobile Robots. *IEEE TRANSACTIONS ON ROBOTICS*, **21**, 364-375 2005.
- [18] Kosecka, J. , Xiaolong Yang. Global localization and relative pose estimation based on scale-invariant features. *Pattern Recognition*, 2004. *ICPR 2004. Proceedings of the 17th International Conference on*, **4**, 319-322 (2004) .
- [19] C. Sagues, Author Vitae J. J. Guerrero. Visual correction for mobile robot homing. *Robotics and Autonomous Systems*, **50**, 41-49 (2005).
- [20] Zhongkun MA, Guy A. E. VANDENBOSCH. Comparison of Weighted Sum Fitness Functions for PSO Optimization of Wideband Medium-gain Antennas. *Radioengineering*, **21**, 504-511 2012.



**Jingdong Yang** is Associate Professor of Shanghai University Science and Technology, in Shanghai of China. He received the M.S. degree from the Department of Computer Science and Engineering of Dalian University of Technology. His Ph.D. was conferred at School of Computer Science and Engineering in Harbin Institute of Technology in China in 2008. His research focused on autonomous navigation for mobile robots, humanoid, and robot soccer, automation, autonomous navigation vehicles, and machine vision, etc.



**Jinghui Yang** is Professor of Business Management at Shanghai Second Polytechnic University. She was the academic leader of Logistics services and management disciplines. She received the M.S. degree from the Department of control science and engineering in 2000 and Ph.D. from the management Department in 2005 in Dalian University of Technology. And Her main research interests are Data Mining, Electronic Commerce, Logistics and information flow Management, Supply Chain management, etc.